DOCTORAL THESIS

Robust Speech Recognition on Intelligent Mobile Devices with Dual-Microphone

University of Granada

Author:
Iván López Espejo

Thesis supervisors:
Antonio Miguel Peinado Herreros
Ángel Manuel Gómez García

Ph.D. Program in Information and Communication Technologies
Department of Signal Theory, Telematics and Communications

Granada, September 2017

Editor: Universidad de Granada. Tesis Doctorales
Autor: Iván López Espejo
ISBN: 978-84-9163-401-0
URI: http://hdl.handle.net/10481/47934

The doctoral candidate **Mr. Iván López Espejo** and the thesis supervisors **Mr. Antonio Miguel Peinado Herreros** and **Mr. Ángel Manuel Gómez García** guarantee, by signing this Doctoral Thesis, that the work has been done by the doctoral candidate under the direction of the thesis supervisors and, as far as our knowledge reaches, in the performance of the work, the rights of other authors to be cited (when their results or publications have been used) have been respected.

Granada, June 8th 2017

Thesis supervisors Doctoral candidate

Sgd.: Antonio M. Peinado Sgd.: Ángel M. Gómez Sgd.: Iván López

To my parents, Isabel and Rafael...
I am who I am thanks to them.

None are more hopelessly enslaved
than those who falsely believe they are free.

Johann Wolfgang von Goethe

The philosophers have only interpreted the world
in various ways; the point, however, is to change it.

Karl Marx

Acknowledgements

First of all, I would like to thank my Ph.D. thesis supervisors, Antonio M. Peinado Herreros and Ángel M. Gómez García, for both their guide and valuable help throughout the development of this research, as well as for their friendliness. Moreover, I also wish to express special gratitude to José A. González López for his outstanding help and care in such a way that I can certainly consider him as my "third supervisor". Of course, I would like to extend my heartfelt appreciation to the other members of the group SigMAT (Signal Processing, Multimedia Transmission and Speech/Audio Technologies) for their comradeship, while a very special thank goes to my fellow Ján Koloda for his friendship over the recent years as well as for those experiences enjoyed together and those to come. I also have to make a special mention of Juan M. Martín Doñas for both his fellowship and help with my concerns about deep learning. I wish him all the best in his future endeavors.

Furthermore, I want to express my gratitude to the research group SpandH (Speech and Hearing) for their warm welcome during my visit at the University of Sheffield. In particular, I cannot forget Erfan Loweimi, who tried to make me feel at home all the time.

Also for its warm welcome I acknowledge all the das-Nano team and, more precisely, Eduardo Azanza Ladrón, who trusted me to be enrolled in a very exciting project with highly qualified professionals and great people.

Last but not least, I want to thank my parents, Isabel and Rafael, for all their help, understanding, dedication and love, as well as those colleagues who have accompanied me over the last years for their friendship and the good times: Jonathan Prados Garzón, Gonzalo Cardenete Burgos, Iván Fernández Bermejo, Emilio Marín López, Nadir Benamirouche, Igor Jauk and Juan Marín Sánchez (for all those unsuccessful attempts to create a new musical project and for allowing me to use the facilities of Curva Polar).

Abstract

Despite the outstanding progress made on automatic speech recognition (ASR) throughout the last decades, noise-robust ASR still poses a challenge. Tackling with acoustic noise in ASR systems is more important than ever before for a twofold reason: *1)* ASR technology has begun to be extensively integrated in intelligent mobile devices (IMDs) such as smartphones to easily accomplish different tasks (e.g. search-by-voice), and *2)* IMDs can be used anywhere at any time, that is, under many different acoustic (noisy) conditions.

On the other hand, with the aim of enhancing noisy speech, IMDs have begun to embed small microphone arrays, i.e. microphone arrays comprised of a few sensors close each other. These multi-sensor IMDs often embed one microphone (usually at their rear) intended to capture the acoustic environment more than the speaker's voice. This is the so-called secondary microphone. While classical microphone array processing (also known as beamforming) may be used for noise-robust ASR purposes, it is reported in the literature that its performance is quite limited when considering very few sensors close each other, one of them being a secondary microphone.

As a result, the main goal of this Thesis is to explore a new series of dual-channel algorithms exploiting a secondary sensor to improve ASR accuracy on IMDs being used in everyday noisy environments.

First, three dual-channel power spectrum enhancement methods are developed to circumvent the limitations of related single-channel feature enhancement methods when applied to such a dual-microphone set-up. These proposals have been referred to as DCSS (Dual-Channel Spectral Subtraction), P-MVDR (Power-Minimum Variance Distortionless Response) and DSW (Dual-channel Spectral Weighting, based on Wiener filtering). In particular, DSW starts from a simple formulation in which it is assumed that the secondary microphone only captures noise and the existence of a homogeneous noise field. Since it is known that both assumptions are not accurate, the Wiener filter (WF)-based weighting is modified through *1)* a bias correction term (to rectify the resulting spectral weights when a non-negligible speech component is present at the secondary channel), and *2)* a noise equalization (inspired by MVDR beamforming)

applied on the secondary channel before spectral weight computation. All of these techniques require knowledge of the relative speech gain (RSG) which relates the clean speech power spectra at the two channels. To obtain the RSG, a two-channel minimum mean square error (MMSE)-based estimator is also developed for this task in this Thesis.

In addition, the vector Taylor series (VTS) approach for noise-robust ASR has been widely applied over the last two decades in a successful manner. Then, VTS feature compensation is extended to be performed on a dual-channel framework in a similar fashion to the aforementioned power spectrum enhancement methods. The overarching element of this dual-channel VTS method is the stacked formulation. From this, an MMSE-based estimator for the log-Mel clean speech features, which relies on a VTS expansion of a dual-channel speech distortion model, is developed. The superiority of our dual-channel approach with respect to the single-channel one is also shown in this Thesis.

To conclude, two dual-channel deep learning-based contributions are presented to deal with the development of two complex (from an analytical point of view) tasks of a noise-robust ASR system. These tasks are missing-data mask and noise estimation, which are faced by taking benefit from the powerful modeling capabilities of deep neural networks (DNNs). More specifically, these DNNs exploit the power level difference (PLD) between the two available channels to efficiently obtain the corresponding estimates with good generalization ability. While missing-data mask and noise estimates can be employed in various ways for noise-robust ASR purposes, in this Thesis they are applied to spectral reconstruction and feature compensation, respectively.

It should be highlighted that our contributions broadly showed an outstanding performance at low signal-to-noise ratios (SNRs), which makes them promising techniques to be used in highly noisy environments such as those where IMDs might be employed.

Acronyms

Acronym	Meaning
AFE	Advanced Front-End
ANN	Artificial Neural Network
ASR	Automatic Speech Recognition
AURORA2-2C-CT	Aurora-2 - 2 Channels - Close-Talk
AURORA2-2C-FT	Aurora-2 - 2 Channels - Far-Talk
CDF	Cumulative Distribution Function
CMN	Cepstral Mean Normalization
CMVN	Cepstral Mean and Variance Normalization
D&S	Delay-and-Sum
DCSS	Dual-Channel Spectral Subtraction
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DNN	Deep Neural Network
DSR	Distributed Speech Recognition
DSW	Dual-channel Spectral Weighting
ED	Eigenvalue Decomposition
EM	Expectation-Maximization
ETSI	European Telecommunications Standards Institute
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
IFT	Inverse Fourier Transform
IIR	Infinite Impulse Response
IMD	Intelligent Mobile Device
LDA	Linear Discriminant Analysis
MAP	Maximum A Posteriori
MCWF	Multi-Channel Wiener Filtering
MFCC	Mel-Frequency Cepstral Coefficient

Acronym	Meaning
MMSE	Minimum Mean Square Error
MSE	Mean Square Error
MVDR	Minimum Variance Distortionless Response
NAT	Noise Adaptive Training (in Chapter 2) or Noise-Aware Training (in Chapters 5, 6 and 7)
NSR	Network-based Speech Recognition
P-MVDR	Power-Minimum Variance Distortionless Response
PDF	Probability Density Function
PLD	Power Level Difference
PSD	Power Spectral Density
RAP	Relative Acoustic Path
RBM	Restricted Boltzmann Machine
RSG	Relative Speech Gain
SNR	Signal-to-Noise Ratio
SPP	Speech Presence Probability
SS	Spectral Subtraction
STDFT	Short-Time Discrete Fourier Transform
STFT	Short-Time Fourier Transform
STQ	Speech and multimedia Transmission Quality
T-F	Time-Frequency
TDoA	Time Difference of Arrival
TGI	Truncated-Gaussian based Imputation
VAD	Voice Activity Detector
VTS	Vector Taylor Series
WAcc	Word Accuracy
WER	Word Error Rate
WF	Wiener Filter

Notation

Typographical conventions	
x	scalar variable
$\mathbf{x}$	vector
$\mathbf{X}$	matrix
$\hat{x}$	estimation of x

Symbols and operators	
$\approx$	approximately equal to
$\propto$	proportional to
$\odot$	element-wise vector multiplication
$*$	convolution
$\operatorname{argmin}_x f(x)$	value of x that minimizes $f(x)$
∇	gradient
$\mathcal{L}$	Lagrangian function

Vectors and matrices	
$\mathbf{x}$	column vector
$\mathbf{1}$	all-ones vector
$\mathbf{0}$	zero vector
$\mathbf{A}$	matrix
$\mathbf{A}^\top$	transpose of $\mathbf{A}$
$\mathbf{A}^{-1}$	inverse of $\mathbf{A}$
$\mathbf{I}$	identity matrix
$\operatorname{diag}(\mathbf{x})$	diagonal matrix with main diagonal $\mathbf{x}$
$\log(\mathbf{x})$	logarithm applied element-wise
$e^{\mathbf{x}}$	exponential function applied element-wise
$\mathbf{C}$	DCT matrix

Probability distributions

$P(\cdot)$	probability
$p(\cdot)$	probability density function
$p(x, y)$	joint density function of x and y
$p(x\|y)$	conditional density function of x given y
$\mathcal{N}(x\|\mu, \sigma)$	Gaussian distribution of mean μ and variance σ^2
$p(\mathbf{x})$	multivariate probability density function
$\mathcal{N}(\mathbf{x}\|\boldsymbol{\mu}, \boldsymbol{\Sigma})$	multivariate Gaussian distribution of mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$
$\mathrm{E}[x]$	expected value of x

Signals

$\mathbf{x}$	clean speech feature vector
$\mathbf{y}$	noisy speech feature vector
$\mathbf{n}$	additive noise feature vector
$\mathbf{h}$	convolutive noise vector

Indices

t	time frame index
f	frequency bin index
T	number of frames in an utterance
$\cdot_x$	clean speech subscript
$\cdot_y$	noisy speech subscript
$\cdot_n$	additive noise subscript

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

1.1 Motivation and overview

Intelligent mobile devices (IMDs) such as smartphones or tablets have revolutionized the way we live. They allow us to carry out a large variety of tasks that make our lives easier, e.g. communicating with other people anywhere at any time or searching for information instantaneously. These devices are pervasively used in our society in such a way that a large percentage of population from all around the world has at least one IMD. Indeed, this is reflected by the IMD sales growth year after year. For example, Figure 1.1 indicates the global smartphone sales in million units from 2010 to 2015. The sales increase over the last years has been spectacular, from less than 300 million smartphones sold in 2010 to nearly 1500 million in 2015.

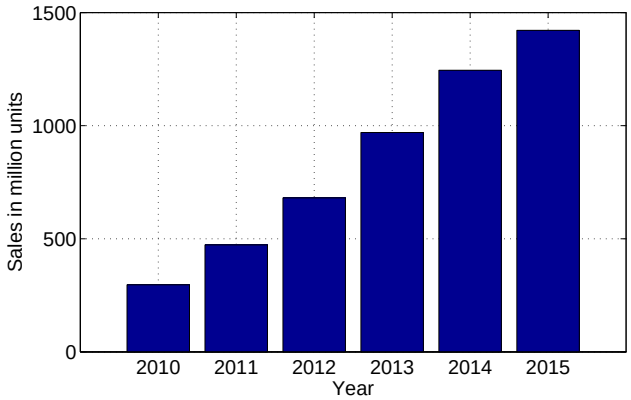

Figure 1.1: Global smartphone sales (in million units) from 2010 to 2015 [171].

Because of both the facts described above and the extraordinary computational power of recent IMDs, automatic speech recognition (ASR) has experienced a new upswing. ASR is a mature technology which has begun to be extensively integrated in IMDs in order to accomplish different tasks such as search-by-voice, dictation, voice control, and many other speech-enabled services. Despite this, ASR systems are still somewhat far from the speech recognition accuracy that the human being exhibits. At this respect, let us consider a small-vocabulary task consisting of the recognition of sequences of digits. In this case, while humans present an error rate below 0.009% [117], some of the best ASR systems achieve a rate not less than 0.55% [207]. Furthermore, such error rates as well as the performance gap between humans and machines are even higher for more complex and larger vocabulary tasks. For instance, in a telephonic conversation context, humans show an error rate around 4%, while ASR systems go up to 12% [25]. Several issues contribute to this performance gap between humans and machines and they are essentially related with the introduction of mismatch between the training and testing conditions of the ASR system (this will be further clarified during the next chapter). One of the most important factors contributing to the performance degradation of an ASR system is acoustic noise and, in particular, background (i.e. additive) noise. Thus, while human beings exhibit a high degree of robustness against noise when recognizing speech, this type of distortion can make ASR systems unusable even when integrating specific techniques to deal with it [117].

Mobile devices can be employed anywhere at any time, in such a way that coping with a wide variety of noisy environments is mandatory to ensure a good user experience when running speech recognition-based applications on these devices. In summary, precisely because of the proliferation of IMDs integrating ASR technology, tackling with noise is more important than ever before.

Over the recent years, with the aim of enhancing noisy speech, these devices have begun to embed small microphone arrays, i.e. microphone arrays comprised of a few sensors close each other. For example, Figure 1.2 illustrates a commercial smartphone embedding two microphones. Apart from the microphone located at the bottom of the device in order to be next to the speaker's mouth when having a conversation, a secondary sensor is placed at the rear of the smartphone. When the speaker talks on the phone, the latter sensor looks towards the environment to capture valuable information that can be used to perform noise cancellation in an easy and efficient manner.

According to the reasons discussed above, the main goal of this Thesis is therefore to design proper techniques to make ASR systems performing on IMDs robust to noise. More precisely, we will exploit the multi-channel information from the small microphone

Figure 1.2: Example of smartphone embedding two microphones the location of which is marked with red circles. Front (left) and back (right) sides of the smartphone are drawn.

arrays embedded in the latest IMDs to outperform related single-channel noise-robust methods. While classical microphone array processing may be used for this purpose, it is shown in the literature that its performance is quite limited under certain small microphone array frameworks [179, 180]. Hence, exploring new approaches in this context seems preferable to better exploit the features of IMDs with several sensors. Finally, it must be pointed out that we will focus on tackling additive/background noise, while convolutive noise due to reverberation and other channel effects is out of the scope of this Thesis.

1.2 Automatic speech recognition

Automatic speech recognition (ASR) refers to a process, conducted by a machine, consisting of the conversion of spoken words into a machine-readable transcription. This transcription can be further manipulated for example to provide to the end user a written version of the sequence of words recognized from the input speech. A very brief introduction to the basics of ASR systems is given in this section.

It can be considered that an ASR system is comprised of two differentiated main modules: front-end and back-end. The front-end is devoted to the extraction of speech features that are appropriate for later recognition. This module is reviewed in Subsection 1.2.1. On the other hand, the back-end, which is introduced in Subsection 1.2.2, is responsible for carrying out the actual speech recognition process from the speech features extracted by the front-end.

quency cepstrum coefficients (MFCC) [20], which arise from a homomorphic transform of the short-term spectrum expressed on a mel frequency scale. Figure 2.2 shows how they may be computed. Their use is motivated by both perceptual and performance aspects. In speech production, the vocal tract may be viewed as a filter acting on a sound source, such as the glottis—this is the source-filter model [57, 72, 125]. In continuous speech, it has been noted that the vocal tract changes shape slowly in continuous speech; therefore at small enough time scales, on the order of 10 ms, it may be considered a filter of fixed characteristics [125]. Hence, a short-time Fourier transform is applied, converting the time domain signal into the frequency or spectral domain. A first-order pre-emphasis filter is usually applied to accen-

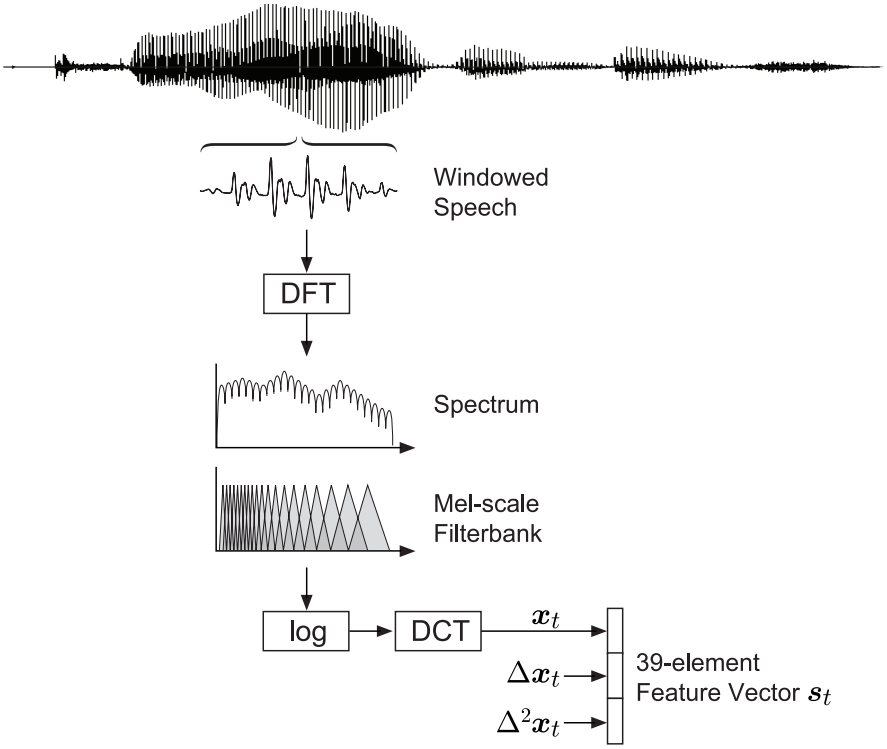

Figure 2.2: Front-end processing for MFCC. Speech waveform converted to smoothed short-term log spectrum every 10 ms. Discrete cosine transform is applied and dynamic terms appended to produce the complete feature vector $\boldsymbol{s}_t$.

Figure 1.3: Diagram of a typical feature extractor for ASR purposes [112].

1.2.1 Speech feature extraction

The purpose of the front-end stage is to extract a series of parameters useful for speech recognition. These parameters, known as speech features, must meet certain desirable properties. Thus, they should represent the most relevant speech characteristics that allow us discriminating among the variety of linguistic units with the less amount of coefficients as possible. In other words, they should be both discriminative and compact, respectively. Moreover, as much as possible, these parameters should be robust (namely insensitive) to different acoustic variations, e.g. ambient noise or inter-speaker variabilities. Several types of speech parameterizations meeting the aforementioned requirements have been proposed in the literature over the years. In fact, some of them such as PNCCs (Power-Normalized Cepstral Coefficients) [101] or PLP (Perceptually-based Linear Prediction) [74] will be commented later in Subsection 2.2.1. Nevertheless, to this day, the most widely used type of speech features are the so-called MFCCs (Mel-Frequency Cepstral Coefficients) [32], which will also be employed throughout the development of this Thesis. Because of this, let us briefly examine how MFCCs work from the diagram shown in Figure 1.3.

First, the speech signal being captured by a microphone is digitized to be converted into a discrete-time signal. Since most of the significant speech energy is concentrated below 5 kHz, it is typical to make use of a sampling rate of 8 kHz or 16 kHz. Then, the discretized speech signal is segmented into overlapping frames of time length between 20 ms and 30 ms, in such a manner that the signal in each frame can be assumed quasi-stationary. To enhance the high frequency components of speech, a pre-emphasis filtering is applied to the signal. The resulting filtered signal is typically windowed by means of a Hamming window, and then transformed into the frequency domain by application of the discrete Fourier transform (DFT). Based on the Mel scale [173] which approximates the frequency resolution of the human ear, a Mel filterbank is employed to transform the magnitude or power spectrum of the speech signal to the Mel-frequency domain. To model the perceptual sensitivity of the human ear, the dynamic range of each filterbank output is compressed by application of the natural logarithm. Since these speech features are still highly correlated, their discrete cosine transform (DCT) is computed to get a more compact representation. This set of coefficients is what we know as MFCCs. This highly decorrelated speech representation is very appropriate to implement the acoustic models of the recognizer in an efficient way involving relatively few parameters. In particular, it is usual to employ 13 MFCCs per each time frame. Finally, if Gaussian mixture models (GMMs) are considered for acoustic modeling, the first and second derivatives of these coefficients are appended to them in order to form a 39-dimensional speech feature vector, which is then used for recognition. These derivatives try to capture the non-stationary behavior of the speech signal over time, something convenient to overcome the limited temporal modeling of hidden Markov models (HMMs) [112]. Alternatively, if artificial neural networks (ANNs) are employed for acoustic modeling, a temporal context is appended to each 13-dimensional MFCC feature vector to allow us learning the speech dynamics.

1.2.2 Back-end

The speech recognition problem can be stated as follows. First, let $\mathbf{X} = (\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_{T-1})$ be a T long sequence of feature vectors extracted from a speech signal. Then, the objective in ASR is to find the most likely sequence of words $\mathbf{W} = (w_1, w_2, ..., w_m)$ from the set of features $\mathbf{X}$. This can be formulated as a maximum a posteriori (MAP) estimation problem. More precisely, the goal of the back-end stage is to find the sequence of words $\hat{\mathbf{W}}$ that maximizes the probability $P(\mathbf{W}|\mathbf{X})$. In practice, this is achieved by the application of the Viterbi algorithm [189], which allows us the decoding of $\mathbf{W}$ from the observations $\mathbf{X}$. This MAP estimation problem can be expressed using the Bayes'

rule in the following way:

$$\begin{aligned} \hat{\mathbf{W}} &= \text{argmax}_{\mathbf{W}}\, P(\mathbf{W}|\mathbf{X}) \\ &= \text{argmax}_{\mathbf{W}}\, \frac{p(\mathbf{X}|\mathbf{W})P(\mathbf{W})}{p(\mathbf{X})} \\ &= \text{argmax}_{\mathbf{W}}\, p(\mathbf{X}|\mathbf{W})P(\mathbf{W}), \end{aligned} \tag{1.1}$$

where $p(\mathbf{X}|\mathbf{W})$ and $P(\mathbf{W})$ are known as the acoustic and language scores, respectively. To find out $p(\mathbf{X}|\mathbf{W})$, which gives us the probability of observing the set of features $\mathbf{X}$ given the phrase $\mathbf{W}$, we require both the lexicon (i.e. the mapping between the written words that can be recognized and the word phonetic transcriptions) and the acoustic model of the recognizer. In addition, $P(\mathbf{W})$ is the prior probability of the sequence of words $\mathbf{W}$. This probability is given by the language model of the recognizer. Some comments on both language and acoustic modeling are given immediately below.

1.2.2.1 Language modeling

As aforementioned, the prior probability of the word sequence hypothesis $\mathbf{W}$ involved in (1.1), $P(\mathbf{W})$, can be determined through the language model of the recognizer. According to its definition, we can see that this probability does not depend on the observed speech signal but only on the language characteristics and, more in particular, on the linguistic task taken into account by the ASR system. The N-gram statistical approach has traditionally been the most popular to deploy language modeling. In an N-gram context, the probability of a word w_i is modeled given the known $N-1$ preceding words $(w_{i-1}, ..., w_{i-N+1})$. Thus, $P(\mathbf{W})$ can be computed as

$$P(\mathbf{W}) = \prod_{i=1}^{m} P(w_i|w_{i-1}, ..., w_{i-N+1}), \tag{1.2}$$

where it is assumed that the sequence $\mathbf{W}$ is comprised of a total of m words. In practice, it is usual to employ bigram ($N = 2$) or trigram ($N = 3$) language models.

Over the last years, there has been a transition from standard N-gram techniques to connectionist approaches for language modeling. For instance, recurrent neural networks (RNNs) are nowadays widely used to fit a probabilistic model to compute $P(\mathbf{W})$. Connectionist language models have shown to be superior to standard N-gram approaches except for their higher computational complexity [135].

entry and exit states of a HMM are non-emitting. This is to facilitate the construction of composite models as explained in more detail later.

The joint probability that $\boldsymbol{O}$ is generated by the model M moving through the state sequence X is calculated simply as the product of the transition probabilities and the output probabilities. So for the state sequence X in Fig. 1.3

$$P(\boldsymbol{O}, X|M) = a_{12}b_2(\boldsymbol{o}_1)a_{22}b_2(\boldsymbol{o}_2)a_{23}b_3(\boldsymbol{o}_3)\ldots \tag{1.4}$$

However, in practice, only the observation sequence $\boldsymbol{O}$ is known and the underlying state sequence X is hidden. This is why it is called a *Hidden Markov Model*.

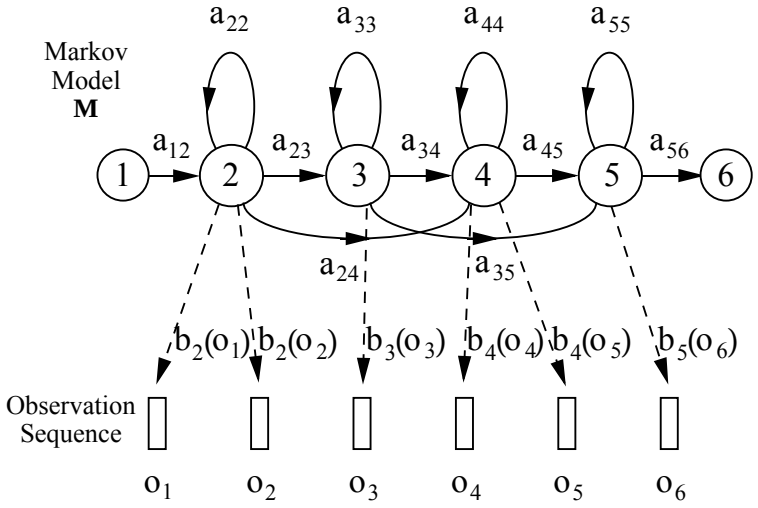

Figure 1.4: Example of hidden Markov model (HMM) [204].

Fig. 1.3 The Markov Generation Model

1.2.2.2 Acoustic modeling

Given that X is unknown, the required likelihood is computed by summing over all possible state sequences $X = x(1), x(2), x(3), \ldots, x(T)$, that is

$$P(\boldsymbol{O}|M) = \sum_{X} a_{x(0)x(1)} \prod_{t=1}^{T} b_{x(t)}(\boldsymbol{o}_t) a_{x(t)x(t+1)} \tag{1.5}$$

The acoustic model is responsible for providing $p(\mathbf{X}|\mathbf{W})$, which explains the likelihood of the set of speech features $\mathbf{X}$ given the word sequence hypothesis $\mathbf{W}$. To this end, every word $w_i \in \mathbf{W}, (i = 1, \ldots, n)$, is usually decomposed into simpler acoustic units such as monophones or triphones from the lexicon of the recognizer. In the current state-of-the-art ASR systems, each of these basic units is typically modeled by means of an HMM with continuous density functions. It must be noticed that HMMs are very appropriate to model time-varying signals. Therefore, each word is represented by the concatenation of several HMMs modeling the corresponding sequence of basic acoustic units contained in the word.

An example of HMM can be seen in Figure 1.4. An HMM is defined by the following elements:

- A set of S states interconnected, $(s_1, \ldots, s_S)$. For example, when an HMM is employed to model a monophone, that is often comprised of $S = 5$ states.

- A set of transition probabilities $\{a_{ij};\ i, j = 1, \ldots, S\}$. Each a_{ij} models the probability of moving from state s_i to state s_j.

- A set of output observation distributions $\{b_j(\mathbf{o});\ j = 1, \ldots, S\}$, where the observed variable $\mathbf{o}$ is the speech feature vector $\mathbf{x}_t$ within an ASR context. Each distribu-

tion $b_j(\mathbf{o} = \mathbf{x}_t)$ express the probability that the feature vector $\mathbf{x}_t$ is observed at state s_j.

The HMM of Figure 1.4 has a total of $S = 6$ states, and the beginning and ending states emit no output symbol. The topology of this HMM is referred to as Bakis, where only left-to-right transitions between states are permitted (the rest of transitions a_{ij} are forced to be zero). Due to the temporal dynamics of the speech signal, the Bakis topology is the one considered for ASR purposes. Another well-known HMM topology is that known as ergodic (i.e. fully-connected), where each state shares a connection with each other.

In practice, the underlying assumption of HMM modeling is that the speech signal is a first-order Markov stochastic process. As a result, the probability of being in state s_j at time t only depends on the state visited at time $t-1$, s_i, that is,

$$P(q_t = s_j | q_{t-1} = s_i, ..., q_1 = s_k) = P(q_t = s_j | q_{t-1} = s_i). \tag{1.3}$$

Thus, $\mathbf{q} = (q_0, ..., q_{T-1})$ is the sequence of states of the model visited over time. These probabilities are required to compute $p(\mathbf{X}|\mathbf{W})$ as specified below.

Again for simplicity, it is assumed in practice that the probability of observing the speech feature vector $\mathbf{x}_t$ only depends on the current state $q_t = s_j$. Until a few years ago, GMMs were widely employed to model the output observation distributions of the HMM states. In the case of using GMMs, the probability density function (PDF) of state s_j, $b_j(\mathbf{x}_t|s_j)$, is expressed as

$$b_j(\mathbf{x}_t|s_j) = \sum_{k=1}^{\mathcal{K}} P(k|s_j) \mathcal{N}\left(\mathbf{x}_t \left| \boldsymbol{\mu}_{s_j}^{(k)}, \boldsymbol{\Sigma}_{s_j}^{(k)} \right.\right), \tag{1.4}$$

where $\mathcal{K}$ is the total number of Gaussian components, and $P(k|s_j)$ is the prior probability of the k-th Gaussian component $\mathcal{N}\left(\mathbf{x}_t \left| \boldsymbol{\mu}_{s_j}^{(k)}, \boldsymbol{\Sigma}_{s_j}^{(k)} \right.\right)$ with mean vector and covariance matrix $\boldsymbol{\mu}_{s_j}^{(k)}$ and $\boldsymbol{\Sigma}_{s_j}^{(k)}$, respectively. Nevertheless, over the recent years, the use of ANNs in replacement of GMMs for acoustic modeling has become pervasive because of the better modeling capabilities of the former [82]. Hence, nowadays, it is preferred to employ deep learning architectures such as deep feedforward neural networks instead of GMMs to produce the state emission likelihoods.

The parameters of the HMMs, namely the transition probabilities of (1.3) and the parameters of the generative model of (1.4) (in the case of using GMMs), are then estimated from a training dataset. Such an estimation is iteratively performed by means of the Baum-Welch algorithm [16] to generate the acoustic model for speech recognition. Once done, $p(\mathbf{X}|\mathbf{W})$ can be calculated by summing over all the possible

state sequences $\mathbf{q} = (q_0, ..., q_{T-1})$ able to produce the word sequence $\mathbf{W}$, that is,

$$p(\mathbf{X}|\mathbf{W}) = \sum_{\mathbf{q}} \prod_{t=0}^{T-1} p(\mathbf{x}_t|q_t) P(q_t|q_{t-1}). \tag{1.5}$$

Finally, we must point out that a macromodel λ is defined from the integration of the acoustic and language models. This macromodel is then used to estimate, by means of the Viterbi algorithm [189], the optimal state sequence $\hat{\mathbf{q}}$ as

$$\hat{\mathbf{q}} = \underset{\mathbf{q}}{\operatorname{argmax}}\, p(\mathbf{q}, \mathbf{X}\,|\lambda)\ , \tag{1.6}$$

from which the word sequence $\mathbf{W}$ of the utterance is recovered.

1.3 Objectives of this Thesis

As we have introduced, ASR systems still suffer from accuracy issues when deployed in noisy environments. Currently, this problem is more important than ever before because of the pervasive use of ASR-based applications running on mobile devices, which can be employed anywhere, at any time. Indeed, tackling with background noise is mandatory to provide a good user experience. Since noise-robust ASR is still an open issue despite all the progress made over the last decades, the key objective of this Thesis is to make further advances on that topic while focusing on a mobile device scenario. Because many IMDs embed small microphone arrays (i.e. arrays comprised of a few number of sensors very close each other), we want to exploit the multi-channel information from them in order to outperform the classical single-channel noise-robust ASR approaches. Moreover, we also know that the performance of classical beamforming with small microphone arrays is limited [179, 180] and, therefore, it is important to develop specific solutions to work successfully in this scenario. More precisely, we can highlight the following objectives:

1. To carry out a literature review about single-channel noise-robust ASR as well as about those multi-channel noise-robust speech processing methods especially targeted at mobile environments.

2. IMDs often embed one microphone (usually at their rear) intended to capture the acoustic environment more than the speaker's voice. This is the so-called secondary microphone. Hence, another objective is to develop a new series of dual-channel algorithms exploiting a secondary sensor to improve the ASR accuracy on IMDs being used in everyday noisy environments.

3. To generate new speech resources under a dual-channel mobile device framework for experimental purposes.

4. To evaluate our developments and comparing them with other state-of-the-art techniques to draw conclusions in order to make further progress.

1.4 Thesis organization

This Thesis is comprised of a total of seven chapters, one of them being the present introduction, along with three appendices. In particular, the last appendix is a comprehensive summary written in Spanish in order to meet with the requirements imposed by the University of Granada regarding the drafting of the doctoral dissertation. The theoretical foundations of this Thesis are stated in Chapter 2. Then, Chapters 3, 4 and 5 are intended to describe our contributions on multi-channel noise-robust ASR on IMDs. The organization of such contributions into three different chapters has been done accounting for both the type of noise-robust approach and the working domain. Finally, in Chapters 6 and 7 the experimental evaluation and the conclusions are shown, respectively. More specifically:

- In Chapter 2, a literature review is carried out to present the theoretical fundamentals that justify our developments. In turn, this chapter is composed of five sections. The speech distortion model serving as a basis to develop a variety of approaches for noise-robust ASR purposes and the effects of the acoustic noise on the speech distribution are presented in the first section. Then, the fundamentals of single-channel robust speech recognition and noise estimation are described. To conclude, apart from a summary, the foundations of multi-channel robust speech processing on IMDs are reviewed. We will focus on beamforming since microphone array processing is typically used along with single-channel noise-robust processing techniques to provide robustness against noise in ASR systems performing on IMDs with several sensors. Additionally, the dual-channel power level difference (PLD) concept is discussed as it is a driving principle of our noise-robust contributions.

- Three dual-channel power spectrum enhancement proposals are formulated in Chapter 3: DCSS (Dual-Channel Spectral Subtraction), P-MVDR (Power-Minimum Variance Distortionless Response) and DSW (Dual-channel Spectral Weighting). DCSS and P-MVDR are basic enhancement methods relying on spectral subtraction (SS) and MVDR principles, respectively. On the other hand, DSW

is based on Wiener filtering and it integrates a noise equalization procedure also based on the MVDR principle. All of them assume that the mobile device has only one front (primary) sensor, as well as a rear microphone to better capture the acoustic environment. Hence, a combinatorial strategy integrating beamforming is presented to be followed in the case of an IMD with more than one front sensor in order to condense the multi-channel information into only two channels. Finally, since all these enhancement proposals require knowledge about the relative speech gain (RSG) between the two available channels, a complex MMSE-based RSG estimation method is also developed.

- In Chapter 4, a dual-channel vector Taylor series (VTS) feature compensation (i.e. enhancement) technique is set out. Once the dual-channel distortion model is properly revisited, the formulae derived from a stacked scheme are shown. Besides this scheme, a more robust alternative approach for the calculation of the posterior probabilities is studied, in which the distortion model at the secondary channel is conditioned to the noisy observation from the primary channel.

- The use of deep learning for dual-channel noise-robust ASR on IMDs is explored in Chapter 5. This chapter begins with a brief revision of applied deep learning procedures. While this overview is focused on speech processing, its inclusion at this point was preferred as we briefly review some deep learning architectures that have nothing to do with the fundamentals of noise-robust speech processing. Then, dual-channel deep neural network (DNN)-based missing-data mask and noise estimation techniques are described. These exploit the dual-channel information in synergy with the powerful modeling capabilities of DNNs to provide accurate estimates in an efficient manner.

- Thereafter, in Chapter 6 the experimental evaluation is presented. Apart from the summary, this chapter contains two more sections: one about the experimental framework and another containing the experimental results. In the first one, the multi-channel speech resources employed for experimental purposes, namely the AURORA2-2C-CT/FT and CHiME-3 databases, are described along with the feature extraction process and the back-end setup of the recognizer. In particular, we must highlight the AURORA2-2C-CT/FT corpora as another contribution of this Thesis. The AURORA2-2C-CT/FT databases emulate the acquisition of noisy speech in everyday environments by means of a dual-microphone smartphone. Then, the word accuracies and/or word error rates achieved by our contributions and comparison techniques are shown. These results are properly

discussed and organized by considering the same contribution chapter structure of this dissertation.

- Finally, the conclusions of this Thesis are presented in Chapter 7 along with a summary of our contributions and future work.

1.5 List of publications and awards

The following publications have been produced as a result of the work in this Thesis (in reverse chronological order):

1. I. López-Espejo, A. M. Peinado, A. M. Gomez and J. A. Gonzalez: *Dual-Channel Spectral Weighting for Robust Speech Recognition in Mobile Devices*. Submitted to Digital Signal Processing.

2. I. López-Espejo, A. M. Peinado, A. M. Gomez and J. A. González: *Dual-Channel VTS Feature Compensation for Noise-Robust Speech Recognition on Mobile Devices*. IET Signal Processing, 11:17–25, 2017.

3. I. López-Espejo, A. M. Peinado, A. M. Gomez and J. M. Martín-Doñas: *Deep Neural Network-Based Noise Estimation for Robust ASR in Dual-Microphone Smartphones*. Lecture Notes in Computer Science, 10077:117–127, 2016.

4. I. López-Espejo, J. A. González, A. M. Gómez and A. M. Peinado: *DNN-Based Missing-Data Mask Estimation for Noise-Robust ASR in Dual-Microphone Smartphones*. In *Proceedings of UKSpeech, July 2–3, Norwich (UK)*, 2015.

5. I. López-Espejo, J. A. González, A. M. Gomez and A. M. Peinado: *A Deep Neural Network Approach for Missing-Data Mask Estimation on Dual-Microphone Smartphones: Application to Noise-Robust Speech Recognition*. Lecture Notes in Computer Science, 8854:119–128, 2014.

6. I. López-Espejo, A. M. Gomez, J. A. González and A. M. Peinado: *Feature Enhancement for Robust Speech Recognition on Smartphones with Dual-Microphone*. In *Proceedings of 22nd European Signal Processing Conference, September 1–5, Lisbon (Portugal)*, 2014.

Moreover, we must highlight that the quality of two of these publications has been endorsed by two international awards:

- Best paper award at IberSPEECH 2016 for the work *Deep Neural Network-Based Noise Estimation for Robust ASR in Dual-Microphone Smartphones*.
- Best student paper award at EUSIPCO (European Signal Processing Conference) 2014 for the work *Feature Enhancement for Robust Speech Recognition on Smartphones with Dual-Microphone*.

CHAPTER 2

Fundamentals of Single- and Multi-Channel Robust Speech Processing

In the introductory chapter we outlined that the performance of every automatic speech recognition (ASR) system can be severely degraded when there exists mismatch between the training and testing conditions. A possible source of mismatch is that referred to the variability inherent to different speakers. In this regard, some factors increasing the variability of the speech signal are gender, age, mood, presence of illness, etc. For instance, these factors have a direct impact on the inter-speaker variability and, therefore, considering a different set of training and testing speakers is an important mismatch source. Furthermore, differences in the length and shape of the vocal tract, dialect or pronunciation are other features that have influence in that sense [145]. Another source of degradation of the ASR system performance might be the channel (as well as the speech coding scheme) used to transmit the speech signal, which is likely to distort the signal as it behaves as a filter the response of which is often far from being flat. A related practical example where this might occur is in the case of network-based speech recognition (NSR) [149], a block diagram of which is depicted in Figure 2.1. For instance, let us think about carrying out ASR by means of a mobile device for search-by-voice purposes. Assuming that an NSR scheme is followed, first, the mobile device captures the voice of the speaker. Then, this is compressed by a speech codec and transmitted through the channel. Finally, the speech features are extracted and used for recognition both on the network side [149], so that the speech coding scheme and the transmission channel potentially introduce mismatch in ASR.

other information through a data channel.

On the other hand, the main advantage of NSR is that there is no need for modifying the existing clients in the case of mobile telephony networks.

In the same way as speech codecs are standardized for mobile telephony or VoIP, it is advised that a standardized feature extractor and encoder be used in DSR clients. The implementation of RSR systems over heterogeneous networks can also be eased by using DSR standards. Figure 1.5 shows two possible scenarios that mix mobile and IP

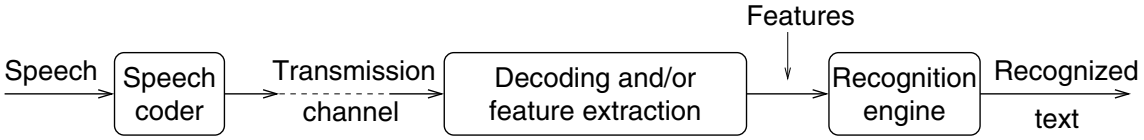

Figure 1.3 Scheme of a network speech recognition (NSR) system.

Figure 2.1: Block diagram of a network-based speech recognition scheme [149].

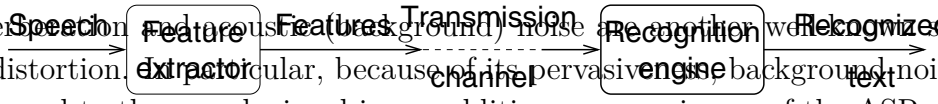

Figure 1.4 Scheme of a distributed speech recognition (DSR) system.

Reverberation and acoustic (background) noise are another well-known sources of speech distortion. In particular, because of its pervasiveness, background noise, which is overlapped to the speech signal in an additive manner, is one of the ASR mismatch sources that has been faced the most over the last decades. In DSR, providing robustness against noise in ASR is of particular importance nowadays due to the wide use of intelligent mobile devices (IMDs) such as smartphones or tablets, which are also employed for ASR purposes [107] as introduced in Chapter 1. Mobile devices can be employed anywhere at any time, in such a way that coping with a wide variety of noisy environments is mandatory to provide a good user experience. For example, some typical mobile usage scenarios are streets (where we can find a range of noise sources such as traffic, construction works or babble), offices (e.g. air-conditioning system and computer noises) or inside vehicles (e.g. engine noise). Many techniques have been proposed to improve ASR robustness against environmental noise and this chapter tries to give an overview of them while introducing the foundations of noise-robust processing for both single- and multi-channel ASR. This will serve as a presentation of the theoretical basis from which the different noise-robust contributions of this Thesis are built upon.

The rest of this chapter is structured as follows. First, the general speech distortion modeling, considered as the basic mathematical framework both to review the noise-robust state-of-the-art approaches and to develop our contributions throughout the following chapters, is explained in Section 2.1. Then, in Section 2.2, the single-channel robust speech recognition fundamentals required to understand our techniques are presented. The revisited approaches in this section have roughly been categorized into four different classes: feature-space, model-based and missing-data approaches, and distortion modeling by vector Taylor series (VTS). For some of these noise-robust methods, a module that accurately estimates the background noise that contaminates speech is necessary. Since this is an important related task, a brief overview of the noise estimation algorithms is provided in Section 2.3. The basis of multi-channel robust speech processing to be considered for ASR purposes on IMDs (the main scope of this Thesis) is given in Section 2.4. As shall be seen, in accordance with the literature, multi-channel robust ASR is mainly based on the combined use of microphone array

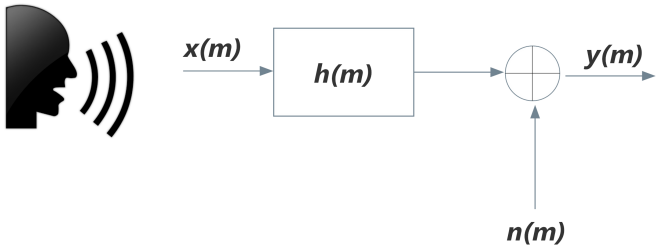

Figure 2.2: Block diagram of a speech distortion model including convolutive and additive noise as environmental distortions.

processing and single-channel noise-robust approaches. This kind of solutions jointly exploits the various embedded microphones in order to further improve the recognition performance regarding the classical single-channel strategies. For this reason, this section emphasizes on beamforming techniques, including post-filtering to mitigate their shortfalls. Finally, the dual-channel power level difference (PLD) concept is discussed as it is a driving principle of our noise-robust methods to be applied on IMDs with several sensors. To conclude the chapter, a summary is presented in Section 2.5.

2.1 Speech distortion modeling

In order to design analytical solutions to be applied for noise-robust ASR purposes, it is necessary to define a mathematical framework whereby we model the interactions between the speech signal of interest and the environmental distortions affecting that signal. For the past two decades, the linear speech distortion model first reported in [5] has been widely adopted as a standard mathematical framework to develop noise-robust methods for ASR. This speech distortion model will be considered from now on and it is represented in Figure 2.2. It consists of the clean speech signal of interest as produced by the speaker, $x(m)$, which is convolved by $h(m)$ representing the channel distortion plus the additive background noise $n(m)$. This defines the speech distortion model

$$y(m) = h(m) * x(m) + n(m), \tag{2.1}$$

where $y(m)$ is the resulting noisy speech signal, and m and $*$ refer to the discrete-time index (i.e. sample index) and the convolution operator, respectively. First, $h(m)$ may eventually characterize the channel impulse response (including the response of the microphone) and/or reverberation. Additionally, $n(m)$ models the total background noise as captured by the system and shaped as the sum of the different contributions

produced by the environmental noise sources. In the case of multiple microphones simultaneously recording a noisy speech signal, a subscript, k, will be used to differentiate which sensor a signal comes from. For example, if we have an array comprised of N microphones, the noisy signal coming from the k-th sensor is denoted as

$$y_k(m) = h_k(m) * x_k(m) + n_k(m), \quad k = 1, ..., N, \tag{2.2}$$

where, admittedly, the different variables $h_k(m)$, $x_k(m)$ and $n_k(m)$ depend on the corresponding sensor. Without loss of generality and for the sake of clarity, the reference to the sensor that captures the signal is omitted in the rest of this section.

As we said in the previous chapter, the most popular type of speech parameterization for ASR is Mel-frequency cepstral coefficients (MFCCs) [32]. Therefore, the speech distortion model in (2.1) is developed in the following until obtaining a relation between the clean speech and the distortions in terms of MFCCs.

First of all, the short-time discrete Fourier transform (STDFT) is applied on (2.1) to express this model in the linear frequency domain as

$$Y(f,t) = H(f,t)X(f,t) + N(f,t), \tag{2.3}$$

where f and t refer to the frequency bin and time frame index, respectively. It must be noticed that Eq. (2.3) implicitly assumes that the length of $h(m)$ is shorter than that of the analysis window. Because of this reason, that equation in the linear frequency domain is not valid when $h(m)$ represents a filter modeling long reverberation times and, therefore, it is a source of modeling errors which impacts on the ASR system performance [6, 107]. From (2.3), it is straightforward to characterize the speech distortion model in the linear power spectral domain as

$$\begin{aligned} |Y(f,t)|^2 &= |H(f,t)X(f,t) + N(f,t)|^2 \\ &= |H(f,t)|^2|X(f,t)|^2 + |N(f,t)|^2 + 2\cos(\alpha_{f,t})\,|H(f,t)||X(f,t)||N(f,t)|, \end{aligned} \tag{2.4}$$

where $\alpha_{f,t}$ is the relative phase between $H(f,t)X(f,t)$ and $N(f,t)$. While there exist some research works successfully exploiting the term $\alpha_{f,t}$ for noise-robust ASR purposes such as [35, 48], the common practice is to neglect it [6, 65, 108, 137, 138, 176]. This can be considered a compromise since the potential degradation of the ASR system performance as a result of neglecting $\alpha_{f,t}$ might compensate the difficulty in manipulating this term. Furthermore, another argument in favor of such a simplification is that the expected value of $\cos(\alpha_{f,t})$ is zero. Hence, (2.4) can be simplified as

$$|Y(f,t)|^2 = |H(f,t)|^2|X(f,t)|^2 + |N(f,t)|^2. \tag{2.5}$$

Then, a filterbank distributed in frequency in accordance with a human perceptual scale is employed to mimic the human auditory system. In particular, MFCCs consider the perceptual Mel scale proposed by S. S. Stevens *et al.* in [173]. As a result, we have a set of L filters with triangular-shaped frequency windows each, which are equidistant in Mel-frequency domain. Let W_{lf} be the frequency response of the l-th filter with $W_{lf} \geq 0$ and $\sum_f W_{lf} = 1$ $(l = 1, ..., L)$, Eq. (2.5) is transformed into the Mel power spectral domain as

$$\begin{aligned} |\tilde{Y}(l,t)|^2 &= \textstyle\sum_f W_{lf}|Y(f,t)|^2 \\ &= \textstyle\sum_f W_{lf}\left(|H(f,t)|^2|X(f,t)|^2 + |N(f,t)|^2\right) \\ &= \textstyle\sum_f W_{lf}|H(f,t)|^2|X(f,t)|^2 + \sum_f W_{lf}|N(f,t)|^2, \end{aligned} \tag{2.6}$$

where $|\tilde{Y}(l,t)|^2$ is the noisy speech Mel power spectrum bin at channel l and time frame t. From defining the clean speech, noise and channel Mel power spectrum bins, respectively, as

$$\begin{aligned} |\tilde{X}(l,t)|^2 &= \textstyle\sum_f W_{lf}|X(f,t)|^2; \\ |\tilde{N}(l,t)|^2 &= \textstyle\sum_f W_{lf}|N(f,t)|^2; \\ |\tilde{H}(l,t)|^2 &= \dfrac{\sum_f W_{lf}|H(f,t)|^2|X(f,t)|^2}{|\tilde{X}(l,t)|^2}, \end{aligned} \tag{2.7}$$

the speech distortion model in the Mel power spectral domain is expressed as follows:

$$|\tilde{Y}(l,t)|^2 = |\tilde{H}(l,t)|^2|\tilde{X}(l,t)|^2 + |\tilde{N}(l,t)|^2. \tag{2.8}$$

Then, the natural logarithm is applied to (2.8) in order to mimic the amplitude resolution of the human ear. First, let us define the following vectors containing log-Mel power spectrum coefficients:

$$\begin{aligned} \mathbf{y} &= \left(\log|\tilde{Y}(1,t)|^2, ..., \log|\tilde{Y}(L,t)|^2\right)^\top; \\ \mathbf{x} &= \left(\log|\tilde{X}(1,t)|^2, ..., \log|\tilde{X}(L,t)|^2\right)^\top; \\ \mathbf{h} &= \left(\log|\tilde{H}(1,t)|^2, ..., \log|\tilde{H}(L,t)|^2\right)^\top; \\ \mathbf{n} &= \left(\log|\tilde{N}(1,t)|^2, ..., \log|\tilde{N}(L,t)|^2\right)^\top. \end{aligned} \tag{2.9}$$

These are defined for a particular time frame while, for the sake of simplicity, an explicit reference to t has been omitted for the variables $\mathbf{y}$, $\mathbf{x}$, $\mathbf{h}$ and $\mathbf{n}$. From the vectors in (2.9), Eq. (2.8) can be written in the log-Mel power spectral domain as

$$\mathbf{y} = \mathbf{x} + \mathbf{h} + \log\left(\mathbf{1} + e^{\mathbf{n}-\mathbf{x}-\mathbf{h}}\right), \tag{2.10}$$

where it must be noticed that the operators $\log(\cdot)$ and $e^{(\cdot)}$ are applied element-wise and $\mathbf{1}$ is an L-dimensional vector filled with ones. Finally, the MFCCs widely used for ASR are obtained by application of the discrete cosine transform (DCT) to the log-Mel coefficients of (2.10). This is achieved by means of the DCT matrix $\mathbf{C}$:

$$\mathbf{y}^c = \mathbf{x}^c + \mathbf{h}^c + \mathbf{C}\log\left(\mathbf{1} + e^{\mathbf{C}^{-1}(\mathbf{n}^c-\mathbf{x}^c-\mathbf{h}^c)}\right), \tag{2.11}$$

where $\mathbf{C}^{-1}$ is the pseudoinverse of $\mathbf{C}$ as the DCT involves a dimensionality reduction and, therefore, $\mathbf{C}$ is not a square matrix. Thus, the noisy speech, clean speech, channel and noise cepstral vectors are respectively obtained from the corresponding log-Mel coefficient vectors as

$$\begin{aligned} \mathbf{y}^c &= \mathbf{C}\mathbf{y}; \\ \mathbf{x}^c &= \mathbf{C}\mathbf{x}; \\ \mathbf{h}^c &= \mathbf{C}\mathbf{h}; \\ \mathbf{n}^c &= \mathbf{C}\mathbf{n}. \end{aligned} \tag{2.12}$$

The speech distortion model that has been developed above is the basis from which a great number of noise-robust ASR methods is formulated. Indeed, depending on the type of approach, that distortion model is considered in a particular domain. For example, the model as in (2.5) or (2.10) might typically be taken into account as a starting point when formulating spectral subtraction (SS) enhancement (see Subsection 2.2.1) or vector Taylor series (VTS) feature compensation (see Subsection 2.2.3), respectively.

To conclude this section, let us take a look at how the statistical distribution of the speech energy is altered in the presence of ambient noise by means of the following simple but illustrative simulation. Hereafter, we only consider a particular filterbank channel, l, after the application of the logarithmic compression. For simplicity, it will be assumed that both clean speech and noise follow Gaussian distributions at that channel in the log-Mel power spectral domain. In this context, if we also assume that there is no channel distortion (i.e. $\mathbf{h} = \mathbf{0}$), (2.10) can be written for the l-th filterbank channel as

$$y_l = x_l + \log\left(1 + e^{n_l - x_l}\right) = \log\left(e^{x_l} + e^{n_l}\right). \tag{2.13}$$

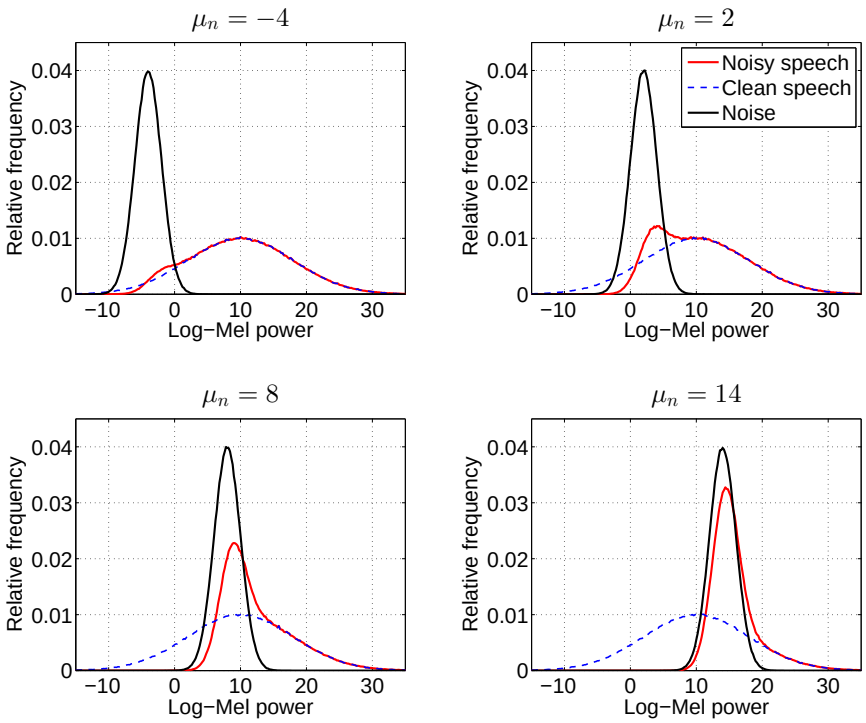

Figure 2.3: Noisy speech, clean speech and noise log-Mel power histograms representing how the statistical distribution of the speech energy is affected in the presence of ambient noise at a particular channel. It is assumed that both clean speech and noise follow Gaussian distributions. The mean and the standard deviation of the clean speech are set to $\mu_x = 8$ and $\sigma_x = 8$, respectively. In the case of the noise, four mean values are considered, $\mu_n = -4, 2, 8, 14$, while the standard deviation is fixed to $\sigma_n = 2$.

In this example we set $p(x_l) = \mathcal{N}(\mu_x = 8, \sigma_x = 8)$, while the standard deviation of the noise density function $p(n_l)$ is fixed to $\sigma_n = 2$ and the noise mean shall be variable. Figure 2.3 shows how the noise affects the statistical distribution of the speech energy when considering different noise mean values, namely $\mu_n = -4, 2, 8, 14$. To create the histograms plotted in this figure, first, clean speech and noise log-Mel power samples were generated by sampling $p(x_l)$ and $p(n_l)$, respectively. Then, such samples were evaluated as in (2.13) to obtain y_l samples and the corresponding noisy speech histograms from them. As can be observed from Figure 2.3, at high signal-to-noise ratios (SNRs), that is, when $\mu_x \gg \mu_n$, the speech distribution remains almost unchanged. As the noise energy increases (i.e. the SNR goes down), the speech Gaussian distribution is right skewed. Finally, if noise energy tends to be comparable to or greater than speech energy (low SNRs), the statistical distribution of the noisy speech tends to be similar to that of the noise, i.e. speech is masked by noise.

2.2 Single-channel robust speech recognition

We have seen that the presence of environmental distortions such as convolutive or additive noise has an impact on the statistical distribution of speech. If we think about an ASR system the acoustic models of which have been trained by means of clean speech data, we are able to state that such distortions might drastically increase the mismatch between the training and testing conditions leading to a severe degradation of the ASR system performance. One immediate approach to face this issue is to rely on multi-condition training, that is, to train the acoustic models of the recognizer by employing distorted speech data similar to those that are expected to be found when using the ASR system in adverse conditions. Unfortunately, this strategy poses a twofold problem: *1)* the acoustic model distribution is largely broadened while *2)* it is really difficult to predict all the adverse conditions in which the ASR system will be used in order to properly train it in advance [107]. Because of these reasons, a number of scalable techniques have been proposed over the past decades to strengthen the ASR systems against distortions in an effective way under a variety of conditions.

While various taxonomies of the noise-robust methods for ASR can be found in the literature [33, 60], below we will present a classification which constitutes a variation of the one reported in [107] in order to concisely discuss only those approaches that are relevant for understanding both our contributions and our experimental evaluation. The categories described in this section are feature-space approaches, model-based approaches, distortion modeling by vector Taylor series (VTS) and missing-data approaches. They are briefly introduced in the following:

- *Feature-space approaches*: This type of methods tries to compensate the speech distortions on the front-end side. Indeed, the feature-space methods do not modify the acoustic model parameters. For example, if a GMM-HMM-based back-end is employed, these methods do not change neither the Gaussian nor the HMM parameters. The same three feature-space approach subcategories as in [107] are considered down below: noise-robust features (which are characterized by a certain degree of insensitivity to distortions), normalization of one or several statistical moments of the features, and feature enhancement. The latter is about suppressing the noise that contaminates the speech features and is the preferred approach in this Thesis mainly due to its versatility and high computational efficiency.

- *Model-based approaches*: Speech distortions can also be compensated on the back-end side or, more precisely, by either estimating or adapting the acoustic model

parameters in a proper way. Two distinct model-based approaches will be discussed later: model adaptation, where the acoustic model parameters are tuned up to explain the speech distortions, and adaptive training, which will be widely used during our experimental evaluation. In contrast to the feature-space approaches, the model-based ones are featured by relatively high computational complexity. Nevertheless, as an advantage, the latter approaches are often more robust against speech distortions than the former ones.

- *Distortion modeling by vector Taylor series*: This approach is formulated from the speech distortion model developed in the previous section in order to compensate the additive and/or the channel noise either on the front-end or the back-end side. That is, this noise-robust strategy may be used for either feature compensation or model adaptation and both schemes will be presented down below in this section. Unlike the classical model adaptation techniques which typically perform linear corrections on the acoustic model parameters, the VTS approach is more accurate and powerful as it employs a physical model that explicitly explains how it is the (non-linear) interaction between the speech signal and the environmental distortions.

- *Missing-data approaches*: This kind of methods is based on handling uncertainty on the feature space. Firstly, spectro-temporal regions of the noisy speech signal which are unreliable because of the presence of distortions are determined. Then, different approaches can be followed, e.g. marginalization or data imputation. While the former ignores those unreliable spectro-temporal regions during the decoding stage (i.e. at the back-end), the latter is based on the reconstruction of such unreliable spectral regions from a statistical point of view [31] (i.e. at the front-end). A critical part of these methods consists of the estimation of missing-data masks to identify the aforementioned unreliable spectro-temporal regions of the noisy speech signal. Particular attention is paid to this issue in the literature since the performance of the missing-data approaches depends heavily on the quality of these masks.

For concision, a number of relevant noise-robust methods for ASR will fall outside of the revision in this section as they will not be taken into account in this doctoral dissertation. Some of the most outstanding of these methods are the stereo data learning-based techniques, which work from learning a mapping function between corrupted and clean speech data during a training phase. One of the most representative methods that falls into this category is SPLICE (Stereo-based Piecewise LInear Compensation for Environments) [34, 36]. Briefly, SPLICE is a minimum mean square error

(MMSE)-based estimator of the clean speech features using a simple non-parametric linear distortion model which is trained from stereo data, i.e. pairs of corrupted and clean speech feature vectors. Moreover, deep learning architectures such as DNNs (Deep Neural Networks) are becoming very popular in order to learn the mapping function between the corrupted and clean speech from stereo data supervised training [201, 202]. In fact, we will account deep learning in Chapter 5 to develop two dual-channel stereo data learning-based methods to estimate missing-data masks and additive noise from dual-channel noisy speech. Nevertheless, a major constraint of this kind of techniques is the difficulty in obtaining stereo data in real-life conditions, conferring these methods a limited interest.

Another category of techniques that will not be discussed in depth is the exemplar-based approaches and, more in particular, non-negative matrix factorization (NMF) for source separation, widely studied over the recent period [11, 58, 197]. They work by modeling the noisy speech features as a linear combination of clean speech and noise exemplars coming from a dictionary. In particular, the dictionary of clean speech exemplars is often defined from the corresponding training dataset of the ASR system. Then, the goal of these techniques is to estimate the set of non-negative linear combination weights that typically minimize the Kullback-Leibler divergence [58] between the noisy observation and the combination of exemplars. Additionally, to represent the noisy speech features by means of a small set of exemplars, a sparsity constraint is normally integrated into the objective function [107]. The calculated weights can be used to estimate the clean speech (and also the noise) features for noise-robust ASR purposes. Notice that these techniques are highly related with the stereo data learning-based methods mentioned above since the former depend on a dictionary of exemplars.

The categories of noise-robust approaches introduced above are revisited with more detail throughout the rest of this section. Ultimately, the selection of a particular noise-robust technique will depend on the requisites and design criteria of our ASR system. For example, as outlined above, if high recognition accuracy is mandatory while a lot of computational and data resources are available, a model-based approach might be chosen. On the contrary, if higher computational efficiency is required while still achieving a good recognition performance, a more flexible option is to deploy a feature-space method [107].

2.2.1 Feature-space approaches

In this subsection we will review those noise-robust methods that compensate the speech distortions solely on the front-end side, i.e. by properly extracting or adapting

the speech features. In turn, the feature-space approaches can be classified into three different subcategories, namely noise-robust features, feature normalization and feature enhancement. In fact, as shall be seen, feature enhancement is the noise-robust scheme mainly followed by our contributions as it simultaneously provides good recognition accuracy in noisy environments and high computational efficiency. Indeed, this same attribute applies to all the feature-space approaches to a greater or lesser extent as, in general, there is a trade-off between recognition accuracy and computational complexity for noise-robust techniques. The aforementioned subcategories are reviewed down below.

2.2.1.1 Noise-robust features

Noise-robust features are characterized by a certain degree of insensitivity to speech distortions. We can further distinguish between two types of them: auditory- and neural network-based features.

Auditory-based features. This kind of features is inspired by the human auditory system to lead to more noise-robust ASR. Despite the large number of auditory-based features, perhaps, the most popular are those derived from the PLP (Perceptually-based Linear Prediction) [74] and RASTA (RelAtive SpecTrAl analysis) [75] processing.

PLP starts by computing the STDFT of the signal by also considering the Hamming window. The result is filtered by means of a filterbank based on the psychoacoustic Bark scale first proposed by E. Zwicker in [212]. Such a filterbank processing is quite similar to that applied for the computation of the MFCCs. In fact, both the perceptual Bark and Mel scales are closely related. The output of the Bark filterbank is pre-emphasized to mimic the equal-loudness characteristic of the human auditory system. Then, the auditory spectrum is approximated by an autoregressive all-pole model [74]. As a consequence, PLP processing fits better with the human hearing system than the traditional linear predictive (LP) analysis.

On the other hand, RASTA features employ an IIR (Infinite Impulse Response) bandpass filter (again inspired by the human auditory system) to remove, per frequency bin, the mean value of the auditory-like spectrum coefficients in a similar way to cepstral mean normalization (CMN) [10], which will be described later. Therefore, RASTA processing is especially appropriate to compensate the effects of channel distortions. This method can be used along with PLP coefficients, resulting in the so-called RASTA-PLP features [75].

Another popular noise-robust auditory-based features are Gammatone frequency cepstral coefficients [164], which are a biologically-inspired variant of MFCCs using

Gammatone filters, and PNCCs (Power-Normalized Cepstral Coefficients) [101]. The latter is a computationally efficient feature extraction approach where Gammatone filters are also used for frequency analysis. A thorough overhaul of the auditory-based feature extraction methods is available in [172]. While the auditory-based features often provide better recognition accuracies than MFCCs, their more difficult generation process makes them of limited interest [107]. Thus, MFCCs is still the most used type of parameterization and, as mentioned above, this will be considered when developing and deploying our contributions.

Neural network-based features. Although DNNs are pervasively used nowadays as replacement of GMMs for acoustic modeling in HMM-based ASR back-ends, various types of ANNs (Artificial Neural Networks) have also been used to generate noise-robust features for GMM-HMM-based ASR systems. First, it must be stated that this is a different kind of approach with respect to the followed by the methods presented above, as ANN-based feature extraction does not exploit any psychoacoustic knowledge. One of the best-known techniques that can be classified into this category is the tandem connectionist feature extraction method of [73] (TANDEM). In TANDEM, an ANN is first discriminatively trained to estimate the posterior probabilities of every subword class given the acoustic observations. Instead of using these probabilities for decoding, TANDEM removes the non-linear activation at the output of the ANN and uses these data, after decorrelating them by PCA (Principal Component Analysis), as features. This method demonstrated its robustness by achieving high error rate reductions on the Aurora-2 [148] noisy continuous digit recognition task [73]. Different variants of TANDEM appeared later in the literature [57]. At this respect, for example, the use of bottleneck neural networks has been explored over the last years [57, 70]. Let us recall that a bottleneck neural network is a feedforward ANN with one of its intermediate hidden layers containing fewer neurons than the rest. This bottleneck layer performs the dimensionality reduction previously tackled by PCA or LDA (Linear Discriminant Analysis) in the conventional tandem methods, leading to better performance [57, 70].

2.2.1.2 Feature normalization

To ensure a higher degree of ASR robustness by reducing the mismatch between the training and test data, the feature normalization methods normalize one or several statistical moments of the speech features. The most well-known feature normalization methods are CMN (Cepstral Mean Normalization) [10] and CMVN (Cepstral Mean and Variance Normalization) [185], which will be used for experimental purposes in this Thesis, and HEQ (Histogram EQualization) [136, 181]. CMN and CMVN normalize the

first and the first two statistical moments, respectively. In addition, HEQ normalizes all the statistical moments of the speech features.

CMN. As introduced, this technique simply normalizes the first statistical moment by subtracting the cepstral mean, usually per utterance, to the cepstral features as follows. Let us assume that a noisy speech utterance is comprised of T frames and its t-th cepstral feature vector is $\mathbf{y}_t^c$ $(t = 0, ..., T-1)$. Then, the cepstral mean $\boldsymbol{\mu}_y$ is computed for this utterance as

$$\boldsymbol{\mu}_y = \frac{1}{T}\sum_{t=0}^{T-1}\mathbf{y}_t^c. \tag{2.14}$$

From (2.14), the resulting cepstral features after the application of CMN, $\bar{\mathbf{y}}_t^c$, are

$$\bar{\mathbf{y}}_t^c = \mathbf{y}_t^c - \boldsymbol{\mu}_y, \quad t = 0, ..., T-1, \tag{2.15}$$

and $\mathrm{E}[\bar{\mathbf{y}}_t^c] \approx \mathbf{0}$, where $\mathrm{E}[\cdot]$ denotes mathematical expectation.

Speech distortion model Eqs. (2.10) and (2.11) in the log-Mel and cepstral domains, respectively, can be rewritten when there is no additive noise, i.e. $\mathbf{n} = \mathbf{n}^c = -\boldsymbol{\infty}$, as

$$\mathbf{y} = \mathbf{x} + \mathbf{h} + \log\left(\mathbf{1} + e^{\mathbf{n}-\mathbf{x}-\mathbf{h}}\right) = \mathbf{x} + \mathbf{h} \tag{2.16}$$

and

$$\mathbf{y}^c = \mathbf{x}^c + \mathbf{h}^c + \mathbf{C}\log\left(\mathbf{1} + e^{\mathbf{C}^{-1}(\mathbf{n}^c-\mathbf{x}^c-\mathbf{h}^c)}\right) = \mathbf{x}^c + \mathbf{h}^c. \tag{2.17}$$

As a result, since the channel effects are additive in both the log-Mel and cepstral domains, it becomes clear that CMN compensates for convolutive (channel) distortions when additive noise is absent. Moreover, it has been shown that even when no channel distortion is present, CMN is able to improve the performance of an ASR system affected by background noise [40]. Besides this, from (2.15) we can guess that this approach makes sense for offline applications or, at least, when a sufficient number of frames is available. Hence, for real-time applications, an alternative to CMN could consist of the use of the IIR bandpass filter of RASTA in accordance with the discussion above.

CMVN. CMVN not only normalizes the first statistical moment of the speech features, but also the second one. This is accomplished by also normalizing the set of cepstral feature vectors in (2.15), $\{\bar{\mathbf{y}}_t^c;\ t = 0, ..., T-1\}$, by its standard deviation

$$\boldsymbol{\sigma}_y = \sqrt{\frac{1}{T-1}\sum_{t=0}^{T-1}(\mathbf{y}_t^c - \boldsymbol{\mu}_y)^2}, \tag{2.18}$$

where both the square root $\sqrt{\cdot}$ and power $(\cdot)^2$ operators are applied element-wise. Therefore, the result of the application of CMVN is

$$\tilde{\mathbf{y}}_t^c = \frac{\bar{\mathbf{y}}_t^c}{\boldsymbol{\sigma}_y} = \frac{\mathbf{y}_t^c - \boldsymbol{\mu}_y}{\boldsymbol{\sigma}_y}, \quad t = 0, ..., T-1, \tag{2.19}$$

where the division operator $\div$ is also applied element-wise, $\mathrm{E}[\tilde{\mathbf{y}}_t^c] \approx \mathbf{0}$ and the variance of every component of $\tilde{\mathbf{y}}_t^c$ is approximately 1.

While CMVN additionally decreases the second-order moment mismatch between the training and test data, we cannot draw a parallel between this fact and the compensation of a particular type of distortion as in the case of CMN. Nevertheless, it is evident that such a statistical normalization affects both the convolutive and additive distortions. Indeed, CMVN has been proven to be superior to CMN when used as normalization technique [107].

HEQ. Finally, this method, which can be applied in either the log-Mel or cepstral domains, works by performing histogram equalization on the test speech features to normalize all the statistical moments [136]. The core idea behind this approach is to transform the statistical distribution of the test data in order to be similar to that of the training data, thereby reducing the mismatch between them. For instance, if we assume that y_c corresponds to a component of the cepstral feature vector $\mathbf{y}^c$, HEQ is applied element-wise as

$$f(y^c) = C_x^{-1}\left(C_y(y^c)\right), \tag{2.20}$$

where $C_y(\cdot)$ is the cumulative distribution function (CDF) of the test data and $C_x^{-1}(\cdot)$ is the inverse CDF of the training data. In practice, both CDFs are approximated from the cumulative histograms correspondingly calculated from the available training and test speech data. The equalization principle reflected by (2.20) has been considered over time to develop a variety of feature normalization methods achieving different improvements, e.g. [39, 78].

2.2.1.3 Feature enhancement

In this subcategory, within the feature-space approaches, we include those methods that try to improve the recognition performance by removing the distortions affecting the speech signal. To do so, this type of methods typically handles the speech distortion model developed in Section 2.1 from a statistical point of view. As aforementioned, feature enhancement is the preferred approach in this Thesis (see Chapters 3 and 4) mainly due to both its versatility and high computational efficiency. Hence, since this is a very broad topic, we only focus here on those approaches that serve as a basis of our

contributions of Chapter 3: spectral subtraction (SS) [22] and Wiener filtering [114]. Moreover, despite vector Taylor series (VTS) feature enhancement/compensation is considered in Chapter 4, its fundamentals are discussed in a later subsection as VTS can also be applied to model adaptation.

Spectral subtraction is a straightforward method that mitigates the additive noise by subtracting, in the frequency domain, an estimate of the noise to the noisy spectrum. This method assumes that the speech and noise signals are uncorrelated. This way, by also assuming that there is no channel distortion, i.e. $|H(f,t)|^2 = 1$, Eq. (2.5) can be simplified as

$$|Y(f,t)|^2 = |X(f,t)|^2 + |N(f,t)|^2. \tag{2.21}$$

Then, a noise estimate, $|\hat{N}(f,t)|^2$, is required. This one may be computed in a simple way by averaging those frames of the spectrum where speech is absent, or by using one of the many noise estimation algorithms available in the literature and briefly reviewed in Section 2.3. Once we have $|\hat{N}(f,t)|^2$, SS estimates every clean speech power spectrum bin, $|\hat{X}(f,t)|^2$, as

$$|\hat{X}(f,t)|^2 = |Y(f,t)|^2 - |\hat{N}(f,t)|^2. \tag{2.22}$$

SS as defined in (2.22) poses a hurdle since it permits negative power spectrum bins. In practice, an additional function, typically the max operator, is employed to avoid that:

$$|\hat{X}(f,t)|^2 = \max\left(|Y(f,t)|^2 - |\hat{N}(f,t)|^2, \eta|Y(f,t)|^2\right), \tag{2.23}$$

where $\eta \in (0,\ 1)$ is a thresholding factor which establishes a lower bound $\eta|Y(f,t)|^2$.

SS can alternatively be rewritten as a linear filtering problem, that is,

$$|\hat{X}(f,t)| = G_{SS}(f,t)|Y(f,t)|, \tag{2.24}$$

where $G_{SS}(f,t)$ would correspond to the SS filter transfer (gain) function, the definition of which is, taking into account the integration of the max operator as in (2.23),

$$G_{SS}(f,t) = \sqrt{\max\left(\frac{\hat{\xi}(f,t)}{\hat{\xi}(f,t)+1}, \eta\right)}, \tag{2.25}$$

where $\hat{\xi}(f,t) = \left(|Y(f,t)|^2 - |\hat{N}(f,t)|^2\right)/|\hat{N}(f,t)|^2$ is an approximation of the instantaneous *a priori* SNR. Thus, from (2.25), SS might be construed as an SNR-dependent attenuator. Therefore, the higher (lower) the SNR, the less (greater) the attenuation

applied to the noisy observation. In particular, when the SNR tends to infinite (zero) $G_{SS}(f,t) \to 1$ ($G_{SS}(f,t) \to 0$), as speech (noise) dominates in that spectral bin.

The following limitations concerning this method can be highlighted:

- The hard thresholding required to avoid negative power spectrum bin estimates can severely affect the performance of the ASR system.
- No knowledge about the speech statistics is considered, in such a way that the performance of SS heavily relies on the accuracy of the noise estimation. In fact, the performance of this method might be seriously degraded in the presence of non-stationary noise.
- Concerning the above items, these facts tend to favor the appearance of "musical noise" [18], which distorts the speech signal and consists of annoying (i.e. audible) short bursts of noise [144, 183].

It has been proven that oversubtraction of the noise estimate helps to mitigate the "musical noise" effect [149], thereby leading to better speech recognition performance [18, 100]. If $\beta > 1$ is the oversubtraction factor, (2.23) can be rewritten by taking into consideration this improvement as

$$|\hat{X}(f,t)|^2 = \max\left(|Y(f,t)|^2 - \beta|\hat{N}(f,t)|^2, \eta|Y(f,t)|^2\right). \tag{2.26}$$

Finally, it must be remarked that SS can be alternatively formulated in the magnitude spectral domain [149, 183].

Wiener filter. Feature enhancement is among the most important applications of this well-known type of linear filter, which is closely related to SS as we will see in the following. With this aim, we first establish that the clean speech signal can be estimated in the time domain from the noisy speech signal by filtering as

$$\hat{x}(m) = h(m) * y(m) = \sum_{k=0}^{p-1} h(k)y(m-k), \tag{2.27}$$

where $h(m)$ is a $(p-1)$-order FIR (Finite Impulse Response) filter. Then, the Wiener filter (WF) approach seeks for the linear filter $h(m)$ that minimizes the mean square error (MSE) between the estimate $\hat{x}(m)$ and the actual clean speech signal $x(m)$. It is straightforward to show that such a strategy leads to the following filter in the frequency domain:

$$H(f,t) = \frac{\mathcal{S}_{xy}(f,t)}{\mathcal{S}_y(f,t)}, \tag{2.28}$$

where $\mathcal{S}_{xy}(f,t)$ corresponds to the cross-PSD (Power Spectral Density) between clean and noisy speech while $\mathcal{S}_y(f,t)$ refers to the noisy speech PSD. By assuming again that the speech and noise signals are uncorrelated as well as there is no channel distortion, (2.28) can be expressed as [114]

$$H(f,t) = \frac{\mathcal{S}_x(f,t)}{\mathcal{S}_x(f,t) + \mathcal{S}_n(f,t)}, \tag{2.29}$$

where, similarly, $\mathcal{S}_x(f,t)$ and $\mathcal{S}_n(f,t)$ are the PSDs of clean speech and noise, respectively.

If we now define the *a priori* SNR in terms of PSDs as $\xi(f,t) = \mathcal{S}_x(f,t)/\mathcal{S}_n(f,t)$, the WF in (2.29) can be rewritten as follows:

$$H(f,t) = \frac{\xi(f,t)}{\xi(f,t)+1}. \tag{2.30}$$

From the WF definition in (2.30) and the gain function $G_{SS}(f,t)$ in (2.25) we can remark the similarities between this method and SS. Thus, $H(f,t)$ can again be understood as an SNR-dependent attenuator and the same discussion as for the SS case is valid here.

Advanced front-end. This feature enhancement method was standardized and released by the STQ ETSI (Speech and multimedia Transmission Quality, European Telecommunications Standards Institute) Aurora working group in 2002 with reference ETSI ES 202 050 [3]. Since then, it has been considered a quite relevant method among the state-of-the-art techniques as well as a reference for experimental comparison purposes. Indeed, the ETSI advanced front-end (AFE) will be evaluated in Chapter 6 to this end. Briefly, the core of AFE consists of a two stage Mel-warped Wiener filter [9] to reduce the effect of the environmental noise. Additionally, AFE also integrates SNR-dependent waveform processing and blind equalization to compensate the possible convolutive (channel) distortion [3, 149]. This blind equalization approach is a good candidate to be used instead of CMN if online channel compensation is required. It has been shown in the literature that AFE is able to achieve high error rate reductions on the Aurora-2 [148] noisy continuous digit recognition task [126].

2.2.2 Model-based approaches

The model-based noise-robust approaches mitigate the mismatch between the training and testing conditions at the back-end side by properly estimating or adapting the acoustic model parameters of the recognizer. As also aforementioned, this kind of

approaches offers a high level of robustness against noise and other sources of variability at the expense of relatively high computational complexity [107]. In turn, these model-based approaches can be classified into two different categories, namely model adaptation and adaptive training, which are shortly revised down below.

2.2.2.1 Model adaptation

Model adaptation is about transforming the acoustic model parameters in order to account for the testing acoustic conditions during recognition. Therefore, every source of acoustic mismatch can be compensated, e.g. ambient noise, channel distortion, inter-speaker variability, etc. MAP (Maximum A Posteriori) adaptation [56] is among the most popular model adaptation techniques and consists of the calculation of those acoustic model parameters $\hat{\Lambda}$ that correspond to the mode of the posterior distribution $p(\Lambda|\mathbf{Y}, \mathcal{H})$, that is,

$$\hat{\Lambda} = \underset{\Lambda}{\operatorname{argmax}}\, p(\Lambda|\mathbf{Y}, \mathcal{H}) = \underset{\Lambda}{\operatorname{argmax}}\, p(\mathbf{Y}|\Lambda, \mathcal{H})p(\Lambda), \tag{2.31}$$

where $p(\Lambda)$ is the prior distribution of the model parameters Λ, and $\mathbf{Y}$ and $\mathcal{H}$ refer to the adaptation data and the corresponding transcriptions, respectively. To find the optimal set of parameters $\hat{\Lambda}$, (2.31) is iteratively solved by means of the expectation-maximization (EM) algorithm.

A model adaptation technique is said to be supervised if the transcriptions of the adaptation utterances are available to guide the process. On the other hand, when such an information is not available, the model adaptation method is unsupervised. For the latter, a two-pass decoding strategy is followed. First, a hypothesis or transcription $\mathcal{H}$ is generated by decoding the adaptation utterance using the initial model Λ. Then, that hypothesis is (often) considered good enough to estimate the new model $\hat{\Lambda}$ by means of adaptation.

The performance of MAP adaptation can be severely affected when carried over few data available, since this scheme only modifies the model parameters of the acoustic units that are observed in the adaptation dataset. Despite there is a number of MAP-based variants in the literature that try to alleviate this problem, e.g. [165, 167, 168], it might be preferable to follow an MLE (Maximum Likelihood Estimation) scheme [55, 104] under such conditions. In other words, the MLE-based adaptation techniques produce more accurate adapted models than the MAP-based ones when the amount of data for adaptation is limited [66].

The most popular model adaptation methods, such as MLLR (Maximum Likelihood Linear Regression) [104], take into account this MLE criterion. MLLR proposes to

adapt the mean vectors of the acoustic model Gaussian PDFs by means of a class-dependent affine transformation:

$$\boldsymbol{\mu}_y^{(k)} = \mathbf{A}_r \boldsymbol{\mu}_x^{(k)} + \mathbf{b}_r, \tag{2.32}$$

where the mean vector of the new model $\boldsymbol{\mu}_y^{(k)}$ is obtained from rectifying the mean vector $\boldsymbol{\mu}_x^{(k)}$ corresponding to the k-th Gaussian component of the initial model. Furthermore, $\mathbf{A}_r$ and $\mathbf{b}_r$ are the parameters (matrix and vector) of the affine transformation (to be estimated) belonging to the r-th regression class. The MLE of these parameters is again carried out by application of the EM algorithm relying on the auxiliary Q-function

$$Q = \sum_t \sum_k \gamma_t^{(k)} \log p_{\hat{\Lambda}}(\mathbf{y}_t^c | k), \tag{2.33}$$

where $\gamma_t^{(k)}$ is the posterior probability for the k-th Gaussian component at time t. By setting derivatives of Q with respect to $\mathbf{A}_r$ and $\mathbf{b}_r$ to zero, the optimal values of the transformation parameters are finally found.

On the other hand, since various acoustic factors such as the environmental distortions have an impact on the signal variance, we should also modify the covariance matrices of the acoustic model Gaussians for higher robustness. Their adaptation can be accomplished as follows [55]:

$$\boldsymbol{\Sigma}_y^{(k)} = \mathbf{H}_r \boldsymbol{\Sigma}_x^{(k)} \mathbf{H}_r^\top, \tag{2.34}$$

where $\boldsymbol{\Sigma}_y^{(k)}$ is the covariance matrix of the new model while the covariance matrix $\boldsymbol{\Sigma}_x^{(k)}$ corresponds to the k-th Gaussian component of the initial model. $\mathbf{H}_r$ is the transformation matrix belonging to the r-th regression class. In this context, the mean correction is first estimated as aforementioned given the initial covariance matrix $\boldsymbol{\Sigma}_x^{(k)}$. Then, given the new mean vector $\boldsymbol{\mu}_y^{(k)}$, $\mathbf{H}_r$ is computed, and $\boldsymbol{\Sigma}_y^{(k)}$ from (2.34). The described process is repeated as many times as needed until convergence.

In [55], the so-called CMLLR (Constrained MLLR) method was proposed, which is a variant of MLLR adaptation where the mean vector and covariance matrix transformation matrices are constrained to be equal, namely

$$\begin{aligned} \boldsymbol{\mu}_y^{(k)} &= \mathbf{A}_r \boldsymbol{\mu}_x^{(k)} - \mathbf{b}_r, \\ \boldsymbol{\Sigma}_y^{(k)} &= \mathbf{A}_r \boldsymbol{\Sigma}_x^{(k)} \mathbf{A}_r^\top. \end{aligned} \tag{2.35}$$

A major advantage of CMLLR is the reduced number of parameters to be estimated along with substantially lower computational complexity with respect to the standard

MLLR adaptation. Furthermore, CMLLR can be alternatively formulated and implemented in a more efficient manner in the feature space. In this case, CMLLR is referred to as fMLLR (feature-space MLLR) and the feature vector $\mathbf{y}^c$ is modified as

$$\mathbf{x}_r^c = \mathbf{A}_r^{-1}\mathbf{y}^c + \mathbf{A}_r^{-1}\mathbf{b}_r. \tag{2.36}$$

During our experimental evaluation in Chapter 6, fMLLR will be employed.

In the next subsection, a more powerful model adaptation approach in comparison with the above techniques, and based on distortion modeling by vector Taylor series (VTS), will be explained. Unlike the above model adaptation techniques which perform linear corrections on the acoustic model parameters, VTS considers a physical model that exploits the non-linear interactions between the speech signal and the environmental distortions [61]. PMC (Parallel Model Combination) [54] is another well-known method for adapting the acoustic model parameters which also uses an explicit speech distortion model in a similar way to VTS, though the latter is still more accurate [6].

2.2.2.2 Adaptive training

Many of the noise-robust ASR algorithms assume that the recognizer has been trained by means of clean speech data. Nevertheless, training the ASR system with only clean data is not always possible, as it may be difficult to collect enough amount of speech data under clean acoustic conditions. Therefore, if the acoustic models of the recognizer are trained from a mixture of speech data collected in different acoustic (noisy) conditions, those feature enhancement methods might fail. The adaptive training strategy helps to overcome this hurdle by applying the same processing at the training and testing phases in order to consistently compensate the same mismatch sources.

Most of these adaptive techniques are based on noise adaptive training (NAT) [34]. The simplest NAT approach is called fNAT (feature-space NAT) and it can be understood as an obvious extension of the multi-condition training. This popular as well as quite robust approach simply consists of training the ASR system with noisy speech data previously compensated by using the same feature enhancement method as during the testing phase. This way, the same sources of mismatch are consistently removed at both the training and testing stages to provide great recognition accuracy in a very straightforward manner. This scheme is also known as multi-style training and the resulting acoustic models are referred to as multi-style acoustic models. For the reasons given above, fNAT will be widely employed in this Thesis during the experimental evaluation.

We can distinguish between two different subcategories within the adaptive training methods, namely the feature- and the model-space joint model training methods. While fNAT belongs to the former, within the latter subcategory we can find some techniques like speaker adaptive training (SAT) [7], and mNAT (model-space NAT) methods such as joint adaptive training (JAT) [113] or irrelevant variability normalization (IVN) [92]. This type of techniques jointly train a canonical acoustic model and a set of transforms typically under an MLE criterion [107]. On the one hand, the canonical acoustic model is a compact HMM model which captures the desired speech variability regardless the acoustic condition (i.e. ambient noise, speaker, etc.). On the other hand, each transform maps the canonical model to another one adapted to the particular acoustic condition that such a transform represents. As an example, let us briefly review how SAT works, as it will also be employed during our experimental evaluation.

SAT is a technique proposed to mitigate the impact of the inter-speaker variability on the recognition performance by adapting the acoustic models to each particular speaker. This method emerged as a better alternative to MLLR and CMLLR for the mentioned purpose. At the training stage, SAT jointly computes a speaker-independent (canonical) acoustic model Λ from multi-speaker training data and a set of transforms by following an MLE criterion. Let us suppose that we have a multi-speaker training dataset obtained from a total of S different speakers. SAT estimates that set of transforms $\boldsymbol{\mathcal{T}} = \left(\boldsymbol{\mathcal{T}}^{(1)}, \boldsymbol{\mathcal{T}}^{(2)}, ..., \boldsymbol{\mathcal{T}}^{(S)}\right)$ (one per speaker) by maximizing the likelihood of the training data as

$$\left(\hat{\Lambda}, \hat{\boldsymbol{\mathcal{T}}}\right) = \underset{\Lambda, \boldsymbol{\mathcal{T}}}{\operatorname{argmax}} \prod_{s=1}^{S} p\left(\mathbf{Y}^{(s)} \left| \mathcal{H}^{(s)}, \Lambda, \boldsymbol{\mathcal{T}}^{(s)} \right.\right), \tag{2.37}$$

where $\mathbf{Y}^{(s)}$ represents the training data coming from the s-th speaker, being $\mathcal{H}^{(s)}$ the corresponding transcriptions. Thus, the s-th transform derived from (2.37) maps the model $\hat{\Lambda}$ to another one adapted to the s-th training speaker. The same speaker-dependent adaptation would be applied during the test phase by properly estimating transforms for the test speakers.

Again, the optimization problem in (2.37) is solved by application of the EM algorithm. First, the transforms are estimated given an initial compact model. Second, given the new transforms, the canonical acoustic model parameters are updated. The described process is repeated as many times as needed until convergence.

2.2.3 Distortion modeling by vector Taylor series

The vector Taylor series (VTS) approach [137, 138] is a powerful strategy which exhibits an outstanding robustness since it relies on a physical distortion model that

explicitly explains how it is the non-linear interaction between the speech signal and the environmental distortions in the log-Mel or cepstral domain. In particular, VTS is used to linearize such a speech distortion model in order to find analytically tractable, closed-form expressions for either model adaptation or feature compensation. Because VTS model adaptation is still more accurate than VTS feature compensation at the expense of higher computational cost (all the acoustic model parameters need to be updated every time the testing acoustic conditions change) [110], the trade-off between these two variables will determine which scheme (i.e. adaptation or compensation) is more suitable to be deployed [107].

To begin with, let us rewrite the non-linear speech distortion model in the cepstral domain of (2.11) as follows:

$$\mathbf{y} = \mathbf{x} + \mathbf{h} + \underbrace{\mathbf{C}\log\left(\mathbf{1} + e^{\mathbf{C}^{-1}(\mathbf{n}-\mathbf{x}-\mathbf{h})}\right)}_{\mathbf{g}(\mathbf{x},\mathbf{h},\mathbf{n})}, \tag{2.38}$$

where $\mathbf{g}(\mathbf{x},\mathbf{h},\mathbf{n})$ is the so-called mismatch function and the superscript c denoting cepstral domain has been omitted for the sake of clarity. This model can be approximated linearly by computing its first-order VTS expansion around the point $\left(\boldsymbol{\mu}_x^{(k)}, \boldsymbol{\mu}_h, \boldsymbol{\mu}_n\right)$, which results

$$\begin{aligned}\mathbf{y} \;\approx\; & \boldsymbol{\mu}_x^{(k)} + \boldsymbol{\mu}_h + \mathbf{g}\left(\boldsymbol{\mu}_x^{(k)}, \boldsymbol{\mu}_h, \boldsymbol{\mu}_n\right) \\ & + \mathbf{J}_x^{(k)}\left(\mathbf{x} - \boldsymbol{\mu}_x^{(k)}\right) + \mathbf{J}_h^{(k)}\left(\mathbf{h} - \boldsymbol{\mu}_h\right) + \mathbf{J}_n^{(k)}\left(\mathbf{n} - \boldsymbol{\mu}_n\right).\end{aligned} \tag{2.39}$$

The choice of the expansion point $\left(\boldsymbol{\mu}_x^{(k)}, \boldsymbol{\mu}_h, \boldsymbol{\mu}_n\right)$ in particular will be justified later, while we can anticipate that $\boldsymbol{\mu}_x^{(k)}$, $\boldsymbol{\mu}_h$ and $\boldsymbol{\mu}_n$ are cepstral mean vectors of clean speech, convolutive and additive noises, respectively. The corresponding Jacobian matrices, namely $\mathbf{J}_x^{(k)}$, $\mathbf{J}_h^{(k)}$ and $\mathbf{J}_n^{(k)}$, are respectively defined as

$$\begin{aligned}\mathbf{J}_x^{(k)} &= \left.\frac{\partial \mathbf{y}}{\partial \mathbf{x}}\right|_{\boldsymbol{\mu}_x^{(k)},\boldsymbol{\mu}_h,\boldsymbol{\mu}_n} = \mathbf{G}^{(k)}; \\ \mathbf{J}_h^{(k)} &= \left.\frac{\partial \mathbf{y}}{\partial \mathbf{h}}\right|_{\boldsymbol{\mu}_x^{(k)},\boldsymbol{\mu}_h,\boldsymbol{\mu}_n} = \mathbf{G}^{(k)}; \\ \mathbf{J}_n^{(k)} &= \left.\frac{\partial \mathbf{y}}{\partial \mathbf{n}}\right|_{\boldsymbol{\mu}_x^{(k)},\boldsymbol{\mu}_h,\boldsymbol{\mu}_n} = \mathbf{I} - \mathbf{G}^{(k)} = \mathbf{F}^{(k)},\end{aligned} \tag{2.40}$$

where $\mathbf{I}$ is the identity matrix and

$$\mathbf{G}^{(k)} = \mathbf{C}\,\mathrm{diag}\left(\frac{\mathbf{1}}{\mathbf{1} + e^{\mathbf{C}^{-1}\left(\boldsymbol{\mu}_n - \boldsymbol{\mu}_x^{(k)} - \boldsymbol{\mu}_h\right)}}\right)\mathbf{C}^{-1}, \tag{2.41}$$

in which diag$(\cdot)$ indicates a diagonal matrix the main diagonal of which corresponds to the argument, and division $\div$ is applied element-wise.

Now, the linear speech distortion model of (2.39) is easier to manipulate analytically than that of (2.38) in order to perform either HMM adaptation or feature compensation. Both types of VTS-based approaches are explained immediately below. Finally, it should be noticed that the VTS approach can be applied in different domains, and not only in the cepstral one. In fact, the most usual alternative domain of application is the log-Mel one [62] and, in Chapter 4, we will extend the VTS feature compensation to a dual-channel mobile device framework in this domain.

2.2.3.1 VTS model adaptation

From now on, let us assume a GMM-based acoustic model comprised of $\mathcal{K}$ Gaussian components and described by the set of parameters $\left\{\boldsymbol{\mu}_x^{(k)}, \boldsymbol{\Sigma}_x^{(k)};\ k = 1, 2, ..., \mathcal{K}\right\}$ trained in clean acoustic conditions. Thus, the goal of VTS model adaptation is to compute a new set of acoustic model Gaussian parameters, i.e. $\left\{\boldsymbol{\mu}_y^{(k)}, \boldsymbol{\Sigma}_y^{(k)};\ k = 1, 2, ..., \mathcal{K}\right\}$, to adapt the acoustic model to some particular adverse acoustic conditions. From the linearized model of (2.39), this is easily accomplished since a linear combination of Gaussian variables follows another Gaussian distribution [150] and it can also be assumed that both $\mathbf{h}$ and $\mathbf{n}$ follow Gaussian distributions. In this way, the expansion around the set of mean vectors $\left(\boldsymbol{\mu}_x^{(k)}, \boldsymbol{\mu}_h, \boldsymbol{\mu}_n\right)$ is justified.

Then, it is easy to show that the parameters of each adapted Gaussian component $p(\mathbf{y}|k) = \mathcal{N}\left(\mathbf{y}\left|\boldsymbol{\mu}_y^{(k)}, \boldsymbol{\Sigma}_y^{(k)}\right.\right)$ can be approximated from (2.39) as

$$\boldsymbol{\mu}_y^{(k)} \approx \boldsymbol{\mu}_x^{(k)} + \boldsymbol{\mu}_h + \mathbf{g}\left(\boldsymbol{\mu}_x^{(k)}, \boldsymbol{\mu}_h, \boldsymbol{\mu}_n\right) \tag{2.42}$$

and

$$\boldsymbol{\Sigma}_y^{(k)} \approx \mathbf{G}^{(k)}\boldsymbol{\Sigma}_x^{(k)}{\mathbf{G}^{(k)}}^\top + \mathbf{F}^{(k)}\boldsymbol{\Sigma}_n{\mathbf{F}^{(k)}}^\top, \tag{2.43}$$

where the VTS model adaptation approach typically assumes that the channel term $\mathbf{h}$ is constant over time. Furthermore, (2.43) is often diagonalized for efficiency purposes, which is justified when VTS is developed in the cepstral domain, where features are more uncorrelated.

Apart from the above static parameters, the model velocity (Δ) and acceleration (Δ^2) dynamic parameters can be adapted as well by using a continuous-time approximation [54]. Thus, the mean vectors and covariance matrices of the dynamic parame-

ters can be approximated in the following way [6]:

$$\begin{aligned}
\boldsymbol{\mu}_{\Delta y}^{(k)} &\approx \mathbf{G}^{(k)}\boldsymbol{\mu}_{\Delta x}^{(k)} + \mathbf{F}^{(k)}\boldsymbol{\mu}_{\Delta n}, \\
\boldsymbol{\mu}_{\Delta^2 y}^{(k)} &\approx \mathbf{G}^{(k)}\boldsymbol{\mu}_{\Delta^2 x}^{(k)} + \mathbf{F}^{(k)}\boldsymbol{\mu}_{\Delta^2 n}, \\
\boldsymbol{\Sigma}_{\Delta y}^{(k)} &\approx \mathbf{G}^{(k)}\boldsymbol{\Sigma}_{\Delta x}^{(k)}{\mathbf{G}^{(k)}}^{\top} + \mathbf{F}^{(k)}\boldsymbol{\Sigma}_{\Delta n}{\mathbf{F}^{(k)}}^{\top}, \\
\boldsymbol{\Sigma}_{\Delta^2 y}^{(k)} &\approx \mathbf{G}^{(k)}\boldsymbol{\Sigma}_{\Delta^2 x}^{(k)}{\mathbf{G}^{(k)}}^{\top} + \mathbf{F}^{(k)}\boldsymbol{\Sigma}_{\Delta^2 n}{\mathbf{F}^{(k)}}^{\top}.
\end{aligned} \tag{2.44}$$

Again for the same reason as for (2.43), $\boldsymbol{\Sigma}_{\Delta y}^{(k)}$ and $\boldsymbol{\Sigma}_{\Delta^2 y}^{(k)}$ are typically made diagonal.

As we can observe, to apply VTS model adaptation we only need to estimate a few convolutive and additive noise statistical parameters. There are plenty of techniques to do this, being very popular those methods based on the EM algorithm [61, 108].

Several improvements to the above VTS-based model adaptation strategy have been published in the literature over the last two decades. For instance, in [24] it is explored (with success) the use of second-order VTS, which significantly outperforms the first-order version in terms of recognition accuracy on the Aurora-4 [146] noisy medium-vocabulary task. In addition, when developing the speech distortion model in Section 2.1, we simplified the expression in (2.4) by neglecting the term $\alpha_{f,t}$ modeling the relative phase between the speech and the noise. Nevertheless, in [109] it is shown that formulating VTS model adaptation from a phase-sensitive speech distortion model contributes to achieve higher recognition accuracy.

2.2.3.2 VTS feature compensation

Alternatively, we can follow a parallel scheme to VTS model adaptation but on the feature space. In this case, a $\mathcal{K}$-component GMM $p(\mathbf{x})$ is employed to model the clean speech features on the front-end side as

$$p(\mathbf{x}) = \sum_{k=1}^{\mathcal{K}} P(k)\mathcal{N}\left(\mathbf{x}\left|\boldsymbol{\mu}_x^{(k)}, \boldsymbol{\Sigma}_x^{(k)}\right.\right), \tag{2.45}$$

where $P(k)$ is the prior probability of the k-th Gaussian component. Then, assuming that the acoustic models of the recognizer are trained with clean speech data, the VTS approach can be embedded in an MMSE estimator for the clean speech features. In general, the MMSE estimation of the clean speech feature vector $\mathbf{x}$ given the noisy observation $\mathbf{y}$ is defined as

$$\hat{\mathbf{x}} = \mathrm{E}\left[\mathbf{x}|\mathbf{y}\right] = \int \mathbf{x}p(\mathbf{x}|\mathbf{y})d\mathbf{x}. \tag{2.46}$$

By taking into account both the $\mathcal{K}$-component GMM-based modeling of the clean speech features in (2.45) and the speech distortion model of (2.38), the MMSE estimation of (2.46) can be rewritten as follows:

$$\hat{\mathbf{x}} = \mathbf{y} - \mathbf{h} - \sum_{k=1}^{\mathcal{K}} P(k|\mathbf{y}) \int \mathbf{C} \log\left(\mathbf{1} + e^{\mathbf{C}^{-1}(\mathbf{n}-\mathbf{x}-\mathbf{h})}\right) p(\mathbf{x}|\mathbf{y}, k) d\mathbf{x}, \tag{2.47}$$

where $P(k|\mathbf{y})$ is the k-th posterior probability, which can be expressed in the following way by using the Bayes' theorem:

$$P(k|\mathbf{y}) = \frac{P(k)\mathcal{N}\left(\mathbf{y}\left|\boldsymbol{\mu}_y^{(k)}, \boldsymbol{\Sigma}_y^{(k)}\right.\right)}{\sum_{k'=1}^{\mathcal{K}} P(k')\mathcal{N}\left(\mathbf{y}\left|\boldsymbol{\mu}_y^{(k')}, \boldsymbol{\Sigma}_y^{(k')}\right.\right)}. \tag{2.48}$$

It must be noticed that the posterior $P(k|\mathbf{y})$ can be easily computed given the clean speech model of (2.45) from just evaluating Eqs. (2.42) and (2.43) to obtain values for the Gaussian parameters.

The MMSE estimation of (2.47) is still difficult to evaluate, so the integration of VTS allows us to approximate it and to make it analytically tractable. Thus, by considering a zero-order VTS approach, the MMSE estimation of (2.47) can be approximated by [137]

$$\hat{\mathbf{x}} = \mathbf{y} - \mathbf{h} - \sum_{k=1}^{\mathcal{K}} P(k|\mathbf{y}) \mathbf{C} \log\left(\mathbf{1} + e^{\mathbf{C}^{-1}\left(\boldsymbol{\mu}_n - \boldsymbol{\mu}_x^{(k)} - \boldsymbol{\mu}_h\right)}\right). \tag{2.49}$$

Similarly, in [176] it is proposed the use of a first-order VTS to approximate the MMSE estimation of the clean speech features by also assuming that the joint probability distribution for $\mathbf{x}$ and $\mathbf{y}$ is Gaussian, that is,

$$\hat{\mathbf{x}} = \sum_{k=1}^{\mathcal{K}} P(k|\mathbf{y}) \left(\boldsymbol{\mu}_x^{(k)} + \boldsymbol{\Sigma}_x^{(k)} \mathbf{G}^{(k)^\top} \boldsymbol{\Sigma}_y^{(k)^{-1}} \left(\mathbf{y} - \boldsymbol{\mu}_y^{(k)}\right)\right). \tag{2.50}$$

As for VTS model adaptation, different improvements have been reported in the literature to the above VTS-based feature compensation scheme over the last years. Similarly to the model adaptation case, in [175], significant benefits in terms of recognition accuracy are achieved by using a second-order VTS expansion of a phase-sensitive speech distortion model. It is well worth highlighting at this point another well-known feature compensation method called ALGONQUIN [51], which can be considered as an alternative to a phase-sensitive VTS approach since ALGONQUIN inherently models the phase term.

2.2.4 Missing-data approaches

The missing-data approaches take advantage of the spectro-temporal redundancy characteristic of the speech signal to achieve good ASR performance in the presence of noise. In other words, this type of methods states that it is not necessary to have the entire speech spectrogram for successful recognition. Therefore, these methods need first to identify which regions of the noisy speech spectrogram are unreliable due to the pervasiveness of noise (i.e. the set of time-frequency spectral bins where noise dominates). Once done, one of two missing-data paradigms can be followed, namely the modification of the recognizer to ignore those unreliable elements during recognition or data imputation. The former paradigm is based on the recognition of incomplete speech spectra as an alternative to model adaptation. The core idea is that the way the observation probabilities is computed at the decoding stage accounts for the reliable and unreliable parts of the spectrogram. Two techniques that can be classified within this category are marginalization [31] and SFD (Speech Fragment Decoding) [13]. In particular, one extreme and widely used kind of marginalization is that where only the reliable elements are used during recognition [107]. On the other hand, data imputation (also known as spectral reconstruction) employs reliable spectro-temporal regions in order to estimate values from a statistical point of view for the unreliable parts of the noisy spectrogram [65, 155].

One of these data imputation algorithms is the TGI (Truncated-Gaussian based Imputation) technique of [65], which will be used for comparison purposes in this Thesis. TGI is based on the well-known *log-max* model [158], which is a simplified version of the speech distortion model in the log-Mel domain of (2.10):

$$\mathbf{y} \approx \max(\mathbf{x}, \mathbf{n}). \tag{2.51}$$

It can be shown that the *log-max* model is quite accurate despite its simplicity [66]. This model states that if the speech energy is greater than that of the noise, speech dominates the scene masking the noise, and vice versa. Thus, $\mathbf{y}$ is an upper bound for the masked clean speech energy, i.e. $\mathbf{x} \in (-\boldsymbol{\infty}, \mathbf{y}]$. This fact is exploited by the TGI method to achieve accurate clean speech feature estimates. The algorithm operates on a frame-by-frame basis. At every time frame t, the noisy observation is segregated into reliable and unreliable components, i.e. $\mathbf{y} = \{\mathbf{y}_r, \mathbf{y}_u\}$. Clean speech estimates for reliable elements are the observations themselves, namely $\hat{\mathbf{x}}_r = \mathbf{y}_r$, while unreliable elements are estimated using MMSE estimation. Taking into account that clean speech is again modeled by means of a GMM with $\mathcal{K}$ components in the log-Mel domain, for

those features labeled as unreliable the clean speech estimate is

$$\hat{\mathbf{x}}_u = \sum_{k=1}^{\mathcal{K}} P\left(k \,|\mathbf{y}_r, \mathbf{y}_u\right) \hat{\mathbf{x}}_u^{(k)}, \tag{2.52}$$

where $\hat{\mathbf{x}}_u^{(k)}$ corresponds to the mean of a right-truncated Gaussian distribution defined in the interval $(-\boldsymbol{\infty}, \mathbf{y}_u]$ given the k-th Gaussian of the clean speech model. The posterior $P\left(k \,|\mathbf{y}_r, \mathbf{y}_u\right)$ can be understood as the weight of the partial estimate $\hat{\mathbf{x}}_u^{(k)}$. It should be noticed that correlations between the different elements in the feature vector can be exploited in a precise way, since $\mathbf{y}_r$ conditions the value of $\hat{\mathbf{x}}_u$ according to the posterior probabilities.

A critical issue concerning the missing-data approaches is to accurately segregate a noisy speech spectrogram into its reliable and unreliable regions. Such a segregation is typically accomplished by means of the so-called (binary) missing-data masks, which label each time-frequency (T-F) bin of a noisy speech spectrogram (pixel) as reliable (speech dominates) or unreliable (noise dominates). These masks are used for both types of missing-data paradigms (i.e. the modification of the recognizer to ignore those unreliable elements during recognition and data imputation) and their performance depend heavily on the quality of those masks. In this regard, a remarkable missing-data mask is the oracle one. The oracle missing-data mask perfectly segregates a noisy speech spectrogram into its reliable and unreliable regions with no errors. Therefore, this type of mask allows us to find out the upper limit a missing-data method performs. Unfortunately, in a real-life scenario, masks must be estimated and the estimation process hopelessly leads to estimation errors. Thus, an unreliable T-F bin can be classified as reliable and that is kept during the application of a missing-data method. On the other hand, classifying a reliable T-F bin as unreliable means that a reliable element is distorted by missing-data processing. Since those missing-data methods that operate by modifying the recognizer are more powerful than the data imputation techniques at the expense of higher computational complexity (similarly to what happened with the model adaptation and feature enhancement paradigms), better performance will be achieved by using a method belonging to the former category (such as marginalization) if a well estimated binary mask is available and the other way round [63, 66]. Instead of making a hard decision about the reliability of every T-F bin, another possibility is to give a soft measure of the reliability of the noisy spectrum regions to be exploited by the missing-data technique. In this case, the corresponding continuous mask in the [0, 1] interval is called soft missing-data mask or soft-mask. The values of such a type of mask can be understood as speech presence probabilities (SPPs). Thus, the closer to 1 (0) the mask value is, the more reliable (unreliable) the T-F bin is since speech (noise) tends to dominate. In fact, it has been shown that when the missing-data mask

where α is the sigmoid slope, and β is the sigmoid center. Appropriate values for these parameters are found via a series of tuning experiments. The valid range for α is $[0, \inf)$. For large values of α the sigmoid becomes steep and the resultant fuzzy mask approximates a traditional discrete missing data mask. In this case we are implicitly assuming a small variance in the noise estimation error. At the other extreme as the value of α tends to 0, we approach a mask where all values are 0.5. If $\alpha = 0$, we are assuming no knowledge of the noise and admitting maximum uncertainty into the mask.

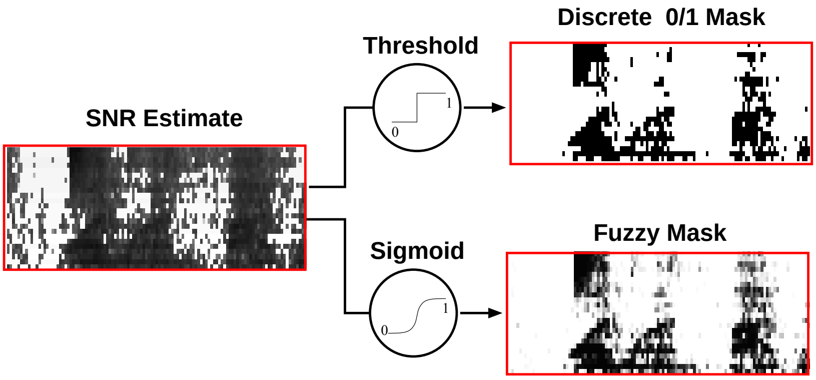

Figure 2.4: A comparison between binary and soft missing-data masks obtained from an *a priori* SNR estimate [14].

Figure 2: Illustration of the difference between discrete and fuzzy missing data masks.

Our use of fuzzy masks has parallels with the work of Renevey and Drygajlo in which a fuzzy mask is used in conjunction with missing data imputation[8].

has to be estimated, it is preferable to follow a soft missing-data approach for better performance [14]. Some brief comments on missing-data mask estimation are given immediately below.

4. EXPERIMENTS WITH FUZZY MASKS

The TIDigits corpus of digit sequences was used. Acoustic vectors were obtained via a 32 channel auditory filter bank (Cooke, 1993) with centre frequencies spaced linearly in ERB-rate from 50 to 8000 Hz. The instantaneous Hilbert envelope at the output of each filter was smoothed with a first order filter with an 8 ms time constant, and sampled at a frame-rate of 10 ms (this is the same representation as employed in [2] except here 32 channels are being used rather than 64). Finally, a cube root compression was applied to the frame of energy values. HTK [10] was used for training, and an in-house C++ decoder for recognition. Twelve models ('1'-'9', 'oh', 'zero' and 'silence') consisting of 8 no-skip, straight-through states with observations modeled with a 10 component diagonal Gaussian mixture were trained on clean speech.[2] An additional 1-state silence model was used to model the brief inter-digit

2.2.4.1 Missing-data mask estimation

Mask estimation is a critical part of the missing-data approaches as their performance heavily relies on the accuracy of such masks. Several approaches for missing-data mask computation can be found in the literature, like the SNR-dependent strategy, which is a popular and straightforward scheme. For instance, let us suppose that we have an estimate of the noise in the Mel power spectral domain $|\hat{N}(l,t)|^2$. Then, the *a priori* SNR for the corresponding T-F bin, $\hat{\xi}(l,t)$, can be calculated by means of the maximum likelihood estimator [43] as:

$$\hat{\xi}(l,t) = 10\log_{10}\left(\max\left(\frac{|\tilde{Y}(l,t)|^2}{|\hat{N}(l,t)|^2} - 1, 0\right)\right), \tag{2.53}$$

where $|\tilde{Y}(l,t)|^2$ is the related noisy power spectrum bin. To obtain every binary missing-data mask value, $m_b(l,t)$, the above *a priori* SNR estimate can be simply thresholded as follows:

$$m_b(l,t) = \begin{cases} 1 & \text{if } \hat{\xi}(l,t) \geq \gamma_m; \\ 0 & \text{otherwise,} \end{cases} \tag{2.54}$$

where γ_m is the SNR threshold in dB. The value of γ_m is often experimentally chosen taking into account the trade-off between false positives and false negatives.

The discrete mask is c
time-frequency point, a
where $X > \beta$. The fuz
through the sigmoid fu

For the derivatives, th
explained in Section 2.
and the fuzzy cases. N
the derivatives was used

A preliminary series o
find appropriate values
sigmoid function emplo
Informal tests were co
values, and then tests w
over a grid of α and β v
experiments were run u
and the factory noise
SNRs. It was found tha
largely independent of
type – as discussed bel

Finally, a third (discrete
(APR) was used to es
techniques. This mask
the positions of the reli
plane, obtained by comp
and treating points as r
a tuned acceptance thre

5. RESULT

Figures 3 and 4 show
helicopter noise at the
first the factory noise
("FBank32") is the bas
delta features or word
("+delta") produces a
3% or 4%. Adding wo
provement. Moving fro
("Fuzzy") leads to furt
The largest gains made
increases from 46% to 6
use a sigmoid that has
corrupted data.

Figure 4 shows a simil
copter noise. Here we s
provements though sta

[2]In previous work in
have been employed, thres
of the discrete missing da

On the other hand, the same *a priori* SNR term of (2.53) may be employed to get a soft-mask $m_s(l,t)$ through the application of the sigmoid function as

$$m_s(l,t) = \frac{1}{1+e^{-\alpha_s\left(\hat{\xi}(l,t)-\beta_s\right)}}, \tag{2.55}$$

where α_s and β_s are the slope and center parameters of the sigmoid function, respectively. It should be noticed that this type of function is quite appropriate for this purpose as it maps the SNR estimate to the [0, 1] interval. Large α_s values approximate the sigmoid function to a unit step function, which degenerates to binary mask as illustrated in Figure 2.4. On the contrary, as $\alpha_s \to 0$, all the mask values tend to be 1/2, representing a maximum uncertainty context. On the other hand, β_s plays a similar role to γ_m as it sets the turning point between the dominance of speech or noise. Properly adjusting these parameters is crucial to achieve proper masks, and such an adjustment will depend, in turn, on the quality of the SNR estimate. This SNR-based approach intended to compute soft-masks is followed in [91], where, additionally, the resulting masks are improved by treating them as images in which we can exploit the time-frequency correlation of speech. Moreover, the coefficients of this SNR-dependent mask can be used to weight the noisy spectrum in order to perform imputation in a very simple manner.

Another possibility is the use of SVMs (Support Vector Machines) as in [192, 193] to classify each T-F bin as reliable or unreliable. Because a key point of this kind of classifiers is the features that are used as input, in [192] this issue is explored by comparing different types of them. In that work it was determined that some auditory-based features such as GFCC and RASTA-PLP yield a robust classification as shown by their accuracy experiments. More precisely, both RASTA-PLP and pitch-based features improve the generalization ability of the classifier to unseen acoustic conditions during the training stage. In [193], linearly separable and discriminative features are extracted by means of a DNN to boost the performance of the SVM-based classification.

In addition, the use of deep learning has become relevant over the recent years to estimate missing-data masks because of the powerful modeling capabilities of that type of architectures. For example, for noise-robust ASR purposes, a similar set of features is used in [139] and [194] to estimate soft-masks by means of a DNN and a CNN (Convolutional Neural Network), respectively. Unfortunately, a comparison between both methods is not provided in the literature. In addition, a very powerful deep learning architecture called BLSTM (Bi-directional Long Short-Term Memory) network, which is able to exploit long temporal correlations, is considered in [76] to also estimate soft-masks for noise-robust ASR purposes through beamforming. Since mask

estimation is a quite difficult task, we will also jointly exploit DNNs and multi-channel information in Chapter 5 to efficiently obtain binary masks for data imputation.

2.3 Noise estimation

A noise estimation stage is required by a number of algorithms intended to provide robustness against noise in ASR. As an example of the latter, we have some feature enhancement methods such as Wiener filtering or VTS feature compensation. Moreover, we have seen that some missing-data mask estimation approaches may also need a noise estimate to work. Again, the performance of all these methods relies on the accuracy of the noise estimates. Because of this reason, despite it seems that the attention has moved over the last years towards other noise-robust solutions which do not require any explicit noise estimation, this issue has traditionally been important in the scientific literature. Hence, in this section we briefly review some prominent classical noise estimation techniques.

This overview will distinguish between different noise estimation categories, namely recursive averaging, minimum statistics and MMSE noise estimation, which are described in the following. Of course, while these are considered the most relevant paradigms, we can find a variety of approaches. For instance, one of the simplest noise estimation techniques consists of averaging the first frames of an utterance since it is often reasonable to assume that no speech is present in these frames. Nevertheless, it is clear that this approach will fail miserably in the presence of non-stationary noise. On the other hand, we also mentioned in Section 2.2 that NMF for source separation has become popular in recent times. Indeed, this approach has also been successfully explored for noise estimation purposes [177].

Recursive averaging. This is one of the oldest and simplest noise estimation paradigms. The recursive averaging philosophy is stated by the formula

$$|\hat{N}(f,t)|^2 = \begin{cases} \alpha|\hat{N}(f,t-1)|^2 + (1-\alpha)|Y(f,t)|^2 & \text{if } \frac{\sum_f |Y(f,t)|^2}{\sum_f |\hat{N}(f,t-1)|^2} < \beta; \\ |\hat{N}(f,t-1)|^2 & \text{otherwise,} \end{cases} \tag{2.56}$$

where α is a smoothing parameter ($0 < \alpha < 1$) and β is a threshold to control the above recursion. When the noisy observation $\sum_f |Y(f,t)|^2$ is less than $\beta \sum_f |\hat{N}(f,t-1)|^2$, it is assumed that speech is absent and the noise power spectrum estimate is updated by taking into account an energy fraction of the current noisy observation. On the other hand, when $\sum_f |Y(f,t)|^2 \geq \beta \sum_f |\hat{N}(f,t-1)|^2$, it is considered that speech is present,

so the recursion is stopped and the last noise estimate is kept. It should be noticed that

$$\gamma(t) = \frac{\sum_f |Y(f,t)|^2}{\sum_f |\hat{N}(f,t-1)|^2} \tag{2.57}$$

can be understood as an estimation of the *a posteriori* SNR. Indeed, different criteria to decide if speech is absent or present can be used instead, e.g. SPP-based criteria. A typically suitable assumption is considering that the first frames of a noisy utterance contain only noise energy. Therefore, the initialization of this algorithm can be carried out as $|\hat{N}(f,0)|^2 = |Y(f,0)|^2$. Moreover, this recursive averaging philosophy can be alternatively applied to estimate the spectral magnitude of the noise instead of its power.

The above approach exhibits poor performance when tackling with non-stationary noise. For instance, if noise energy considerably increases in a particular time frame, the method will believe that speech is present. As a consequence, the noise estimate will not be updated, leading to noise being underestimated. This issue was partially solved by Cohen in 2002 [29] in an efficient way in terms of computational complexity. His algorithm was called MCRA (Minima Controlled Recursive Averaging), which is able to track rapid shifts in the noise spectra. Unlike the above basic recursive averaging approach, MCRA does not stop updating the noise estimate when speech is present. Instead, a soft estimation is carried out based on an SPP, $p(f,t)$, as

$$|\hat{N}(f,t+1)|^2 = \tilde{\alpha}(f,t)|\hat{N}(f,t)|^2 + (1-\tilde{\alpha}(f,t))|Y(f,t)|^2, \tag{2.58}$$

where

$$\tilde{\alpha}(f,t) = \alpha + (1-\alpha)p(f,t) \tag{2.59}$$

is a time-varying smoothing parameter. The SPP is computed from a kind of *a posteriori* SNR estimate defined as $S_r(f,t) = S(f,t)/S_{min}(f,t)$, where $S(f,t)$ is a smoothed measure of the local noisy speech energy and $S_{min}(f,t)$ is the minimum of $S(f,t)$ within a window of length D. Thus, the proposed recursive SPP estimator is given by

$$p(f,t) = \alpha_p p(f,t-1) + (1-\alpha_p)I(f,t), \tag{2.60}$$

where $\alpha_p \in (0,\ 1)$ is another smoothing parameter and

$$I(f,t) = \begin{cases} 1 & \text{if } S_r(f,t) > \delta; \\ 0 & \text{otherwise.} \end{cases} \tag{2.61}$$

It must be noted that δ is a new threshold to be experimentally determined. Therefore, if $S_r(f,t)$ is greater than δ, it is considered that speech is present and vice versa.

An improved version of MCRA (IMCRA) was also published by Cohen in 2003 [28]. IMCRA differentiates itself from MCRA in two main aspects: a factor to compensate the bias of the MCRA estimator is introduced and the SPP is now computed in a more robust way. Hence, let B be a factor that compensates the bias of the noise estimation in (2.58), the resulting noise estimate in IMCRA is

$$|\tilde{N}(f,t+1)|^2 = B \cdot |\hat{N}(f,t+1)|^2 = \underbrace{\left.\frac{|N(f,t)|^2}{\mathrm{E}\left[|\hat{N}(f,t)|^2\right]}\right|_{\xi(f,t)=0}}_{B} \cdot |\hat{N}(f,t+1)|^2. \tag{2.62}$$

Additionally, the SPP $p(f,t)$ is calculated following a Bayesian approach by also assuming that the noisy speech STFT (Short-Time Fourier Transform) coefficients can be modeled by a complex Gaussian distribution.

Other upgrades to recursive averaging-based noise estimation were reported later in the literature, such as MCRA-MAP (MCRA-Maximum A Posteriori) [102] or EMCRA (Enhanced MCRA) [46], which will not be discussed here for the sake of brevity.

To conclude this part, let us make some comments about the noise estimation algorithm published by Rangachari *et al.* in 2006 [156]. This method is strongly based on MCRA. According to its authors, their algorithm exhibits a greater adaptation speed to changes in the noise dynamics than the previous MCRA-based methods. Rangachari's algorithm starts by computing a smoothed noisy speech periodogram as follows:

$$P(f,t) = \alpha P(f,t-1) + (1-\alpha)|Y(f,t)|^2. \tag{2.63}$$

Then, $P(f,t)$ is employed to get the corresponding SPP as in MCRA (see Eq. (2.60)). A kind of *a posteriori* SNR is again estimated as $S_r(f,t) = P(f,t)/P_{min}(f,t)$, where $P_{min}(f,t)$ is obtained by means of the following non-linear rule:

$$P_{min}(f,t) = \begin{cases} \gamma P_{min}(f,t-1) + \frac{1-\gamma}{1-\beta}(P(f,t) - \beta P(f,t-1)) & \text{if } P_{min}(f,t-1) < P(f,t); \\ P(f,t) & \text{otherwise,} \end{cases} \tag{2.64}$$

where β and γ are heuristic parameters to be experimentally set. As for MCRA, if $S_r(f,t) > \delta(f)$ speech is present and $I(f,t) = 1$. Otherwise, speech is absent and $I(f,t) = 0$. Therefore, the only difference between Rangachari's algorithm and MCRA at this point is that the threshold $\delta(f)$ is frequency-dependent. Finally, the noise power spectrum is recursively estimated as in (2.58).

Minimum statistics. This method, that was proposed by Martin in 2001 [130], is intended to give noise estimates in the linear power spectral domain. It works by

tracking through time the minimum value, $P_{min}(f,t)$, of the smoothed noisy speech periodogram $P(f,t)$ within a window of length D. In other words, $P_{min}(f,t)$ corresponds to the noise estimate in this method.

The smoothed noisy speech periodogram required to derive $P_{min}(f,t)$ is computed as

$$P(f,t) = \alpha(f,t)P(f,t-1) + (1-\alpha(f,t))|Y(f,t)|^2, \tag{2.65}$$

where $\alpha(f,t)$ is a smoothing parameter, which changes across time and frequency to improve the noise estimation accuracy. This smoothing parameter is calculated by minimizing the MSE between $P(f,t)$ and the actual noise power $\sigma_N^2(f,t)$ when speech is absent, that is,

$$\hat{\alpha}(f,t) = \underset{\alpha(f,t)}{\operatorname{argmin}} \left(\mathrm{E}\left[\left(P(f,t) - \sigma_N^2(f,t) \right)^2 \middle| P(f,t-1), \sigma_X^2(f,t) = 0 \right] \right), \tag{2.66}$$

in which $\sigma_X^2(f,t)$ refers to the actual clean speech power. Since the minimum value of a set of random variables is smaller than its mean for non-trivial distributions, the above noise estimation, $P_{min}(f,t)$, is necessarily biased. Hence, the minimum statistics method introduces a bias compensation factor, $B_{min}(f,t)$, to get the final noise estimation $\hat{\sigma}_N^2(f,t)$ as follows:

$$\hat{\sigma}_N^2(f,t) = \underbrace{\frac{1}{\mathrm{E}[P_{min}(f,t)]\,\big|_{\sigma_N^2(f,t)=1}}}_{B_{min}(f,t)} \cdot P_{min}(f,t) = B_{min}(f,t) \cdot P_{min}(f,t). \tag{2.67}$$

MMSE noise estimation. This category refers to those algorithms that estimate the noise power spectrum through an MMSE criterion. Two well-known methods belonging to this classification are those developed by Yu in [208] and Hendriks in [72]. In fact, the Hendrik's estimator is based on the previous Yu's work, so we will only focus on describing the former method hereunder.

The MMSE noise estimation method of Hendriks proceeds from the following assumptions:

- An additive noise distortion model (i.e. there is no convolutive distortion).
- The noisy speech STFT coefficients are i.i.d. (independent and identically distributed) zero-mean complex random variables.
- Speech and noise are statistically independent.

Thus, $|N(f,t)|^2$ is estimated from an MMSE criterion as the second-order moment of the noise spectral magnitude, that is,

$$\begin{aligned} |\hat{N}(f,t)|^2 &= \mathrm{E}\left[|N(f,t)|^2 \,|Y(f,t)\right] \\ &= \int_0^{+\infty}\int_0^{2\pi} |N(f,t)|^2 p\left(|N(f,t)|, \Delta(f,t)|\, Y(f,t)\right) d\Delta(f,t) d|N(f,t)|, \end{aligned} \tag{2.68}$$

where $\Delta(f,t)$ is the corresponding noise phase term. Then, the conditional PDF $p\left(|N(f,t)|, \Delta(f,t)|\, Y(f,t)\right)$ is determined by means of the Bayes' theorem as

$$p\left(|N(f,t)|, \Delta(f,t)|\, Y(f,t)\right) = \frac{p\left(Y(f,t)|\,|N(f,t)|, \Delta(f,t)\right) p(|N(f,t)|, \Delta(f,t))}{p(Y(f,t))}, \tag{2.69}$$

where, assuming that both the speech and noise STFT coefficients follow complex Gaussian distributions,

$$p\left(Y(f,t)|\,|N(f,t)|, \Delta(f,t)\right) = \frac{1}{\pi\sigma_X^2(f,t)} e^{\frac{2|N(f,t)||Y(f,t)|\cos(\Delta(f,t)-\Theta(f,t))-|Y(f,t)|^2-|N(f,t)|^2}{\sigma_X^2(f,t)}} \tag{2.70}$$

and

$$p(|N(f,t)|, \Delta(f,t)) = \frac{|N(f,t)|}{\pi\sigma_N^2(f,t)} \exp\left(-\frac{|N(f,t)|^2}{\sigma_N^2(f,t)}\right). \tag{2.71}$$

It must be noticed that $\Theta(f,t)$ is the phase term of the noisy speech. Additionally, the *a priori* distribution of the noisy speech can be computed by marginalizing as follows:

$$\begin{aligned} p(Y(f,t)) &= \int_0^{+\infty}\int_0^{2\pi} p(Y(f,t), |N(f,t)|, \Delta(f,t)) d\Delta(f,t) d|N(f,t)| \\ &= \int_0^{+\infty}\int_0^{2\pi} p(Y(f,t)||N(f,t)|, \Delta(f,t)) p(|N(f,t)|, \Delta(f,t)) d\Delta(f,t) d|N(f,t)|. \end{aligned} \tag{2.72}$$

By combining (2.72) and (2.69) with (2.68), the MMSE estimation of the noise power spectrum can be rewritten as

$$\begin{aligned} |\hat{N}(f,t)|^2 &= E\left[|N(f,t)|^2 |Y(f,t)\right] \\ &= \frac{\int_0^{+\infty}\int_0^{2\pi} |N(f,t)|^2 p(Y(f,t)||N(f,t)|, \Delta(f,t)) p(|N(f,t)|, \Delta(f,t)) d\Delta(f,t) d|N(f,t)|}{\int_0^{+\infty}\int_0^{2\pi} p(Y(f,t)||N(f,t)|, \Delta(f,t)) p(|N(f,t)|, \Delta(f,t)) d\Delta(f,t) d|N(f,t)|}. \end{aligned} \tag{2.73}$$

It can be shown that the above equation can be alternatively expressed in an easier way in terms of the *a priori* and *a posteriori* SNRs, $\xi(f,t)$ and $\zeta(f,t)$, as

$$|\hat{N}(f,t)|^2 = \left(\frac{1}{(1+\xi(f,t))^2} + \frac{\xi(f,t)}{(1+\xi(f,t))\zeta(f,t)}\right) |Y(f,t)|^2. \tag{2.74}$$

Despite the estimation in (2.74) is unbiased, in practice it can exhibit a bias as a result of the used *a priori* SNR estimate. Then, the noise PSD estimate is derived from properly compensating the SNR-dependent bias of (2.74). Furthermore, the latter unbiased estimate is smoothed by application of a first-order recursion to decrease the estimation variance.

An illustrative comparison between all these noise estimation algorithms in terms of the mean and variance of the estimation error is given in [178]. In this work, it is concluded that, in general, the evaluated noise estimation algorithms perform better at lower than at higher SNRs. Only minimum statistics and the MMSE noise estimation algorithm of Hendriks exhibit a more stable performance regardless the SNR value. Furthermore, this latter method is the most robust one in the presence of rapid shifts of the noise spectra, while all of them perform very well when tackling with stationary noise.

2.4 Multi-channel robust speech processing on IMDs

Since multi-channel robust speech processing is an immensely vast topic, we focus in this section on its application to IMDs as this is the scope of this Thesis. In particular, most of this review focuses on multi-channel speech enhancement approaches because they have widely been considered for multi-channel noise-robust ASR in a successful way. Multi-channel robust speech processing on IMDs has gained popularity over the recent years due to both its potential regarding the single-channel solutions and a decrease in the price of hardware.

The rest of this section is organized as follows. First, an overview of multi-channel robust ASR on IMDs is presented. Since beamforming is a basic cornerstone of multi-channel robust ASR, some of its fundamentals are also discussed. Despite beamforming (or spatial filtering) exhibits some important constraints when performing on arrays comprised of a few sensors close each other (as it is the case of IMDs), such an approach is typically followed in the literature. Nevertheless, it is fair to say that we can find that the beamforming shortcomings are overcome in some way by post-filtering, i.e. by additional processing at the output of the spatial filter. Then, we give some comments on the most well-known fixed beamformers, delay-and-sum and MVDR (Minimum Variance Distortionless Response), as well as on adaptive array processing. Post-filtering is later revisited in Subsection 2.4.3. To conclude this section, the dual-channel power

level difference (PLD) principle is discussed. As we will see, this principle explains the spatial particularities of the speech and noise signals in a dual-microphone set-up where the secondary microphone is usually placed in an acoustic shadow. We cannot forget that the dual-microphone set-up can be widely found in many portable devices such as smartphones. Since beamforming does not account for the particularities of this scenario, the PLD scheme is often a more reasonable choice for designing noise-robust ASR solutions in this context. In general, the PLD principle will be taken into account when developing the contributions presented in the subsequent chapters of this Thesis.

2.4.1 Overview of multi-channel robust ASR on IMDs

First research works on multi-channel noise-robust ASR were based on the use of features extracted from an enhanced waveform in terms of SNR by means of beamforming [93, 143]. However, increasing the speech signal quality does not necessarily mean better accuracy of the speech recognizer since this one works on a feature-level basis. Because of this reason, the improvements that could be achieved by following a classical beamforming approach were limited in the first instance [160]. Thus, the scientific community in this area began to move towards designing specific multi-channel noise-robust ASR techniques. For example, in [161], a filter-and-sum beamformer is optimized in order to produce a sequence of features which maximizes the likelihood of generating the correct hypothesis. This technique, specifically intended to speech recognition, and called LIMABEAM (LIkelihood-MAximizing BEAMforming), definitively outperformed the classical beamforming scheme at that time. Several improvements to LIMABEAM were reported later in the literature and a remarkable one is S-LIMABEAM (Subband LIMABEAM) [162]. In contrast to the basic approach of [161], S-LIMABEAM uses subband processing principles to define an algorithm that can be successfully applied to highly reverberant environments.

Apart from the well-known beamforming techniques (commented later in Subsection 2.4.2) applied to noise-robust ASR, there are many different and creative solutions depending on the characteristics of the microphone array (i.e. a set of microphones spatially distributed). For instance, it was proposed in [142] an algorithm to select the signal from one microphone of the available ones in the array to be used for feature extraction and recognition. This makes sense since the relative position between the speaker and the sensors of the array can be variable, and therefore the quality of the signals captured by the microphones. This technique works by selecting the channel that produces the least mismatch when comparing speech recognition hypothesis made from compensated and noisy feature vectors. It has been shown that the use of one

selected channel can even produce less word error rates than the use of delay-and-sum beamforming [99].

As we know, a particular case of interest is the small microphone array, composed by a few sensors close each other. As also introduced, small microphone arrays have experienced a wide spreading over the last years as they are being embedded in the latest IMDs, such as smartphones or tablets. To take advantage of this small array feature, in [134] it was proposed to enhance the output waveform from a filter-and-sum beamformer by Wiener post-filtering to carry out an estimation of the *a priori* SNR. Then, this is mapped to a soft missing-data mask by means of a sigmoid function for noise-robust ASR purposes on small microphone arrays. This scheme showed to be effective by outperforming related single-channel approaches in terms of word accuracy.

A major step in multi-channel noise-robust ASR on IMDs was the occurrence of the 3rd CHiME Speech Separation and Recognition Challenge (CHiME-3) [15]. This challenge allowed the researchers making further developments in noise-robust ASR on IMDs with several sensors. More precisely, CHiME-3 encouraged to deal with speech distortions present when using a tablet with six microphones in everyday, noisy environments. A number of solutions were submitted by the participants and those benefited from a combination of single- and multi-channel techniques. While a large proportion of word error rate reduction came from exploiting the power of deep learning for both acoustic and language modeling [169, 190, 203, 210], multi-channel processing also played its part. In this regard, we can find several works that integrate, on the front-end side, a beamformer followed by additional processing at its output to mitigate the shortfalls of the spatial filtering [12, 76, 125, 153, 210].

A good example of a system that combines several single- and multi-channel approaches to achieve an outstanding performance is that of the challenge winners [203]. In this work, MVDR beamforming is applied to the previously dereverberated array signals in order to enhance speech. However, it must be pointed out that the baseline performance is critically improved by using convolutional and recurrent neural networks for acoustic and language modeling, respectively. Recurrent neural networks (RNNs) for language modeling and MVDR beamforming are also used in [190] along with fMLLR for inter-speaker variability compensation. Indeed, a beamforming stage is integrated by many of the challenge participants, e.g. [12, 76, 90, 95, 125, 153, 190, 210]. Apart from MVDR beamforming [12, 90, 95, 153, 190, 210], the use of delay-and-sum beamforming is also explored in a successful manner [12, 95, 125, 153]. Additionally, a quite robust type of beamforming with remarkable performance is studied in [76]. This beamformer, called GEV (Generalized EigenValue), is based on maximizing the SNR of the beamformer output in each frequency bin separately. GEV requires knowledge

about the speech and noise PSDs, which are computed from soft-masks estimated by means of a BLSTM neural network. At the output of the beamformer, a post-filtering stage is appended to improve its performance. Post-filtering is also considered in other works submitted to the challenge, e.g. [12, 125, 153, 210], and more details about it are given in Subsection 2.4.3. Adaptive beamforming was considered in this challenge as well through the use of MCA (Multi-Channel Alignment) in [174], which demonstrates to be simple and effective at the same time. Moreover, multi-channel Wiener filtering (MCWF) is explored in [191] obtaining significant improvements.

Following the success of the CHiME-3 Challenge, the CHiME-4 Challenge [187] constitutes a step forward in terms of difficulty as the participants are additionally constrained to only use one and two microphones (i.e. one of those that face forward plus the one that faces backwards) of the tablet apart from the six available. By using the six microphones in the tablet, the winners of the CHiME-4 Challenge [41] achieved an impressive performance with a word error rate around 2% on the corresponding medium-vocabulary recognition task. This was achieved by again combining single- and multi-channel techniques: *1)* beamforming-based enhancement, *2)* diversified training data using the noisy data of each channel, and the multiple beamformers' outputs data of 6 channels and 2 channels, *3)* deep convolutional neural networks (DCNNs) for acoustic modeling, and *4)* LSTM-based language modeling. Another interesting contribution to this challenge with nice results is that of [199], where the beamforming weights are calculated in different manners by either employing neural networks or via MLE.

Apart from the case of the tablet described above, we should also consider the smartphone scenario because of its relevance. Smartphones rarely embed more than two or three sensors due to their reduced size. Moreover, classical beamforming exhibits poor performance in this case because there are only two or three sensors very close each other, and one or two of them are often placed in an acoustic shadow regarding the speaker's mouth [179, 180]. In this situation, it will be shown that it is better to follow different strategies, such as the PLD principle. Thus, noise-robust speech processing on dual-microphone smartphones will be reviewed in Subsection 2.4.4.

2.4.2 Beamforming

Spatial filtering, better known as beamforming, is a mature technology the origin of which is in narrowband signal processing intended to applications such as RADAR (RAdio Detection And Ranging). Beamforming is also extended to broadband signals such as speech, in such a way that this technology can be applied to enhance a distorted speech signal captured by a microphone array. In summary, beamforming is about

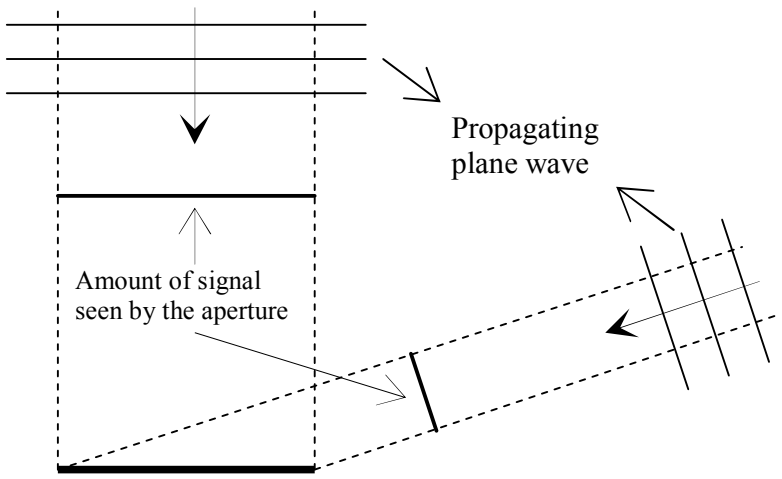

Figure 2.5: Depiction of a passive and continuous linear aperture receiving from two different sources [79].

Figure 2.2: Incoming signal received by a linear aperture.

exploiting propagation physical principles in order to design spatial filters that "look" towards the target source direction while attenuating the undesired signals coming from other directions. To do this, a set of spatially distributed sensors (namely array) is employed.

First, an introduction about beamforming explaining some of its fundamentals is given along with an overview of noise fields. Then, two of the most popular fixed beamforming techniques which are widely used in the literature, i.e. delay-and-sum and MVDR, are reviewed. To conclude this subsection, brief comments on adaptive beamforming are provided. It must be pointed out that the writing of this subsection has been mainly based on the references [79, 132, 160].

2.4.2.1 Fundamentals and noise fields

A notion about what beamforming is has been given immediately above. To further deepen, let us define an aperture as a space region receiving (passive aperture) or emitting (active aperture) propagation waves. Microphones and loudspeakers are good examples of passive and active apertures, respectively. Moreover, a microphone array might be understood as a passive and continuous aperture which is sampled at a finite number of discrete points [132]. Figure 2.5 depicts a passive and continuous linear aperture receiving from two different sources. As can be observed, the amount of signal seen by the aperture depends on the direction of arrival.

For illustrative purposes as well as for simplicity, let us suppose a linear micro-

1. Aperture physical constraints over all frequencies.
2. The signal is assured to arrive from the far field, thus

$$|\mathbf{p}| \geq \frac{2L^2}{\lambda} \tag{2.11}$$

The aperture function may be written as

$$A_R(p_a) = rect(x_a/L) \tag{2.12}$$

where $rect(.)$ is the rectangular function. In this case, the resulting directivity pattern simplifies to

$$D_R(j\alpha_x) = \mathcal{F}\{rect(x_a/L)\} = Lsinc(\alpha_x L) \tag{2.13}$$

where

$$sinc(x) \equiv \frac{sin(x)}{x} \tag{2.14}$$

The plot of directivity pattern corresponding to a uniform aperture function is shown in Figure 2.3. The directivity pattern covers the area of $-\lambda/L \leq \alpha_x \leq \lambda/L$ is referred to as the *main lobe* and its extent is termed the *beam width*. From the micro-pattern plot, the zeros are located at $\alpha_x = m\lambda/L$. Note that the beam width of

phone array comprised of N isotropic sensors receiving a signal $y(m)$ from an arbitrary source. The N sensors are linearly distributed and the distance between each pair of adjacent sensors is d. Furthermore, the total length of the array is L. Again for illustrative purposes and for the sake of simplicity, we will assume that $y(m)$ arrives to the microphone array from a far-field source, that is,

$$\rho > \frac{2L^2}{\lambda}, \tag{2.75}$$

where ρ is the distance between the emitting source and the array, and λ is the wavelength of the signal received by the array. Despite the wavefront is spherical, under the far-field assumption it is reasonable to approximate that the source propagates as a plane wave at a sufficient distance. The far-field framework leads to simpler mathematical derivations regarding the near-field source case as it can be assumed that the same signal amplitude is observed by every element of the array. Indeed, the following formulae could be adapted to account for near-field sources (spherical wavefronts) by just taking into consideration that the signal amplitude attenuates proportionally to the increase of the distance.

By taking into account all the considerations above as well as assuming that all the sensors exhibit the same frequency response, at the output of the microphone array we observe the signal

$$x(m) = \sum_{n=1}^{N} y(m - (n-1)\tau), \tag{2.76}$$

where

$$\tau = \frac{d\cos(\theta)}{c_s} \tag{2.77}$$

is the time difference of arrival (TDoA) between two adjacent sensors, and θ and c_s are the angle of incidence of the arriving wave and the speed of sound, respectively. The latter variable depends on different factors such as the pressure and temperature of the fluid where the sound propagates. In the air, $c_s \approx 340$ m/s. The impulse response of the microphone array can be computed by just replacing $y(m)$ by the Dirac delta function $\delta(m)$ in (2.76):

$$h(m) = \sum_{n=1}^{N} \delta(m - (n-1)\tau). \tag{2.78}$$

The above equation can alternatively be expressed in the frequency domain by application of the discrete-time Fourier transform (DTFT) as

$$H(f,\theta) = \sum_{n=1}^{N} e^{-j2\pi f((n-1)\tau)} = \sum_{n=1}^{N} e^{-j2\pi f\left((n-1)\frac{d\cos(\theta)}{c_s}\right)}, \tag{2.79}$$

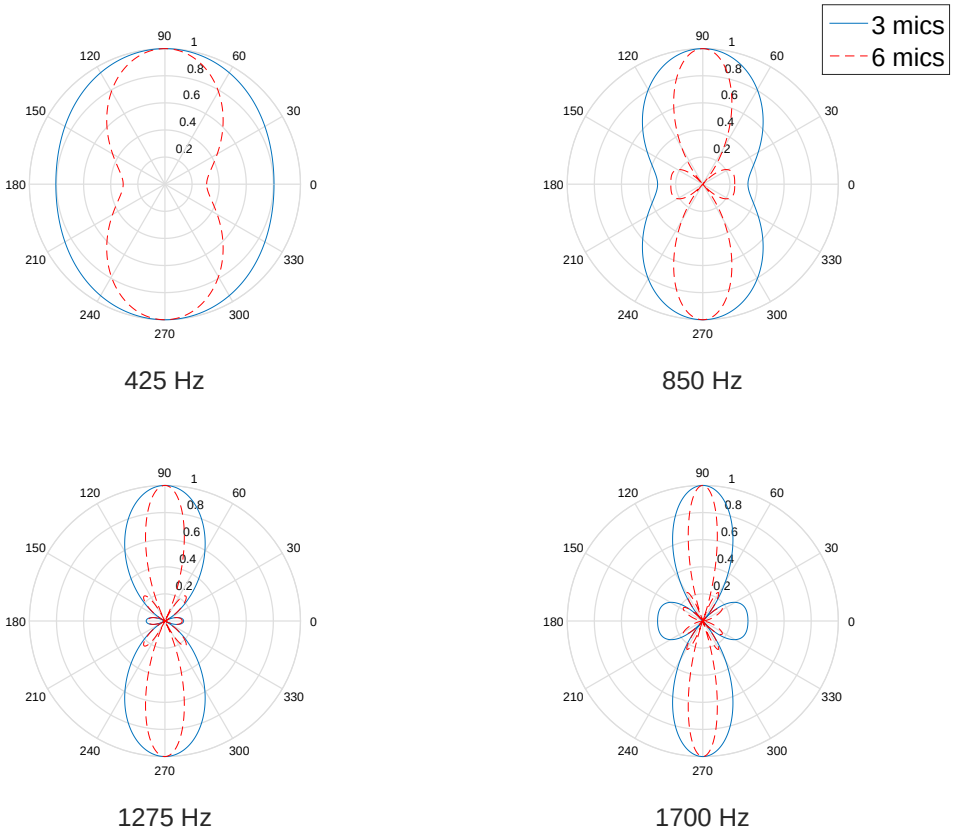

Figure 2.6: Beampatterns of two microphone arrays with 3 and 6 sensors at four different frequency values: 425 Hz, 850 Hz, 1275 Hz and 1700 Hz. In both arrays, the inter-element spacing is $d = 0.1$ m.

where f means (normalized) frequency and $H(f, \theta)$ corresponds to the microphone array response in the frequency domain. As we can see, the system response depends on both the frequency and the direction of arrival of the signal. Figure 2.6 shows polar plots of $H(f, \theta)$ drawn at four different frequency values for two different microphone arrays with 3 and 6 sensors, and $d = 0.1$ m in both cases. These plots are known as directivity patterns or beampatterns, and they represent the response of the aperture as a function of the direction of arrival of the signal, θ. As we can see, the greater the frequency or the number of sensors, the greater the directivity of the array. More precisely, now we can state that beamforming is about designing a desired shaping and steering of the array directivity pattern [132] to direct its main lobe towards the look direction, i.e. where the target source is.

According to the Nyquist-Shannon sampling theorem, temporal aliasing appears when a signal the maximum frequency of which is f_{max}, is sampled below a $f_s = 2f_{max}$

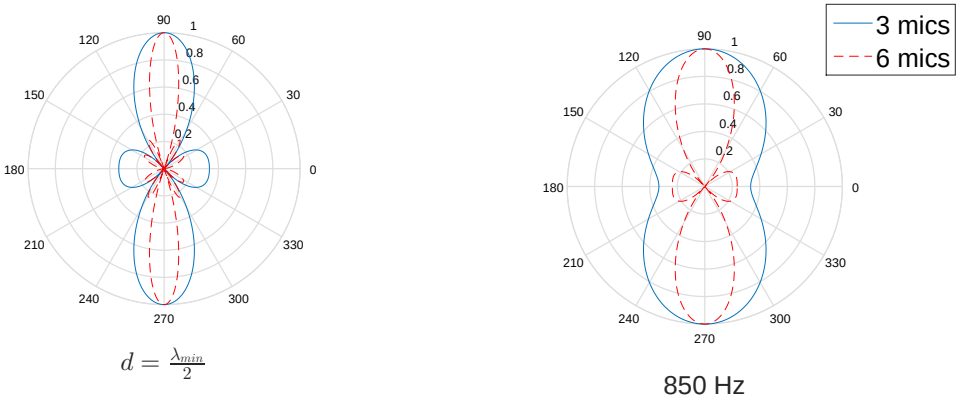

Figure 2.7: Example of spatial aliasing (right) for the microphone arrays of Figure 2.6.

sampling rate. Analogously, this concept is extended to the spatial filtering scenario. Thus, it can be shown that to avoid spatial aliasing it is required that

$$d < \frac{\lambda_{min}}{2}, \tag{2.80}$$

where λ_{min} is the minimum wavelength of the target signal. This is known as the spatial sampling theorem [99]. If (2.80) is not guaranteed, the target signal is undersampled in the spatial domain and large sidelobes appear looking toward undesired directions. This is illustrated by Figure 2.7.

Noise fields. A noise field refers to a space region where noise propagates. Noise fields are characterized by the so-called coherence function, $\Gamma_{kl}(f)$, which is defined as [30]

$$\Gamma_{kl}(f) = \frac{\Phi_{kl}(f)}{\sqrt{\Phi_{kk}(f)\Phi_{ll}(f)}}, \tag{2.81}$$

where, on the one hand, $\Phi_{kl}(f)$ is the cross-PSD between the noise observed at two different spatial points k and l. Similarly, $\Phi_{kk}(f)$ and $\Phi_{ll}(f)$ are PSDs of noise at each point (k and l). From (2.81), we can see that the coherence function $|\Gamma_{kl}(f)| \in [0,\ 1]$ measures the degree of correlation of two noise signals observed at different spatial points. In microphone arrays, sometimes it is useful to define a noise coherence matrix for calculation purposes as follows:

$$\mathbf{\Gamma}(f) = \begin{pmatrix} \Gamma_{11}(f) & \cdots & \Gamma_{1N}(f) \\ \vdots & \ddots & \vdots \\ \Gamma_{N1}(f) & \cdots & \Gamma_{NN}(f) \end{pmatrix}, \tag{2.82}$$

where $k, l = 1, ..., N$ refer to the sensors of the microphone array comprised of N elements.

The most representative types of noise fields are the coherent, incoherent and diffuse noise fields. These are briefly described down below.

In a **coherent noise field**, signals arrive to the microphone array with no reflections, dispersion or dissipation caused by the acoustic environment. Noise signals at any two spatial points are strongly correlated in such a manner that $|\Gamma_{kl}(f)| \approx 1 \; \forall f$. A coherent noise field often corresponds to open areas with no obstacles.

At any two different spatial points, noise signals are completely uncorrelated in an **incoherent noise field**. Therefore, $\Gamma_{kl}(f) \approx 0 \; \forall f$ and $\mathbf{\Gamma}(f) \approx \mathbf{I}_N$, where $\mathbf{I}_N$ is an $N \times N$ identity matrix. The incoherent noise is also known as spatially white noise, and it is difficult to find in real-life conditions.

Finally, a **diffuse noise field**, which is also called spherical isotropic noise field, exhibits two main features: homogeneity (namely the same noise PSD is observed at any two spatial points) and isotropy (i.e. the incident intensity distribution of noise is uniform). A diffuse noise field is quite appropriate to describe real-life reverberant environments such as offices or vehicle interiors. The coherence function is given by

$$\Gamma_{kl}(f) = \text{sinc}\left(\frac{2\pi f d_{kl}}{c_s}\right), \tag{2.83}$$

where $\text{sinc}(x) = \sin(x)/x$ and d_{kl} is the distance between the spatial points k and l. From (2.83), we can determine that for very close spatial points the coherence tends to be high at low frequencies. As both d_{kl} and f increase, the coherence sharply decreases. The main diagonal of $\mathbf{\Gamma}(f)$ is an all-ones vector and $|\Gamma_{kl}(f)| < 1 \; \forall k \neq l$ and $f > 0$.

2.4.2.2 Delay-and-sum beamforming

Delay-and-sum is the simplest beamforming method. It consists of the compensation of the time differences of arrival of the target signal to the microphone array. Intuitively, delay-and-sum produces a target signal constructive interference while expecting some degree of noise destructive interference. In fact, if the noise signals captured by the array microphones are uncorrelated to each other and to the target signal, delay-and-sum yields a 3 dB increase of the SNR at its output for every doubling of the number of sensors in the array [99]. The way that delay-and-sum beamforming works is illustrated by Figure 2.8.

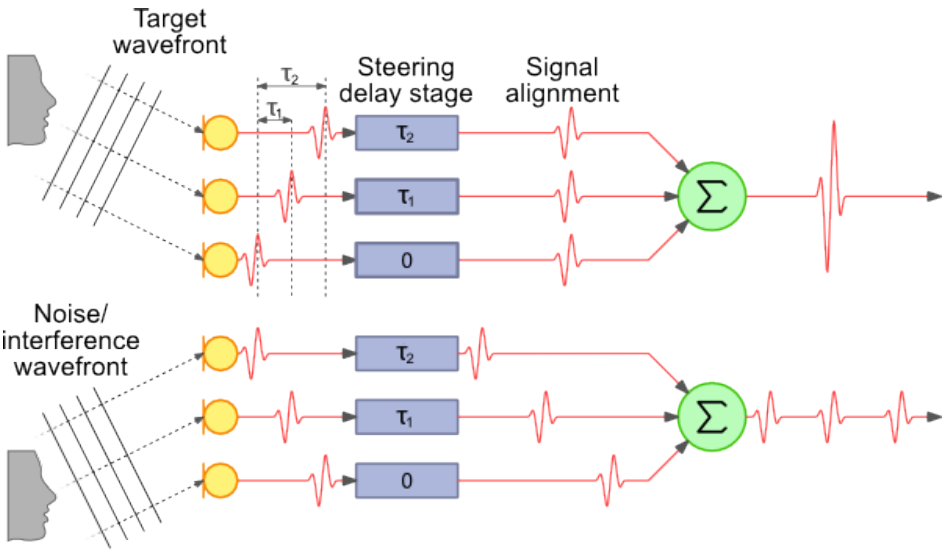

Figure 2.8: Diagram on how delay-and-sum beamforming works [69].

Let $y_n(m)$ be the signal observed by the n-th ($n = 1, ..., N$) sensor of the microphone array, at the output of a delay-and-sum beamformer we obtain the signal

$$x(m) = \sum_{n=1}^{N} \bar{w}_n y_n(m - \tau_n), \tag{2.84}$$

where typically $\{\bar{w}_n = 1/N;\ n = 1, ..., N\}$ is the set of weights chosen to average the time-aligned signals $y_n(m - \tau_n)$ and

$$\tau_n = \frac{d_n}{c_s} = \frac{(n-1)d\cos(\theta)}{c_s} \tag{2.85}$$

is the TDoA between the reference sensor ($n = 1$) and the n-th one. For TDoA computation, a variety of methods can be found in the literature and many of them are based on cross-correlation [160]. For instance, in [21], several clustering- and angular spectrum-based methods for TDoA computation are evaluated and compared. In that work, it is concluded that an SNR-based angular spectrum method and GCC-PHAT (Generalized Cross-Correlation with PHAse Transform) perform the best for small and larger spacing between microphones, respectively.

Alternatively, the set of weights in (2.84) can be selected to attribute more or less importance to each sensor. In such a case, the resulting beamformer is called weighted delay-and-sum. For example, the weighted delay-and-sum beamformer reported in [8] was used by several participants of the CHiME-3 Challenge, e.g. [12, 125], due to the good results provided by this technique with low computational complexity.

In general, beamforming is often directly applied in the frequency domain as

$$X(f) = \mathbf{w}^H(f)\mathbf{y}(f), \tag{2.86}$$

where $x(m)$ is obtained by taking $X(f)$ back to the time domain by application of the inverse Fourier transform (IFT). Moreover, the superscript H denotes Hermitian transpose, and $\mathbf{w}(f)$ and $\mathbf{y}(f)$ are, respectively, vectors of weights and observations:

$$\begin{aligned} \mathbf{w}(f) &= (W_1(f), ..., W_N(f))^\top; \\ \mathbf{y}(f) &= (Y_1(f), ..., Y_N(f))^\top. \end{aligned} \tag{2.87}$$

For unweighted delay-and-sum it is clear that

$$W_n(f) = \frac{1}{N} e^{-j2\pi f \tau_n}, \quad n = 1, ..., N. \tag{2.88}$$

Delay-and-sum can be understood as a particular case of a more general type of beamforming known as filter-and-sum. In filter-and-sum beamforming, every array sensor is associated with a filter $h_n(m)$, in such a manner that at the output of the beamformer we have the signal

$$x(m) = \sum_{n=1}^{N} \sum_{l=0}^{p-1} h_n(l) y_n(m - l - \tau_n). \tag{2.89}$$

Of course, filter-and-sum beamforming can be alternatively performed directly in the frequency domain. Generally, a beamformer weight in this domain is expressed in terms of its gain and phase, parameters that are designed to shape the beampattern and to direct the main lobe towards the desired look direction, respectively. Several additional criteria can be followed to design beamforming weights and MVDR [79] is one of the most popular, which is presented later in this subsection.

Because delay-and-sum can be considered a filter-and-sum beamformer comprising zero-order FIR filters (only one filter coefficient per array microphone) according to (2.89), it is clear that it is not able to deal with reverberant environments. To circumvent this issue, a matched filter approach was proposed in [50]. Nevertheless, in practice, this variant provides small improvements in terms of recognition accuracy over the conventional delay-and-sum beamforming [59].

2.4.2.3 Minimum variance distortionless response beamforming

Minimum variance distortionless response (MVDR) beamforming is a particular type of filter-and-sum beamforming which looks for minimizing the noise power at its output

under an MSE criterion, while ensuring that the target signal is not distorted. Without loss of generality, the discussion below is particularized to a speech enhancement task. For notational convenience, let us rewrite the additive noise distortion model presented in Section 2.1 as

$$y_n(m) = s_n(m) + \nu_n(m) = \underbrace{a_n s(m-\tau_n)}_{s_n(m)} + \nu_n(m), \tag{2.90}$$

where $y_n(m)$, $s_n(m)$ and $\nu_n(m)$ are the noisy speech, clean speech and noise signals as captured by the n-th sensor of the microphone array. Additionally, $s(m)$ is the speech signal emitted by the source, a_n is a gain factor and, in this case, τ_n represents the time that the signal takes to travel from the source to the array sensor n. Eq. (2.90) can be expressed in the frequency domain as follows:

$$Y_n(f) = S_n(f) + V_n(f) = a_n S(f) e^{-j2\pi f \tau_n} + V_n(f). \tag{2.91}$$

By employing the definition in (2.87) for $\mathbf{y}(f)$ along with

$$\begin{aligned} \mathbf{d}(f) &= \left(a_1 e^{-j2\pi f \tau_1}, ..., a_N e^{-j2\pi f \tau_N}\right)^\top; \\ \boldsymbol{\nu}(f) &= \left(V_1(f), ..., V_N(f)\right)^\top, \end{aligned} \tag{2.92}$$

it is clear that (2.91) can be rearranged in vector notation in the following way:

$$\mathbf{y}(f) = S(f)\mathbf{d}(f) + \boldsymbol{\nu}(f), \tag{2.93}$$

where $\mathbf{d}(f)$ is the so-called steering vector. The steering vector allows us to spatially locate the target source with respect to the microphone array (it also accounts for the different speech gains due to the different amplitudes of the captured speech signals). Then, if we define $\boldsymbol{\Phi}_\nu(f) = \mathrm{E}\left[\boldsymbol{\nu}(f)\boldsymbol{\nu}^H(f)\right]$, MVDR calculates the beamforming weights by solving

$$\begin{gathered} \mathbf{w}(f) = \mathrm{argmin}_{\mathbf{w}(f)}\, \mathbf{w}^H(f)\boldsymbol{\Phi}_\nu(f)\mathbf{w}(f), \\ \text{subject to } \mathbf{w}^H(f)\mathbf{d}(f) = 1. \end{gathered} \tag{2.94}$$

It must be noticed that the constraint $\mathbf{w}^H(f)\mathbf{d}(f) = 1$ ensures a distortionless response for the target signal when applying the weights to (2.93). The optimization problem stated in (2.94) is solved by Lagrange multipliers yielding

$$\mathbf{w}(f) = \frac{\boldsymbol{\Phi}_\nu^{-1}(f)\mathbf{d}(f)}{\mathbf{d}^H(f)\boldsymbol{\Phi}_\nu^{-1}(f)\mathbf{d}(f)}. \tag{2.95}$$

MVDR beamforming is quite sensitive to estimation errors of the noise spatial correlation matrix $\mathbf{\Phi}_\nu(f)$ and the steering vector. In fact, due to a bad estimation of the latter parameter, the MVDR beamformer provided by the organizers of the CHiME-3 Challenge yields a drop in performance regarding a simple multi-condition training. For this reason, a simple delay-and-sum highly outperformed that MVDR beamformer [95]. Nevertheless, this issue was solved by some participants [203, 210] by integrating more robust steering vector estimation methods based on eigenvalue decomposition of the clean speech spatial covariance matrix, which is a popular scheme for this purpose.

The MVDR weights in (2.95) can be rewritten in terms of the noise coherence matrix as

$$\mathbf{w}(f) = \frac{\mathbf{\Gamma}_\nu^{-1}(f)\mathbf{d}(f)}{\mathbf{d}^H(f)\mathbf{\Gamma}_\nu^{-1}(f)\mathbf{d}(f)}. \tag{2.96}$$

Thus, it is interesting to see that an MVDR beamformer operating within an incoherent noise field, i.e. $\mathbf{\Gamma}_\nu(f) = \mathbf{I}_N$, matches delay-and-sum beamforming, that is,

$$\mathbf{w}(f) = \frac{\mathbf{I}_N^{-1}\mathbf{d}(f)}{\mathbf{d}^H(f)\mathbf{I}_N^{-1}\mathbf{d}(f)} = \frac{\mathbf{d}(f)}{\|\mathbf{d}(f)\|^2}. \tag{2.97}$$

On the other hand, when considering a diffuse noise coherence matrix in (2.96), the resulting weights correspond to the so-called superdirective beamformer. Let us define the array gain as the ratio between the SNR at the output of the beamformer and that at the array reference sensor. Then, while MVDR maximizes the array gain for the estimated noise field, superdirective beamforming aims at maximizing the array gain in a diffuse noise field.

All of the previously discussed beamformers are fixed in the sense that their parameters (e.g. weights) do not change along time. In contrast, adaptive beamforming is able to dynamically adapt its parameters to particular acoustic conditions (e.g. type of noise, particular speaker, etc.), which may make it more versatile. Adaptive beamforming is very briefly introduced down below.

2.4.2.4 Adaptive beamforming

Adaptive beamforming is able to dynamically modify the beamformer parameters in order to adapt it to particular acoustic (e.g. noise or speaker) conditions. The most paradigmatic example of this type of beamforming is GSC (Generalized Sidelobe Canceller) [71]. A block diagram of GSC beamforming is depicted in Figure 2.9. This is comprised of two differentiated stages: a fixed beamforming block and an adaptive stage. First, the fixed beamforming block may be implemented as any of the beamformers seen above, e.g. delay-and-sum or MVDR. Second, the adaptive stage is intended

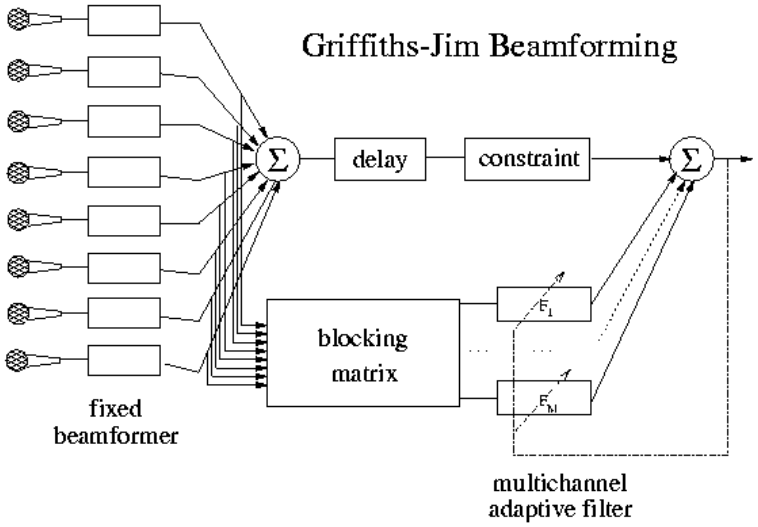

Figure 2.9: Block diagram of the Griffiths-Jim GSC beamformer [42].

to sidelobe cancellation. As we can see from Figure 2.9, a blocking matrix (BM) is placed preceding the adaptive part in order to cancel the target signal in the lower branch of the GSC beamformer. This way, the set of adaptive filters is designed to minimize the power at the output of the beamformer while ensuring that the desired signal is not affected.

A major limitation of adaptive beamforming is that it might lead to target signal cancellation because the BM permits target signal leakage, especially in reverberant environments [132]. This issue has prevented the generalized use of this type of beamforming for speech recognition purposes [160].

2.4.3 Post-filtering

As introduced above, post-filtering refers to the additional processing applied at the output of the beamformer to mitigate its shortfalls, e.g. low directivity at low frequencies, inaccurate estimation of the steering vector and the spatial correlation matrices, inability to remove noise coming from the look direction, etc.

One of the earliest post-filters was the proposed by Zelinski, and it is based on Wiener filtering [132]. This post-filter is computed from the cross-PSDs derived from the different array sensors, in such a way that the knowledge for frequency filtering also includes spatial information. Indeed, this frequency filtering complements and enhances the spatial filtering. Furthermore, it can be shown that concatenating MVDR

beamforming and Wiener post-filtering is equivalent to multi-channel Wiener filtering (MCWF) [94, 103]. Formally, let $\mathbf{w}_{opt}(f) = \mathbf{\Phi}_{YY}^{-1}(f)\boldsymbol{\phi}_{YX}(f)$ be the corresponding MCWF weights (where $\boldsymbol{\phi}_{YX}(f)$ is the cross-PSD vector between the array inputs and the target signal and $\mathbf{\Phi}_{YY}(f)$ is the PSD matrix of the array inputs), it can be shown that these weights can be alternatively expressed as [103]

$$\mathbf{w}_{opt}(f) = \underbrace{\frac{\mathcal{S}_x(f)}{\mathcal{S}_x(f) + \mathcal{S}_n(f)}}_{h_{post}(f)} \cdot \underbrace{\frac{\mathbf{\Phi}_{\nu}^{-1}(f)\mathbf{d}(f)}{\mathbf{d}^H(f)\mathbf{\Phi}_{\nu}^{-1}(f)\mathbf{d}(f)}}_{\text{MVDR beamforming}}, \tag{2.98}$$

where $h_{post}(f)$ is the Wiener post-filter, and $\mathcal{S}_x(f)$ and $\mathcal{S}_n(f)$ are the speech and noise PSDs at the output of the beamformer, respectively.

Later, Marro *et al.* studied in [127] the interaction between the Zelinski post-filter and the beamformer, determining that the former depends on the input SNR as well as on the degree of noise reduction achieved by the beamformer. Then, McCowan proposed in [133] an improvement to the Zelinski post-filter. First, we should recall that the latter method is formulated by assuming incoherent noise (the noise signals from any two different sensors are uncorrelated). In contrast, since this assumption is especially inaccurate for those arrays the sensors of which are close each other, McCowan assumes knowledge of the noise coherence function to estimate the Wiener post-filter from the auto- and cross-PSDs of the microphone array noisy inputs. More precisely, a diffuse noise field model is considered by [133]. In turn, in [103], Lefkimmiatis *et al.* improved the McCowan post-filter by taking advantage of the noise reduction performed by the MVDR beamforming to achieve a more accurate estimation of the noise PSD at the output of the beamformer. Then, this estimation is used to define the Wiener post-filter transfer function.

While the classical Wiener post-filtering approach reviewed above can be useful for speech enhancement, it may not be adequate for noise-robust ASR purposes. For example, in [153], it is shown that the use of Zelinski [118] and Simmer [166] post-filters may yield a drop in performance of the delay-and-sum beamforming when employed for noise-robust ASR on a multi-microphone tablet. Fortunately, there is a number of nice strategies recently explored for multi-channel noise-robust ASR on IMDs. For instance, the aforementioned GEV beamformer of [76] is further improved by a single-channel post-filter called BAN (Blind Analytic Normalization), which is intended to obtain a distortionless response in the target direction. Furthermore, for MVDR, a multi-channel noise reduction (MCNR) post-filter depending on the steering vector, and based on the noise-to-signal plus noise ratio (NSNR), was proposed in [210]. One of its major advantages is that MCNR does not need knowledge of the noise field type. Besides this, several neural network-based post-filters have also been reported in the

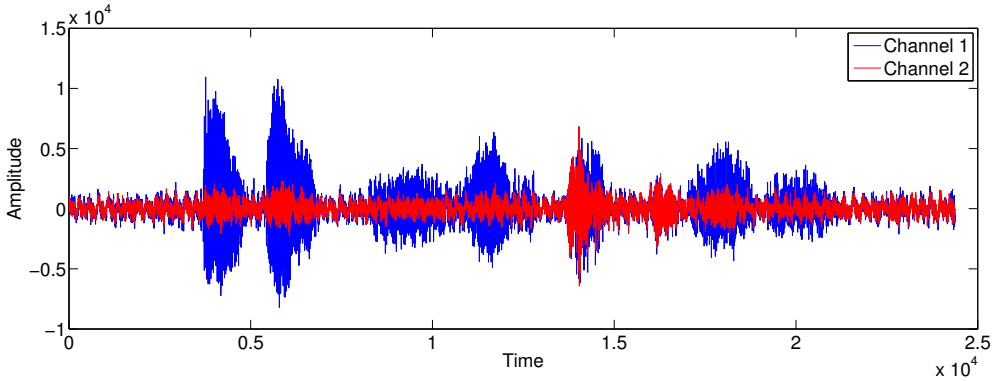

Figure 2.10: Example of noisy speech signal captured with a dual-microphone smartphone employed in close-talk position. Channels 1 and 2 refer to the primary and secondary microphones located at the bottom and rear of the device, respectively.

literature [12, 125, 151]. For example, in [12], a DNN-based spectral mapper is used to predict the clean speech filterbank features from an enhanced version of the beamformer output. Such an enhanced version is computed by means of the application of a model-based source separation mask to the beamformed signal. In [151], a DNN is employed to derive the post-filter weights. To do this, the DNN is fed with the beamformed signal along with an estimation of the residual noise present at the beamformer output. Also in this work, a quite simple but effective post-filter is explored, which is referred to as MaxPower post-filter. It simply consists of backprojecting the beamformer output to the array microphones by using the steering vector. Then, for each T-F bin, the output of the post-filter is the maximum value among the different channels. These post-filters are applied in [151] after three types of GSC beamforming: GSC with sparse BM as well as with ABM (adaptive BM) both using delay-and-sum as fixed beamformer, and GSC with ABM using MVDR as fixed beamformer. The latter proved to be the superior architecture among those evaluated.

2.4.4 Dual-channel power level difference

In this subsection, let us consider a dual-microphone smartphone as the one depicted in Figure 1.2. This device has two sensors, one of them located at its bottom and the other at its rear. These will be referred to as the primary and secondary microphones, respectively. The primary microphone is purposely placed to ensure a direct and short path between the sensor and the speaker's mouth. In other words, the primary microphone is intended to capture the voice of the speaker. On the other hand, because of its

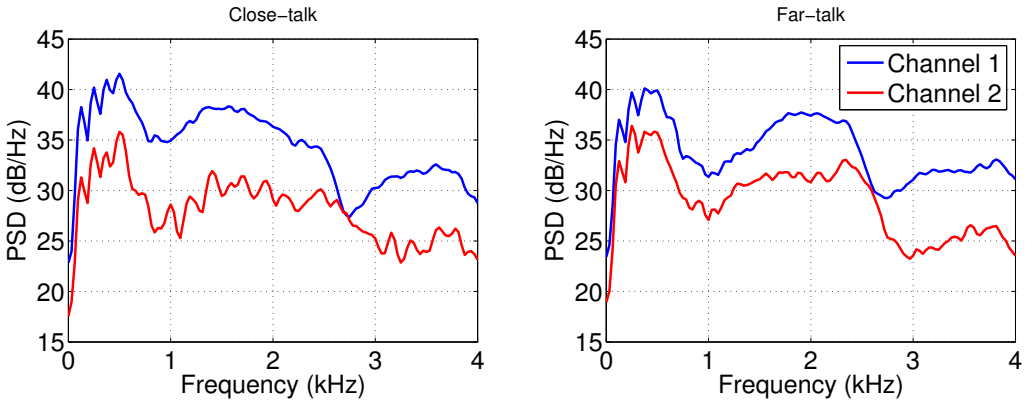

Figure 2.11: Clean speech PSDs obtained from a dual-microphone smartphone employed in close- (left) and far-talk (right) conditions. Channels 1 and 2 refer to the primary and secondary microphones located at the bottom and rear of the device, respectively.

location, the secondary sensor is intended to get information about the acoustic environment more than capturing the speaker's voice. Hence, when the device is employed in close-talk position (i.e. the loudspeaker of the smartphone is placed at the ear of the user), speech is much attenuated at the secondary microphone with respect to the primary one. Coupled with this, it is expected that both microphones observe similar noise signals. This is because it is likely that the device is used within a diffuse noise field and the two microphones are very close each other. Both issues can be observed in the example of Figure 2.10, which represents a noisy speech signal captured by a dual-microphone smartphone used in close-talk position. In addition, to reinforce these ideas as well as for further clarity, Figures 2.11 and 2.12 depict clean speech and car noise PSDs, respectively, derived from the same mobile set-up. In close-talk conditions (left side), it is clear that the clean speech PSD is generally greater at the primary microphone than at the secondary one across the whole frequency range. While the speech PSD level difference between both microphones is smaller in far-talk condition (namely when the user holds the smartphone in one hand at a certain distance from her/his face) than in the close-talk scenario, the secondary sensor is still in an acoustic shadow and the speech PSD is greater at the primary than at the secondary channel. As expected, regardless the use scenario, the noise PSD is quite similar at both channels because of the homogeneity property of diffuse noise fields.

Thus, because there are only two microphones and one of them is located in an acoustic shadow regarding the speaker's mouth, classical beamforming exhibits poor performance when applied to the configuration here described [179, 180]. Therefore, it might be preferable to exploit the speech power level difference (PLD) between the

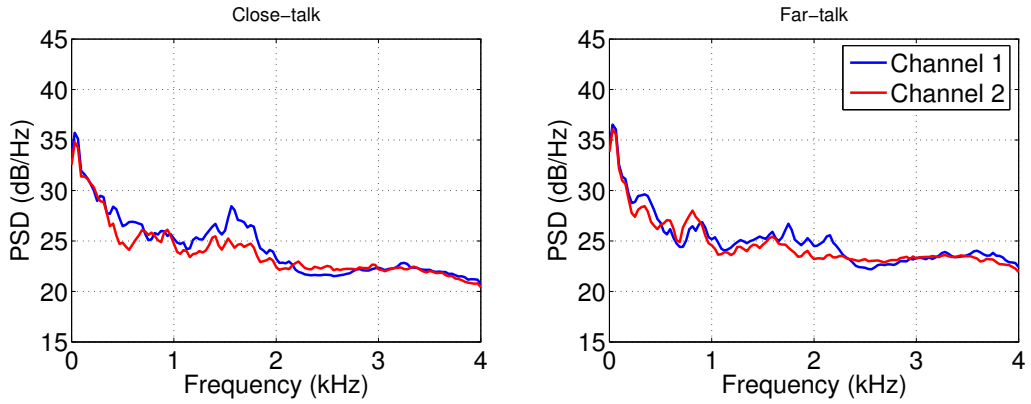

Figure 2.12: Car noise PSDs obtained from a dual-microphone smartphone employed in close- (left) and far-talk (right) conditions. Channels 1 and 2 refer to the primary and secondary microphones located at the bottom and rear of the device, respectively.

two microphones as a principle to design dual-channel algorithms for noise-robust ASR purposes in this context. Indeed, this PLD principle will be considered throughout the next chapters to develop the different contributions in this Thesis.

As can be found in the literature, speech enhancement has taken benefit from the PLD [52, 97, 205, 209]. In particular, the work reported in [97] is broadly representative, where it is proposed a Wiener filter (WF)-based speech enhancement exploiting the information from a dual-microphone smartphone in close-talk position. Additionally in the same work, a dual-channel recursive averaging noise PSD estimator called PLDNE (PLD-based Noise Estimation) is also proposed. This estimator is formulated by assuming both a homogeneous noise field as well as speech is much more attenuated at the secondary microphone than at the primary one. In fact, such assumptions are also employed to compute a WF for enhancement, which is applied to the primary microphone (as the SNR is higher at that sensor than at the secondary one). Evidences on the convenience of following this scheme in comparison with different single-channel strategies are provided in the paper. The same authors extended this work in order to operate in hands-free/far-talk conditions [140]. In this new case, to compute the spectral gain for enhancement, the noise PSD is obtained from a single-channel algorithm based on SPP as well as on a dual-microphone technique exploiting the coherence properties of the speech and noise signals.

Furthermore, the dual-channel speech enhancement method of [209] is highly inspired by [97]. In [209], a measure known as power level ratio (PLR) is considered to obtain a kind of SPP for the primary microphone which is then adjusted by means of a sigmoid function to achieve the spectral gain for enhancement. This method shows

very competitive results in terms of PESQ (Perceptual Evaluation of Speech Quality) with low computational complexity, something very important in portable electronic devices. Moreover, in [52], an inter-microphone noisy speech PSD relation, similar to the PLD, is used to compute an SPP involved in the estimation of a spatial noise correlation matrix. Such a matrix is then used in an MVDR filter applied to enhance the noisy speech captured by the dual-microphone smartphone. To conclude, we could refer the case of [26], where the PLD principle is exploited to design a voice activity detector (VAD) for dual-channel mobile phones. This technique proves to outperform conventional single-microphone VAD methods, as well as the dual-microphone ones based on the well-known magnitude-squared coherence [23].

2.5 Summary

As we know, the performance of every ASR system is severely degraded when there exists mismatch between the training and testing conditions. There is a number of mismatch sources, and one of the most significant is background noise because of its pervasiveness. Because providing robustness against noise in ASR on IMDs is the scope of this Thesis, in this chapter we have introduced the foundations of the noise-robust processing for both single- and multi-channel ASR. This has served as a presentation of the theoretical basis from which the different noise-robust contributions of this Thesis are built in later chapters.

In the first instance, it was explained the general speech distortion modeling considered as the basic mathematical framework both to review noise-robust state-of-the-art approaches and to develop our contributions throughout the following chapters. Along with this, we analyzed how the statistical distribution of the speech energy is altered in the presence of ambient noise.

Then, the single-channel robust speech recognition fundamentals were presented. This was done through revisiting some of the most relevant noise-robust approaches, according to our purposes, categorized into four different classes: feature-space approaches, model-based approaches, distortion modeling by vector Taylor series (VTS) and missing-data approaches. Moreover, we highlighted different advantages and drawbacks of these four classes, so the selection of the most adequate approach in each case depends on the ASR use scenario. For example, we mentioned that, in contrast to the feature-space approaches, the model-based methods are characterized by relatively high computational complexity. However, as an advantage, the latter methods are often more robust against speech distortions. Also, we saw that the VTS strategy, which may be used for either feature enhancement or model adaptation, is more accurate

for the latter purpose than the classical model adaptation techniques because VTS employs a physical model that explicitly explains how it is the non-linear interaction between the speech signal and the environmental distortions. Regarding the missing-data approaches, it was argued that a critical part of those methods is the estimation of masks identifying the unreliable spectro-temporal regions of the noisy speech signal.

For some noise-robust methods, a module estimating the background noise that contaminates speech is required (e.g. for Wiener filtering or VTS feature enhancement). Indeed, the performance of those methods often relies on the accuracy of such noise estimates. On this basis, despite the attention moved over the last years towards other noise-robust solutions which do not require any explicit noise estimation, we briefly reviewed some prominent classical noise estimation techniques as this issue is still important.

In the second part of this chapter we focused on multi-channel robust speech processing applied to IMDs. We stated that multi-channel robust speech processing has gained popularity over the last years due to both its potential regarding the single-channel solutions and the decrease in the price of hardware. First, we provided an overview of multi-channel robust ASR on IMDs. We focused on the recent CHiME challenges devoted to noise-robust ASR on a multi-microphone tablet and we observed that an outstanding recognition performance is achieved by combining single- and multi-channel algorithms. At this respect, a widely used multi-channel scheme consists of microphone array processing followed by some kind of post-filtering to overcome the shortcomings of beamforming. Therefore, since beamforming is a fundamental pillar of multi-channel robust ASR, some of its basics were presented along with the main noise fields. Next, we have commented the most well-known fixed beamformers, namely delay-and-sum and MVDR, as well as adaptive array processing. Then, we reviewed different classical Wiener post-filters along with the most recent approaches specifically intended to multi-channel noise-robust ASR. To conclude, the dual-channel power level difference (PLD) principle was presented. As we saw, this principle explains the spatial singularities of the speech and noise signals in a dual-microphone IMD set-up. Essentially, we determined that clean speech energy is greater at the primary than at the secondary microphone, while similar noise PSDs are observed by both microphones. Finally, we pointed out that this knowledge will be taken into account when developing our contributions as beamforming presents severe limitations in this context.

CHAPTER 3

Multi-Channel Power Spectrum Enhancement

As it has been presented in Chapter 2, one of the most popular approaches over the years to provide robustness in automatic speech recognition (ASR) is feature enhancement, which can be considered a subcategory of the feature-space approaches. While feature enhancement has been extensively studied within a single-channel context, the same progress has not been made for a multi-channel framework. Although we can find specific multi-channel feature enhancement techniques such as multi-channel Wiener filtering (MCWF), most of the robust solutions in this topic consists of the concatenation of beamforming and single-channel feature enhancement methods which behave as a sort of post-filter. Even in the case of MCWF, we should remind that this technique can be decomposed into the concatenation of MVDR (Minimum Variance Distortionless Response) beamforming and single-channel Wiener filtering. It becomes clear that single-channel feature enhancement methods cannot properly exploit all the existent particularities in a multi-channel framework, so its potentials are wasted.

This chapter details our contributions to multi-channel feature enhancement by presenting three power spectrum enhancement methods which specifically take advantage of the spatial properties of speech and noise in a dual-microphone configuration. As it was introduced in the previous chapters, the dual-microphone configuration is especially interesting since it can be widely found in the latest mobile devices, where the main purpose of the secondary microphone is to get clearer information about the acoustic environment than the primary one. Hence, our power spectrum enhancement methods take benefit from this characteristic to circumvent the limitations of the single-channel feature enhancement methods when applied to such a dual-microphone set-up.

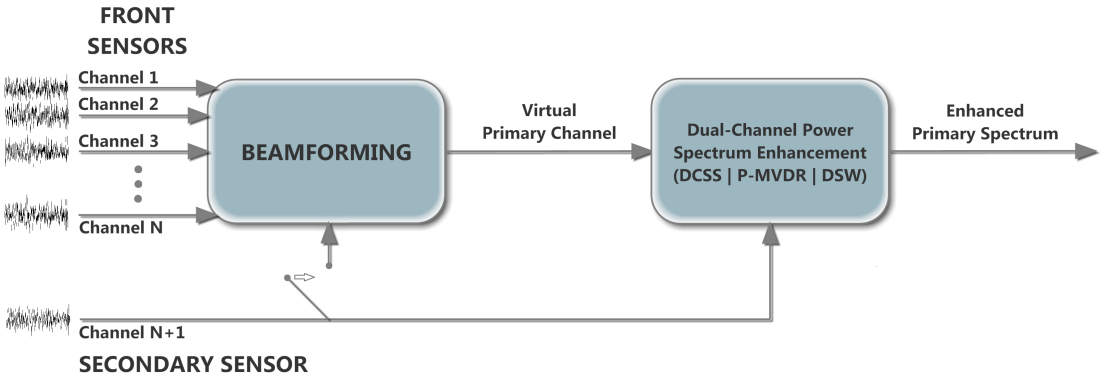

Figure 3.1: Power spectrum enhancement scheme followed when the IMD has more than one front sensor.

The contributions developed in this chapter have been denominated as DCSS (Dual-Channel Spectral Subtraction), P-MVDR (Power-MVDR) and DSW (Dual-channel Spectral Weighting, based on Wiener filtering). It will be shown that all of these techniques require knowledge of the relative speech gain (RSG) which relates the clean speech power spectra at both channels. To obtain this parameter, a two-channel minimum mean square error (MMSE)-based estimator is also developed for this task in this Thesis and presented in Section 3.4.

It has to be mentioned that, without loss of generality, several issues will be illustrated throughout this chapter by considering a dual-microphone smartphone employed in either close- or far-talk conditions. It is considered that this set-up constitutes a perfect scenario (as well as realistic and common) to exemplify different concepts, avoiding at the same time distractions coming from other more complex configurations.

3.1 Combinatorial strategy

Apart from embedding a microphone whose main mission is capturing information about the acoustic environment, an intelligent mobile device (IMD) can easily integrate more sensors that can be exploited to provide further ASR robustness. These sensors are most of the time comparable to a primary microphone in case they are oriented towards the speaker (front sensors). Since our methods are intended to take benefit from a secondary sensor under the explained dual-channel framework, an additional procedure should be established to be followed in the case of more than two sensors were present in an IMD. Thus, we found it useful to follow the strategy considered in the recent multi-channel noise-robust ASR literature consisting of the application of

microphone array pre-processing (and, more in particular, beamforming) to compute a related virtual primary channel to be used along with the secondary one by our dual-channel techniques. In such a case, the contributions developed in the following behave as post-filters. It must be remarked that beamforming for virtual primary channel computation may be applied by either considering all the microphones in the IMD (i.e. by also integrating the secondary sensor) or only the front ones, depending on what is more advantageous in terms of ASR performance. Apart from providing a single (virtual) primary channel, the use of beamforming presents an additional advantage, which is that this virtual primary channel has a higher SNR (Signal-to-Noise Ratio) than any other signal coming from a particular microphone in the IMD, being generally increased the recognition accuracy of the ASR system. A block diagram showing this combinatorial strategy can be seen in Figure 3.1, where an IMD has $N+1$ microphones: N front microphones plus a secondary one.

3.2 Basic approaches

The two basic dual-channel power spectrum enhancement approaches DCSS and P-MVDR are formulated in this section. On the one hand, DCSS is based on spectral subtraction (SS), while P-MVDR is a power spectrum enhancement method based on MVDR which discards the phase information to overcome the limitations of classical MVDR beamforming when applied on the presented dual-microphone set-up.

Hereinafter it is considered a simplified version of the distortion model developed in Section 2.1 in such a manner that the convolutive distortion is not taken into account. Nevertheless, the latter type of distortion could be mitigated in a subsequent feature normalization front-end stage that might be implemented, for instance, as CMN (Cepstral Mean Normalization). Hence, let us now suppose a noisy speech signal $y_k(m)$ which can be expressed as the sum of a clean speech signal $x_k(m)$ and a noise $n_k(m)$, i.e. $y_k(m) = x_k(m) + n_k(m)$, where $k = 1, 2$ indicates the microphone that captures the signal, namely the primary and secondary ones, respectively[1]. Assuming that speech and noise are independent, this additive model can be expressed in terms of power spectra as,

$$|Y_1(f,t)|^2 = |X_1(f,t)|^2 + |N_1(f,t)|^2; \tag{3.1}$$

$$|Y_2(f,t)|^2 = |X_2(f,t)|^2 + |N_2(f,t)|^2, \tag{3.2}$$

where $f = 0, 1, ..., \mathcal{M} - 1$ and $t = 0, 1, ..., T - 1$ denote the frequency bin and time frame indices, respectively, $\mathcal{M}$ is the total number of linear frequency bins and T is

[1]In the case of multiple front sensors, $k = 1$ refers to the virtual primary channel obtained by means of beamforming.

the number of frames in an utterance. As the primary microphone is faced towards the speaker, it is assumed that the signal captured by this microphone has a higher SNR than the signal captured by the secondary sensor and, hence, the objective of both DCSS and P-MVDR is to provide an estimate of the clean speech power spectrum at the primary channel, $|\hat{X}_1(f,t)|^2$, by taking advantage of the dual-channel information.

3.2.1 Dual-channel spectral subtraction

To formulate this alternative spectral subtraction (SS) approach, it is first assumed that the clean speech power at the secondary channel is related with the clean speech power at the first one through the so-called relative speech gain (RSG) term $\mathcal{A}_{21}(f,t)$, i.e. $|X_2(f,t)|^2 = \mathcal{A}_{21}(f,t)|X_1(f,t)|^2$. This factor can be interpreted as the target speech signal transfer function between the two microphones. In this way, (3.2) can be rewritten as,

$$|Y_2(f,t)|^2 = \mathcal{A}_{21}(f,t)|X_1(f,t)|^2 + |N_2(f,t)|^2. \tag{3.3}$$

Along with the relationship defined in (3.3) we can also relate the noise power spectra at the first and secondary channels as $|N_1(f,t)|^2 = G_{12}(f,t)|N_2(f,t)|^2$. In such a case, (3.1) can be rewritten as,

$$|Y_1(f,t)|^2 = |X_1(f,t)|^2 + G_{12}(f,t)|N_2(f,t)|^2. \tag{3.4}$$

Similarly to $\mathcal{A}_{21}(f,t)$, factor $G_{12}(f,t)$ can be understood as the frequency response of a new linear filter that relates the noise signals captured by the two available sensors. As it has been explicitly remarked, both $\mathcal{A}_{21}(f,t)$ and $G_{12}(f,t)$ are time-dependent. On the one hand, for example, $\mathcal{A}_{21}(f,t)$ might change over time due to variations on the relative location between the speaker and the microphones or on the environment acoustics. Different approaches can be considered for the computation of this term. One way to calculate the RSG may be through the square of the ratio between the speech gains for the secondary and primary sensors coming from the steering vector used in beamforming. Nevertheless, an alternative MMSE-based method is developed for this purpose in Section 3.4. On the other hand, $G_{12}(f,t)$ is subject to changes from variations on the relative position between the noise sources and the microphones as well as due to other environmental acoustic factors. A straightforward approach to obtain $G_{12}(f,t)$ is explained below.

By combining Equations (3.4) and (3.3) we obtain the following dual-channel spectral subtraction (DCSS) estimator for every frequency bin f and time frame t:

$$|\hat{X}_1(f,t)|^2 = \frac{|Y_1(f,t)|^2 - G_{12}(f,t)|Y_2(f,t)|^2}{1 - G_{12}(f,t)\mathcal{A}_{21}(f,t)}. \tag{3.5}$$

Then, the result in (3.5) is bounded below in order to avoid any possible negative power spectrum bin:

$$|X_1'(f,t)|^2 = \max\left(|\hat{X}_1(f,t)|^2, \eta|Y_1(f,t)|^2\right), \tag{3.6}$$

where η is a thresholding factor.

For every frequency bin f the relative noise gain factor $G_{12}(f,t)$ can be estimated by minimizing the following mean square error (MSE):

$$E_f = \mathrm{E}\left[\left(|N_1(f,t)|^2 - G_{12}(f,t)|N_2(f,t)|^2\right)^2\right]. \tag{3.7}$$

Let us first define

$$\phi_{N,f,t}(k,l) = \mathrm{E}\left[|N_k(f,t)|^2|N_l(f,t)|^2\right], \quad k,l = 1,2, \tag{3.8}$$

which is a noise cross-correlation coefficient. By solving $\partial E_f/\partial G_{12}(f,t) = 0$, we can derive the desired estimate as,

$$\hat{G}_{12}(f,t) = \frac{\phi_{N,f,t}(1,2)}{\phi_{N,f,t}(2,2)}. \tag{3.9}$$

Figure 3.2 depicts the actual $G_{12}(f,t)$ factor over time, at two different frequency bins, obtained when recording bus noise with a dual-microphone smartphone employed in close- and far-talk conditions. A great variety of real-life acoustic environments presents the characteristic of homogeneity [182], i.e. same sound PSD (Power Spectral Density) at any point of the space. This means that it would be typical to find $G_{12}(f,t)$ values around 1. Nevertheless, while the actual $G_{12}(f,t)$ curves plotted lie somewhere at around 1, they are far from being the unit due to different reasons. One of the most important is the distinct frequency responses that the sensors exhibit. In particular, the primary microphone often presents a better frequency characteristic with a higher sensitivity than the secondary one, in such a way that it is likely to observe, depending on the frequency bin, $G_{12}(f,t)$ values greater than 1. However, this issue is also conditioned by the acoustic shadows that can be found in a mobile device usage scenario. Thus, for instance, when using a dual-microphone smartphone in close-talk position, the head of the speaker casts a stronger shadow over the primary microphone than in the far-talk case. Hence, in close-talk conditions the greater sensitivity of the primary microphone is offset by this fact and $G_{12}(f,t)$ might decrease its magnitude when compared with far-talk conditions. Finally, it should be noticed that even in the presence of an ideal homogeneous noise field, all the factors explained above plus many others would make $G_{12}(f,t)$ likely to differ from the unit.

A straightforward possible approach to be used in practice for the estimation of the cross-correlation coefficient $\phi_{N,f,t}(k,l)$ $(k,l = 1,2)$ and, therefore, for the estimation

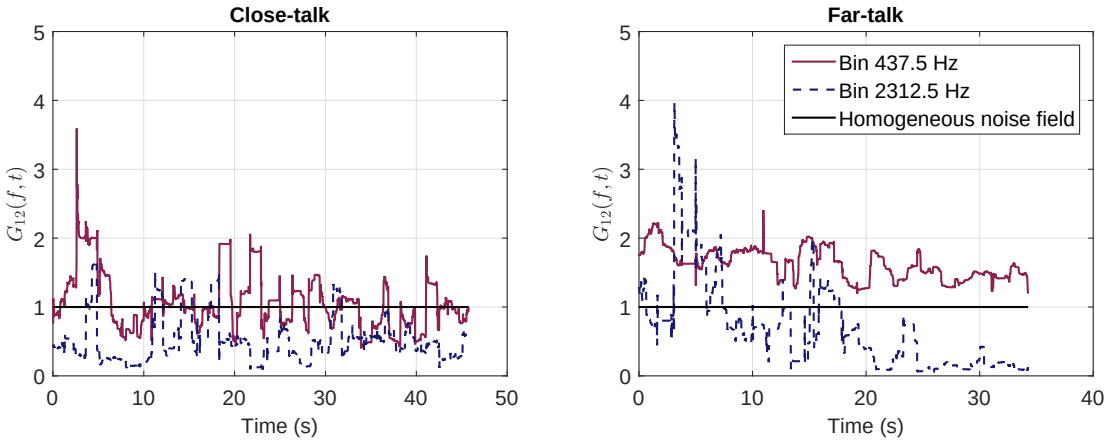

Figure 3.2: Actual $G_{12}(f,t)$ over time, at two different frequency bins, obtained when recording bus noise with a dual-microphone smartphone employed in close- (left) and far-talk conditions (right). $G_{12}(f,t)$ as in a homogeneous noise field is also represented as a reference.

of the relative noise gain factor, $G_{12}(f,t)$, is the following. By assuming that the first and last M frames of every noisy speech utterance are only noise, $\phi_{N,f,t}(k,l)$ can be computed on an utterance basis per frequency bin f as the sample cross-correlation over those frames. Indeed, this will be the approach considered for this Thesis.

3.2.2 Power-MVDR

Power-MVDR also estimates the clean speech power spectrum at the primary channel and it is inspired by the MVDR beamformer [79]. P-MVDR discards the phase information in order to overcome the limitations of the classical MVDR beamforming when applied to the considered dual-channel framework. According to this, the proposed linear estimator of the clean speech power spectrum at frequency bin f and time frame t at the primary channel can be expressed as,

$$|\hat{X}_1(f,t)|^2 = \mathbf{w}_{f,t}^{\top}\mathbf{y}(f,t) = \mathbf{w}_{f,t}^{\top}\begin{pmatrix} |Y_1(f,t)|^2 \\ |Y_2(f,t)|^2 \end{pmatrix}, \tag{3.10}$$

where $\mathbf{w}_{f,t}$ is a 2×1 weighting vector that must be estimated from the dual signal. By reusing Eq. (3.3), the estimator in (3.10) can now be expressed as,

$$|\hat{X}_1(f,t)|^2 = \mathbf{w}_{f,t}^{\top}|X_1(f,t)|^2\begin{pmatrix} 1 \\ \mathcal{A}_{21}(f,t) \end{pmatrix} + \mathbf{w}_{f,t}^{\top}\begin{pmatrix} |N_1(f,t)|^2 \\ |N_2(f,t)|^2 \end{pmatrix}. \tag{3.11}$$

Our goal now is to obtain the weighting vector that minimizes the mean square noise-dependent term in (3.11), that is,

$$\mathbf{w}_{f,t} = \underset{\mathbf{w}_{f,t}}{\operatorname{argmin}} \, \mathrm{E}\left[\left(\mathbf{w}_{f,t}^{\top}\boldsymbol{\nu}(f,t)\right)^{2}\right], \tag{3.12}$$

where $\boldsymbol{\nu}(f,t) = (|N_1(f,t)|^2, |N_2(f,t)|^2)^{\top}$. By rearranging terms, (3.12) is now rewritten as,

$$\begin{aligned} \mathbf{w}_{f,t} &= \operatorname{argmin}_{\mathbf{w}_{f,t}} \mathbf{w}_{f,t}^{\top}\mathrm{E}\left[\boldsymbol{\nu}(f,t)\boldsymbol{\nu}(f,t)^{\top}\right]\mathbf{w}_{f,t} \\ &= \operatorname{argmin}_{\mathbf{w}_{f,t}} \mathbf{w}_{f,t}^{\top}\boldsymbol{\Phi}_N(f,t)\mathbf{w}_{f,t}, \end{aligned} \tag{3.13}$$

in which $\boldsymbol{\Phi}_N(f,t)$ is the 2×2 noise spatial correlation matrix

$$\boldsymbol{\Phi}_N(f,t) = \begin{pmatrix} \phi_{N,f,t}(1,1) & \phi_{N,f,t}(1,2) \\ \phi_{N,f,t}(2,1) & \phi_{N,f,t}(2,2) \end{pmatrix}, \tag{3.14}$$

where $\boldsymbol{\Phi}_N(f,t) = \boldsymbol{\Phi}_N^{\top}(f,t)$. Similarly to the case of DCSS, $\phi_{N,f,t}(k,l)$ is the noise power correlation between channels k and l $(k,l = 1,2)$. In order to build the noise spatial correlation matrix of (3.14), the cross-correlation coefficients are obtained as the sample cross-correlation computed from the first and last M frames of each utterance as in DCSS. The minimization in (3.13) must be subject to the distortionless speech constraint

$$\mathbf{w}_{f,t}^{\top}\begin{pmatrix} 1 \\ \mathcal{A}_{21}(f,t) \end{pmatrix} = 1 \tag{3.15}$$

and, therefore, the optimization problem can be solved by Lagrange multipliers considering the cost function $\mathcal{L}(\mathbf{w}_{f,t},\lambda) = \mathbf{w}_{f,t}^{\top}\boldsymbol{\Phi}_N(f,t)\mathbf{w}_{f,t} - \lambda\left(\mathbf{w}_{f,t}^{\top}(1,\mathcal{A}_{21}(f,t))^{\top} - 1\right)$. We can obtain the optimal weighting vector by solving $\nabla\mathcal{L}(\mathbf{w}_{f,t},\lambda) = 0$. Since the resolution procedure is completely analogous to that of MVDR beamforming, this is not detailed here. Therefore, the final weighting vector results,

$$\mathbf{w}_{f,t} = \frac{\boldsymbol{\Phi}_N^{-1}(f,t)\,(1,\mathcal{A}_{21}(f,t))^{\top}}{(1,\mathcal{A}_{21}(f,t))\,\boldsymbol{\Phi}_N^{-1}(f,t)\,(1,\mathcal{A}_{21}(f,t))^{\top}}. \tag{3.16}$$

Once the weighting vector $\mathbf{w}_{f,t}$ has been applied to the dual-channel noisy observation, the estimate in (3.10) must also be bounded below as in (3.6) in order to prevent negative power spectrum bins. As can be noted, Eq. (3.16) has the same form as the MVDR beamforming weighting vector but in the linear power spectral domain, where $(1,\mathcal{A}_{21}(f,t))^{\top}$ plays a similar role to a steering vector referred to the primary microphone.

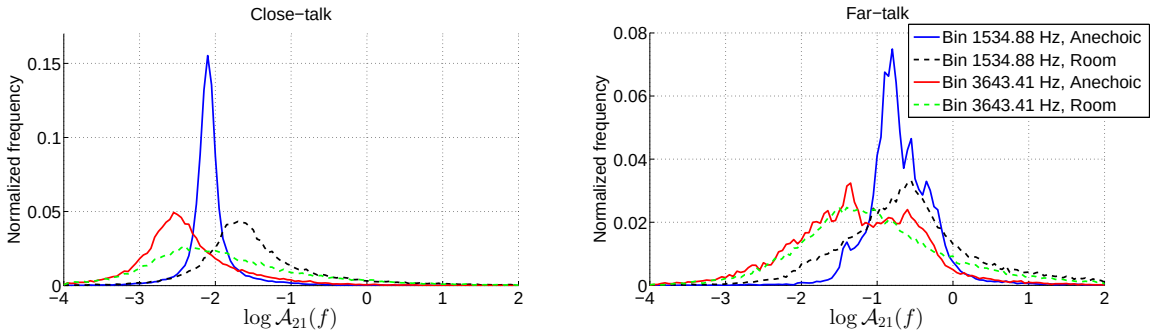

Figure 3.3: Example of $\log \mathcal{A}_{21}(f)$ histograms, at two different frequency bins, obtained for a dual-microphone smartphone employed in close- (left) and far-talk conditions (right). Two different acoustic environments are considered: an anechoic chamber and a small furnished room.

3.2.3 Performance analysis

It is worth examining both DCSS and P-MVDR from a common perspective. Thus, we can realize that both methods operate according to a linear combination of the dual-channel noisy observation. First, by assuming that a pair of weights $\{w_{f,t}^{(1)}, w_{f,t}^{(2)}\}$ is available at each frequency bin f and time frame t, we can express the clean speech power spectrum bin estimate for both DCSS and P-MVDR as,

$$|\hat{X}_1(f,t)|^2 = w_{f,t}^{(1)}|Y_1(f,t)|^2 + w_{f,t}^{(2)}|Y_2(f,t)|^2. \tag{3.17}$$

For DCSS it can be shown (Eqs. (3.5) and (3.9)) that those weights are,

$$\begin{aligned} w_{f,t}^{(1),\text{DCSS}} &= \frac{\phi_{N,f,t}(2,2)}{\phi_{N,f,t}(2,2) - \mathcal{A}_{21}(f,t)\phi_{N,f,t}(1,2)}; \\ w_{f,t}^{(2),\text{DCSS}} &= -\frac{\phi_{N,f,t}(1,2)}{\phi_{N,f,t}(2,2) - \mathcal{A}_{21}(f,t)\phi_{N,f,t}(1,2)}. \end{aligned} \tag{3.18}$$

On the other hand, by expanding (3.16) as well as considering the symmetry of the noise spatial correlation matrix $\mathbf{\Phi}_N(f,t)$ (i.e. $\phi_{N,f,t}(2,1) = \phi_{N,f,t}(1,2)$), it is straightforward to derive the following expressions for the couple of weights obtained with P-MVDR:

$$\begin{aligned} w_{f,t}^{(1),\text{P-MVDR}} &= \frac{\phi_{N,f,t}(2,2) - \mathcal{A}_{21}(f,t)\phi_{N,f,t}(1,2)}{\phi_{N,f,t}(2,2) - 2\mathcal{A}_{21}(f,t)\phi_{N,f,t}(1,2) + \mathcal{A}_{21}^2(f,t)\phi_{N,f,t}(1,1)}; \\ w_{f,t}^{(2),\text{P-MVDR}} &= \frac{\mathcal{A}_{21}(f,t)\phi_{N,f,t}(1,1) - \phi_{N,f,t}(1,2)}{\phi_{N,f,t}(2,2) - 2\mathcal{A}_{21}(f,t)\phi_{N,f,t}(1,2) + \mathcal{A}_{21}^2(f,t)\phi_{N,f,t}(1,1)}. \end{aligned} \tag{3.19}$$

Before proceeding, let us take a look at Figure 3.3, which shows examples of $\log \mathcal{A}_{21}(f)$ histograms for a dual-microphone smartphone employed in close- (left) and far-talk conditions (right). The natural logarithm to the $\mathcal{A}_{21}(f,t)$ samples has been applied when computing these histograms for visual clarity. Additionally, notice that in these examples $\mathcal{A}_{21}(f,t)$ at every time frame t has been considered a realization of the variable given a particular frequency bin f. The depicted histograms correspond to the same two different frequency bins in two acoustic environments: an anechoic chamber and a small furnished room. First of all, it is noticeable that the probability mass of $\log \mathcal{A}_{21}(f)$ is wider in a far- than in a close-talk configuration. This can mainly be explained by a greater variability in the relative position between the speaker and the mobile device in far-talk conditions. Moreover, wider probability mass can also be observed for the small furnished room with respect to the anechoic chamber due to the acoustic variability of the former environment, which is more complex. Finally, it should be noticed that the probability mass in far-talk conditions is shifted to the right on the horizontal axis with respect to close-talk conditions. This indicates that the relative speech energy attenuation between the two sensors of the mobile device is greater in close-talk position than in a far-talk scenario. In this way, it is expected that more information about the adverse acoustic environment can be obtained from the secondary sensor in close- than in far-talk conditions, making the latter a more challenging problem.

Therefore, in accordance with both Figure 3.3 and the discussion presented in the above paragraph, $\mathcal{A}_{21}(f,t)$ often has, relatively, a small magnitude, especially in close-talk conditions. From (3.18) and (3.19), it is evident that as $\mathcal{A}_{21}(f,t) \to 0$ the couple of weights for both DCSS and P-MVDR tend to be the same. Particularly, into the limit,

$$\begin{aligned} \lim_{\mathcal{A}_{21}(f,t)\to 0} w_{f,t}^{(1),\text{DCSS}} &= \lim_{\mathcal{A}_{21}(f,t)\to 0} w_{f,t}^{(1),\text{P-MVDR}} = 1; \\ \lim_{\mathcal{A}_{21}(f,t)\to 0} w_{f,t}^{(2),\text{DCSS}} &= \lim_{\mathcal{A}_{21}(f,t)\to 0} w_{f,t}^{(2),\text{P-MVDR}} = -\frac{\phi_{N,f,t}(1,2)}{\phi_{N,f,t}(2,2)}. \end{aligned} \tag{3.20}$$

In fact, when $\mathcal{A}_{21}(f,t) = 0$ speech is absent at the secondary channel and, therefore (for both DCSS and P-MVDR),

$$\begin{aligned} w_{f,t}^{(2)}|Y_2(f,t)|^2 &= -\frac{\phi_{N,f,t}(1,2)}{\phi_{N,f,t}(2,2)}|N_2(f,t)|^2 \\ &= -\hat{G}_{12}(f,t)|N_2(f,t)|^2 \\ &= -|\hat{N}_1(f,t)|^2. \end{aligned} \tag{3.21}$$

Under these conditions, the secondary channel gives a noise estimate for the primary channel and $w_{f,t}^{(1)} = 1$, so that (3.17) becomes a classical spectral subtraction, i.e. $|\hat{X}_1(f,t)|^2 = |Y_1(f,t)|^2 - \hat{G}_{12}(f,t)|N_2(f,t)|^2 = |Y_1(f,t)|^2 - |\hat{N}_1(f,t)|^2$.

Hence, we can conclude that a very similar performance of both DCSS and P-MVDR can be expected when $\mathcal{A}_{21}(f,t) \to 0$, i.e. in close-talk conditions. On the contrary, in a far-talk scenario the clean speech energy at the secondary sensor may sometimes be comparable to that captured by the primary sensor, that is $\mathcal{A}_{21}(f,t) \simeq 1$. This mainly occurs because of the following concurring two reasons: *1)* the mobile device is located in front of the speaker within the direct clean speech acoustic path, and *2)* the acoustic environment produces reflections of the clean speech signal that increase the clean speech energy at the secondary microphone along with the speech diffraction at the borders of the device. Of course, there also exist reflections and diffraction of the clean speech signal in a close-talk scenario. However, in the latter situation the mobile device (and, therefore, the secondary microphone) looks towards a direction which is approximately orthogonal to that the voice is projected by the speaker, so little clean speech energy is received at the secondary microphone by reflections or diffraction. As the magnitude of $\mathcal{A}_{21}(f,t)$ starts to increase, DCSS and P-MVDR show a different performance (i.e. when applied to far-talk conditions).

A relevant weakness of DCSS comes from the numerical instability when $\mathcal{A}_{21}(f,t) \to \phi_{N,f,t}(2,2)/\phi_{N,f,t}(1,2) = \hat{G}_{12}^{-1}(f,t)$. From (3.18), we can see that in such a situation $\{w_{f,t}^{(1),\mathrm{DCSS}}, w_{f,t}^{(2),\mathrm{DCSS}}\} \to \{+\infty, -\infty\}$. For example, in the context of a homogeneous noise field, we can expect $\hat{G}_{12}(f,t) \approx 1$ since almost the same noise PSD is observed by both microphones [205]. If the clean speech power at both sensors is comparable (as could especially happen in far-talk conditions) and, therefore, $\mathcal{A}_{21}(f,t) \to 1$, the numerical instability might appear. On the other hand, in P-MVDR, the denominator, according to (3.19), is only zero for

$$\mathcal{A}_{21}^{*}(f,t) = \frac{\phi_{N,f,t}(1,2)}{\phi_{N,f,t}(1,1)} \pm \sqrt{\frac{\phi_{N,f,t}(2,2)}{\phi_{N,f,t}(1,1)}}\sqrt{\rho_{N,f,t}^2 - 1}, \tag{3.22}$$

where

$$\rho_{N,f,t} = \frac{\phi_{N,f,t}(1,2)}{\sqrt{\phi_{N,f,t}(1,1)\phi_{N,f,t}(2,2)}} \tag{3.23}$$

is the Pearson's correlation coefficient (i.e. $\rho_{N,f,t} \in [-1,1]$). Unless $\rho_{N,f,t} = \pm 1$, it is evident that the imaginary part of (3.22) is non-zero and, therefore, $\mathcal{A}_{21}^{*}(f,t) \in \mathbb{C}$. Since actually $\mathcal{A}_{21}(f,t) \in [0,+\infty)$, a numerical instability problem as in DCSS could hardly appear in P-MVDR. In this regard, we can conclude that P-MVDR is a more robust method than DCSS.

3.3 Dual-channel spectral weighting based on Wiener filtering

For this third power spectrum enhancement contribution, a dual-channel spectral weighting approach based on Wiener filtering (widely used by many speech enhancement methods [19, 115, 170]) is adopted. To estimate the *a priori* SNR required by the WF, it is initially assumed in Subsection 3.3.1 that the secondary microphone captures no speech and that noise acquired by this microphone coincides with the one captured by the primary microphone. Although these assumptions can be acceptable in some situations (e.g., as we have seen, when the mobile device is used in close-talk position and within a homogeneous noise field [97, 205]), in general, they will not be satisfied. Hence, two modifications to the basic WF weighting are introduced, which are intended to overcome the lack of realism of these two initial assumptions. First, a bias correction term is introduced in Subsection 3.3.2 to rectify the resulting spectral weights when a non-negligible speech component is present at the secondary channel. Second, we develop a noise equalization procedure in Subsection 3.3.3 to be applied on the secondary channel before spectral weight computation to make the noise PSDs at both channels similar. Finally, a spectral weight post-processing and an overview of the full enhancement system are provided in Subsection 3.3.4.

Firstly, it must be stated that the same dual-channel additive noise distortion model in the power spectral domain as presented at the beginning of Section 3.2 is considered from now onwards. As it is well-known, the Wiener filter (WF) is optimal in the sense of minimizing the MSE between the target signal and the estimated one given the input corrupted signal. Under our framework, the desired optimal non-causal filter in the frequency domain is given by [114]

$$H_1(f,t) = \frac{\mathcal{S}_{x_1}(f,t)}{\mathcal{S}_{x_1}(f,t) + \mathcal{S}_{n_1}(f,t)}, \tag{3.24}$$

where $\mathcal{S}_{x_1}(f,t)$ and $\mathcal{S}_{n_1}(f,t)$ are the PSDs of the clean speech and the noise, respectively, at the primary channel. Thus, the clean speech power spectrum bin $|X_1(f,t)|^2$ can be estimated as

$$|\hat{X}_1(f,t)|^2 = H_1^2(f,t)|Y_1(f,t)|^2. \tag{3.25}$$

It should be noted that $H_1^2(f,t) \in [0,\ 1]$ may be seen as a WF-based spectral weight such that $H_1^2(f,t) \to 1$ ($H_1^2(f,t) \to 0$) if speech (noise) dominates. The WF can be alternatively expressed as

$$H_1(f,t) = \frac{\xi_1(f,t)}{\xi_1(f,t) + 1}, \tag{3.26}$$

where

$$\xi_1(f,t) = \frac{\mathcal{S}_{x_1}(f,t)}{\mathcal{S}_{n_1}(f,t)} \tag{3.27}$$

is the *a priori* SNR of the primary channel, which can be alternatively expressed as

$$\xi_1(f,t) = \frac{\mathcal{S}_{y_1}(f,t) - \mathcal{S}_{n_1}(f,t)}{\mathcal{S}_{n_1}(f,t)} \tag{3.28}$$

under our additive noise distortion model, namely,

$$\mathcal{S}_{y_k}(f,t) = \mathcal{S}_{x_k}(f,t) + \mathcal{S}_{n_k}(f,t) \quad (k=1,2). \tag{3.29}$$

A straightforward single-channel approach for obtaining $\xi_1(f,t)$ consists of directly estimating the noise PSD $\mathcal{S}_{n_1}(f,t)$ from signal $y_1(m)$ [45]. As we know, this is often a difficult task since speech and noise overlap. In the following, we will alternatively exploit the spatial characteristics of speech and noise under the considered dual-microphone set-up to obtain estimates of $\xi_1(f,t)$ from the available signals $y_1(m)$ and $y_2(m)$.

3.3.1 Biased spectral weight estimation

Previous work on dual-channel noise reduction has shown that, when a mobile device is used in close-talk position, the clean speech PSD is considerably greater at the primary sensor than at the secondary one (as also discussed in Subsection 3.2.3) while a similar noise PSD can be expected at both sensors (i.e. $\mathcal{S}_{n_1}(f,t) \approx \mathcal{S}_{n_2}(f,t) \gg \mathcal{S}_{x_2}(f,t)$) [97, 205]. As was mentioned above, this is due to the geometry of the speaker-device acoustic system (the secondary microphone is purposely placed in an acoustic shadow with respect to the speech source) and the typical existence of a homogeneous noise field. Under ideal conditions, we can consider that $\mathcal{S}_{n_1}(f,t) = \mathcal{S}_{n_2}(f,t)$ and $\mathcal{S}_{x_2}(f,t) = 0$. Therefore, $\mathcal{S}_{n_1}(f,t) = \mathcal{S}_{y_2}(f,t)$ and the *a priori* SNR of Eq. (3.28) can be expressed as

$$\xi_{1,b}(f,t) \;=\; \frac{\mathcal{S}_{y_1}(f,t) - \mathcal{S}_{y_2}(f,t)}{\mathcal{S}_{y_2}(f,t)}. \tag{3.30}$$

Hence, by using the result in (3.30), the corresponding WF is

$$H_{1,b}(f,t) = \frac{\mathcal{S}_{y_1}(f,t) - \mathcal{S}_{y_2}(f,t)}{\mathcal{S}_{y_1}(f,t)}. \tag{3.31}$$

In our proposal, the PSDs of the two available noisy signals required by *a priori* SNR computation in (3.30) are obtained by applying a two-dimensional 3×3 mean smoothing filter over the spectrogram $|Y_k(f,t)|^2$ $(k=1,2)$, that is,

$$\hat{\mathcal{S}}_{y_k}(f,t) = \frac{1}{\mathcal{D}} \sum_{\nu=-1}^{1} \sum_{\tau=-1}^{1} |Y_k(f+\nu, t+\tau)|^2, \tag{3.32}$$

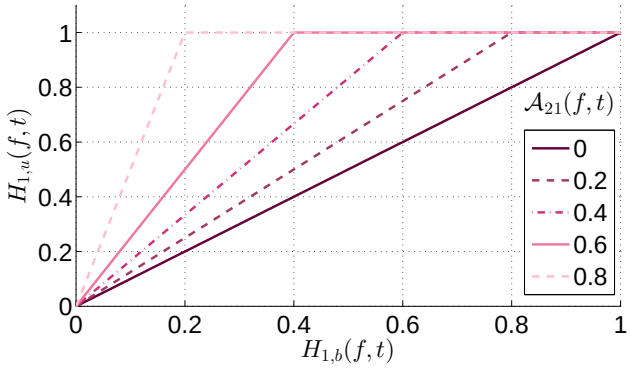

Figure 3.4: Result of applying the bias correction term on $H_{1,b}(f,t)$ for several $\mathcal{A}_{21}(f,t)$ values.

where we also assume $|Y_k(f,t)|^2 = 0$ for $f,t < 0$, $f \geq \mathcal{M}$ and $t \geq T$, and $\mathcal{D}$ is a normalizing factor equal to 9 except in the borders ($\mathcal{D} = 6$) and the corners ($\mathcal{D} = 4$) of the spectrogram. From these PSDs, the *a priori* SNR $\xi_{1,b}(f,t)$ is estimated in practice as,

$$\hat{\xi}_{1,b}(f,t) = \max\left(\frac{\hat{\mathcal{S}}_{y_1}(f,t) - \hat{\mathcal{S}}_{y_2}(f,t)}{\hat{\mathcal{S}}_{y_2}(f,t)}, \eta_\xi\right), \tag{3.33}$$

where this expression is floored at η_ξ in order to avoid negative values. Finally, the estimated WF $\hat{H}_{1,b}(f,t)$ is obtained by substituting (3.33) into (3.26).

The WF estimation described above strongly depends on the accuracy of the two assumptions made, that is, a negligible speech component at the secondary sensor and similar noise PSDs at both sensors. While these assumptions can be acceptable in some specific cases, in general, they will not be accurate. In the next subsections we will present two procedures that will allow us the application of the WF-based spectral weighting to a wider range of situations.

3.3.2 Unbiased spectral weight estimation

The assumption of a negligible speech component at the secondary channel may be appropriate, for instance, when a dual-microphone smartphone is employed in a close-talk position [97], but it will clearly fail when the device is used in far-talk conditions [140], as it was determined by the analysis in Subsection 3.2.3 supported by the $\log \mathcal{A}_{21}(f,t)$ example histograms. Indeed, the rear-side microphone also captures a component of speech mainly because of diffraction at the borders of the device and reflections from the acoustic environment. In this case, $\mathcal{S}_{y_2}(f,t) > \mathcal{S}_{n_2}(f,t)$ so that $\xi_{1,b}(f,t)$ and, therefore, $H_{1,b}(f,t)$, will be underestimated, i.e. biased (indicated by subscript b in the

above variables). In order to solve this issue, a bias correction term is introduced in the following.

Let us consider that the clean speech PSD at the secondary channel is related to that at the primary one by means of the relative speech gain (RSG) term $\mathcal{A}_{21}(f,t)$, i.e. $\mathcal{S}_{x_2}(f,t) = \mathcal{A}_{21}(f,t)\mathcal{S}_{x_1}(f,t)$. Once again assuming a homogeneous noise field ($S_{n_1}(f,t) = S_{n_2}(f,t)$), Eq. (3.29) for $k = 2$ can be written as

$$\begin{aligned} \mathcal{S}_{y_2}(f,t) &= \mathcal{A}_{21}(f,t)\mathcal{S}_{x_1}(f,t) + \mathcal{S}_{n_1}(f,t) \\ &= \mathcal{A}_{21}(f,t)\left(\mathcal{S}_{y_1}(f,t) - \mathcal{S}_{n_1}(f,t)\right) + \mathcal{S}_{n_1}(f,t). \end{aligned} \tag{3.34}$$

This equation allows us to express the noise PSD at the primary channel in terms of the PSDs of the available noisy signals, that is,

$$\mathcal{S}_{n_1}(f,t) = \frac{\mathcal{S}_{y_2}(f,t) - \mathcal{A}_{21}(f,t)\mathcal{S}_{y_1}(f,t)}{1 - \mathcal{A}_{21}(f,t)}. \tag{3.35}$$

By substituting this noise PSD into (3.28) we obtain the following expression for the *a priori* SNR:

$$\xi_{1,u}(f,t) = \frac{\mathcal{S}_{y_1}(f,t) - \mathcal{S}_{y_2}(f,t)}{\mathcal{S}_{y_2}(f,t) - \mathcal{A}_{21}(f,t)\mathcal{S}_{y_1}(f,t)}, \tag{3.36}$$

where subscript u indicates an unbiased approach. The SNR expression in (3.36) yields the following WF:

$$H_{1,u}(f,t) = \frac{\mathcal{S}_{y_1}(f,t) - \mathcal{S}_{y_2}(f,t)}{\mathcal{S}_{y_1}(f,t)(1 - \mathcal{A}_{21}(f,t))}. \tag{3.37}$$

By comparing this expression with that of Eq. (3.31), we observe that the WF bias can be corrected by dividing (3.31) by $B(f,t) = (1 - \mathcal{A}_{21}(f,t))$. In other words, the new WF can be obtained from the one in the previous subsection by applying the bias correction term $B^{-1}(f,t)$ as

$$H_{1,u}(f,t) = B^{-1}(f,t)H_{1,b}(f,t). \tag{3.38}$$

Figure 3.4 shows the effect of the bias correction term on $H_{1,b}(f,t)$ for different values of $\mathcal{A}_{21}(f,t)$. As can be observed, if $\mathcal{A}_{21}(f,t) = 0$ (i.e. no speech is captured by the secondary microphone), the assumption when calculating $\xi_{1,b}(f,t)$ through (3.30) holds true, so that the WF is not modified. On the other hand, as $\mathcal{A}_{21}(f,t)$ increases, the initial underestimation of $H_{1,b}(f,t)$ due to a non-negligible speech component at the secondary channel is rectified. It must be noted that, since $H_{1,u}(f,t) > 1$ has no physical sense, $B^{-1}(f,t)H_{1,b}(f,t)$ has been bounded by 1 in Figure 3.4.

Again from (3.32), the *a priori* SNR of (3.36), $\xi_{1,u}(f,t)$, is estimated in practice as,

$$\hat{\xi}_{1,u}(f,t) = \max\left(\frac{\hat{\mathcal{S}}_{y_1}(f,t) - \hat{\mathcal{S}}_{n_1}(f,t)}{\hat{\mathcal{S}}_{n_1}(f,t)}, \eta_\xi\right), \tag{3.39}$$

where, similarly to the case of (3.33), this expression is floored at η_ξ to avoid negative values. Moreover, the noise PSD $\mathcal{S}_{n_1}(f,t)$ calculated through (3.35) is also thresholded by η_n to avoid negative PSD bins, i.e.,

$$\hat{\mathcal{S}}_{n_1}(f,t) = \max\left(\frac{\hat{\mathcal{S}}_{y_2}(f,t) - \hat{\mathcal{A}}_{21}(f,t)\hat{\mathcal{S}}_{y_1}(f,t)}{1 - \hat{\mathcal{A}}_{21}(f,t)}, \eta_n\right). \tag{3.40}$$

Finally, the estimated WF $\hat{H}_{1,u}(f,t)$ is obtained by substituting (3.39) into (3.26).

3.3.3 Noise equalization

The assumption $\mathcal{S}_{n_1}(f,t) \approx \mathcal{S}_{n_2}(f,t)$ used for deriving the WF-based weighting in Subsections 3.3.1 and 3.3.2 could be acceptable when the mobile device is employed within a homogeneous noise field (e.g., in a diffuse noise field as in interior spaces, urban streets with high-rise buildings, etc. [182]). However, this assumption may not be satisfied in several scenarios even in the presence of a homogeneous noise field. For instance, this may happen in the case of two microphones with different characteristics (as exemplified at the end of Subsection 3.2.1), or as a result of the use of a virtual primary channel, since the application of beamforming modifies the spectral characteristics of the original noise. In this subsection we describe a noise equalization procedure to be performed before spectral weight computation. This procedure transforms the signal at the secondary channel so that its noise component is forced to follow the one at the primary channel while keeping the speech component untouched (this is a distortionless constraint similar to that of MVDR beamforming). Hence, we aim at obtaining a new spectrum $|\bar{Y}_2(f,t)|^2 = |\bar{X}_2(f,t)|^2 + |\bar{N}_2(f,t)|^2$, where $|\bar{X}_2(f,t)|^2 \approx |X_2(f,t)|^2$ and $|\bar{N}_2(f,t)|^2 \approx |N_1(f,t)|^2$, to be used instead of $|Y_2(f,t)|^2$ for the estimation of the PSD $\mathcal{S}_{y_2}(f,t)$ required in Eqs. (3.31) and (3.37) for the WF computation.

To make our equalization procedure more effective, we will additionally introduce an overestimation of the noise power spectrum. This overestimation is inspired by the oversubtraction typically applied in spectral subtraction (SS) which helps to reduce the "musical" artifacts, yielding a better recognition performance [18, 100]. In our case, since the PSD of the secondary channel carries the information about the noise required in (3.31) and (3.37), we will consider an overestimation factor $\beta(f,t) \geq 1$ so that

$$|\bar{Y}_2(f,t)|^2 \approx |X_2(f,t)|^2 + \beta(f,t)|N_1(f,t)|^2. \tag{3.41}$$

For computing $\beta(f,t)$, we follow the same approach reported in [183], where

$$\beta(f,t) = \left(1 + \frac{std\left(|N_1(f,t)|^2\right)}{|N_1(f,t)|^2}\right) \tag{3.42}$$

in which $std\left(|N_1(f,t)|^2\right)$ is the standard deviation of $|N_1(f,t)|^2$ at frequency bin f and time frame t. As a result, (3.41) can be rewritten as,

$$|\bar{Y}_2(f,t)|^2 \approx |X_2(f,t)|^2 + |N_1(f,t)|^2 + std\left(|N_1(f,t)|^2\right). \tag{3.43}$$

In particular, and following a constrained beamforming-like strategy, we will obtain $|\bar{Y}_2(f,t)|^2$ from the following linear combination of the dual-channel noisy observation:

$$\begin{aligned} |\bar{Y}_2(f,t)|^2 =& \mathbf{g}_{f,t}^{\top} \begin{pmatrix} |Y_2(f,t)|^2 \\ |Y_1(f,t)|^2 \end{pmatrix} \\ =& \mathbf{g}_{f,t}^{\top} \begin{pmatrix} |X_2(f,t)|^2 \\ |X_1(f,t)|^2 \end{pmatrix} + \mathbf{g}_{f,t}^{\top} \begin{pmatrix} |N_2(f,t)|^2 \\ |N_1(f,t)|^2 \end{pmatrix}, \end{aligned} \tag{3.44}$$

where $\mathbf{g}_{f,t}$ is the weight vector to be estimated. To avoid any possible negative power spectrum bin in (3.44), $|\bar{Y}_2(f,t)|^2$ is bounded below by $\eta|Y_2(f,t)|^2$, where $0 < \eta \ll 1$ is the same thresholding factor of (3.6). By using the RSG $\mathcal{A}_{21}(f,t)$ and $\bar{\boldsymbol{\nu}}(f,t) = \left(|N_2(f,t)|^2, |N_1(f,t)|^2\right)^{\top}$, (3.44) can be rewritten as

$$|\bar{Y}_2(f,t)|^2 = \mathbf{g}_{f,t}^{\top} \begin{pmatrix} 1 \\ \mathcal{A}_{21}^{-1}(f,t) \end{pmatrix} |X_2(f,t)|^2 + \mathbf{g}_{f,t}^{\top} \bar{\boldsymbol{\nu}}(f,t). \tag{3.45}$$

Then, by comparing (3.43) and (3.45) we can observe that our goal is to estimate the weight vector $\hat{\mathbf{g}}_{f,t}$ that transforms $\mathbf{g}_{f,t}^{\top}\bar{\boldsymbol{\nu}}(f,t)$ into $|N_1(f,t)|^2 + std\left(|N_1(f,t)|^2\right)$ with an MMSE criterion plus a distortionless constraint for $|X_2(f,t)|^2$. In other words, if we define $\boldsymbol{\alpha}(f,t) = \left(1, \mathcal{A}_{21}^{-1}(f,t)\right)^{\top}$ and

$$\varepsilon_{f,t} = \left(|N_1(f,t)|^2 + std\left(|N_1(f,t)|^2\right)\right) - \mathbf{g}_{f,t}^{\top}\bar{\boldsymbol{\nu}}(f,t), \tag{3.46}$$

we want to calculate

$$\begin{aligned} &\hat{\mathbf{g}}_{f,t} = \text{argmin}_{\mathbf{g}_{f,t}} \, \mathrm{E}\left[\varepsilon_{f,t}^2\right]; \\ &\text{subject to } \; \mathbf{g}_{f,t}^{\top}\boldsymbol{\alpha}(f,t) = 1. \end{aligned} \tag{3.47}$$

The optimization problem above is again solved by the Lagrange multipliers method, yielding the weight vector estimate

$$\hat{\mathbf{g}}_{f,t} = \bar{\boldsymbol{\Phi}}_N^{-1}(f,t)\left[\boldsymbol{\gamma}_N(f,t) - \frac{\boldsymbol{\alpha}^{\top}(f,t)\bar{\boldsymbol{\Phi}}_N^{-1}(f,t)\boldsymbol{\gamma}_N(f,t) - 1}{\boldsymbol{\alpha}^{\top}(f,t)\bar{\boldsymbol{\Phi}}_N^{-1}(f,t)\boldsymbol{\alpha}(f,t)}\boldsymbol{\alpha}(f,t)\right], \tag{3.48}$$

which, as can be seen, depends on the noise spatial correlation matrix $\bar{\mathbf{\Phi}}_N(f,t)$ and the overestimated noise spatial correlation vector $\boldsymbol{\gamma}_N(f,t)$. First, $\bar{\mathbf{\Phi}}_N(f,t)$ is defined as

$$\bar{\mathbf{\Phi}}_N(f,t) = \begin{pmatrix} \phi_{N,f,t}(2,2) & \phi_{N,f,t}(2,1) \\ \phi_{N,f,t}(1,2) & \phi_{N,f,t}(1,1) \end{pmatrix}, \tag{3.49}$$

where it should be reminded that $\phi_{N,f,t}(k,l) = \mathrm{E}\left[|N_k(f,t)|^2|N_l(f,t)|^2\right]$ $(k,l=1,2)$ is a cross-correlation coefficient as for the case of both DCSS and P-MVDR. Second, the overestimated noise spatial correlation vector is

$$\boldsymbol{\gamma}_N(f,t) = \boldsymbol{\phi}_N^{(1)}(f,t) + std\left(|N_1(f,t)|^2\right)\boldsymbol{\mu}_N(f,t), \tag{3.50}$$

where

$$\boldsymbol{\phi}_N^{(1)}(f,t) = \begin{pmatrix} \phi_{N,f,t}(2,1) \\ \phi_{N,f,t}(1,1) \end{pmatrix} \tag{3.51}$$

is a noise spatial correlation vector and the noise mean vector is expressed as

$$\boldsymbol{\mu}_N(f,t) = \begin{pmatrix} \mathrm{E}\left[|N_2(f,t)|^2\right] \\ \mathrm{E}\left[|N_1(f,t)|^2\right] \end{pmatrix}. \tag{3.52}$$

Since the mathematical derivation of the noise equalization weighting vector is considered of particular interest, it is detailed in Appendix A.

In practice, the required noise statistical parameters $\bar{\mathbf{\Phi}}_N(f,t)$ and $\boldsymbol{\gamma}_N(f,t)$ may be estimated during noise-only periods identified by means of a voice activity detector (VAD). In particular, in this Thesis both the noise spatial correlation matrix $\bar{\mathbf{\Phi}}_N(f,t)$ of Eq. (3.49) and the overestimated noise spatial correlation vector $\boldsymbol{\gamma}_N(f,t)$ of (3.50) are calculated as follows. First, two initial noise spatial correlation matrices as well as two initial overestimated noise spatial correlation vectors are computed per utterance and frequency bin f. One is obtained from the first M frames ($\bar{\mathbf{\Phi}}_{N,f}^{(0)}$ and $\boldsymbol{\gamma}_{N,f}^{(0)}$, respectively) and the other from the last M frames ($\bar{\mathbf{\Phi}}_{N,f}^{(e)}$ and $\boldsymbol{\gamma}_{N,f}^{(e)}$, respectively). It must be noted that it is assumed that those first and last M frames of every utterance contain only noise energy. Then, $\bar{\mathbf{\Phi}}_N(f,t)$ ($\boldsymbol{\gamma}_N(f,t)$) is calculated by means of linear interpolation between $\bar{\mathbf{\Phi}}_{N,f}^{(0)}$ ($\boldsymbol{\gamma}_{N,f}^{(0)}$) and $\bar{\mathbf{\Phi}}_{N,f}^{(e)}$ ($\boldsymbol{\gamma}_{N,f}^{(e)}$). Indeed, the linear interpolation approach [159] (as well as the one taken into account by both DCSS and P-MVDR for the calculation of the cross-correlation coefficients of noise) is an offline noise estimation strategy. In this respect, it has to be said that it is usual to find in the literature that the computation of different types of parameters, such as noise statistics-related parameters, is performed on an utterance-by-utterance basis. Thus, for instance, in [203], the clean speech spatial covariance matrix, needed to derive the steering vector by eigenvalue decomposition for MVDR beamforming, is computed per utterance and frequency bin by using all the

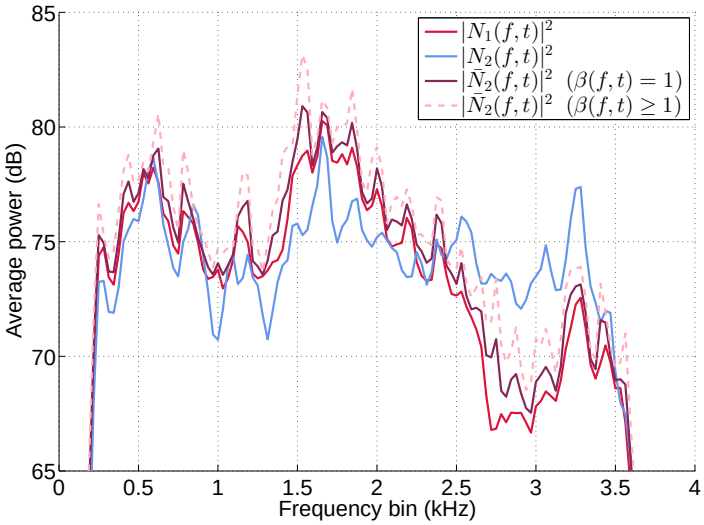

Figure 3.5: Example of the developed noise equalization when applied on an utterance from a dual-microphone smartphone used in close-talk position. Both the estimated noise average power $|\bar{N}_2(f,t)|^2$ $(\beta(f,t)=1)$ and its overestimated version, $|\bar{N}_2(f,t)|^2$ $(\beta(f,t)\geq 1$ as in Eq. (3.42)), are represented by frequency bin along with the actual noise average power from the two channels.

frames in each utterance. Additionally, in [90], the noise spatial covariance matrix for MVDR beamforming is also calculated by employing the beginning and ending parts of each utterance.

An application example of the noise equalization procedure developed above is depicted in Figure 3.5. The figure shows the noise spectra obtained from a dual-microphone smartphone in close-talk position averaged across time over the whole utterance. It can be observed that the equalized noise $|\bar{N}_2(f,t)|^2$ is much more similar to $|N_1(f,t)|^2$ than the original $|N_2(f,t)|^2$. The effect of the noise overestimation factor $\beta(f,t)$ is also shown.

To conclude, it should be pointed out that by setting the constraint of Eq. (3.47) as $\mathbf{g}_{f,t}^{\top}\boldsymbol{\alpha}(f,t)=0$, the noise equalization procedure described in this subsection is able to produce estimates of the noise power spectrum at the primary channel. Such estimates might be used together with single-channel Wiener filtering to perform the sought spectral weighting. Nevertheless, preliminary speech recognition experiments revealed that the combination of the noise equalizer here described plus the unbiased spectral weighting of Subsection 3.3.2 (to compensate for the presence of speech energy at the secondary channel) is superior to that alternative.

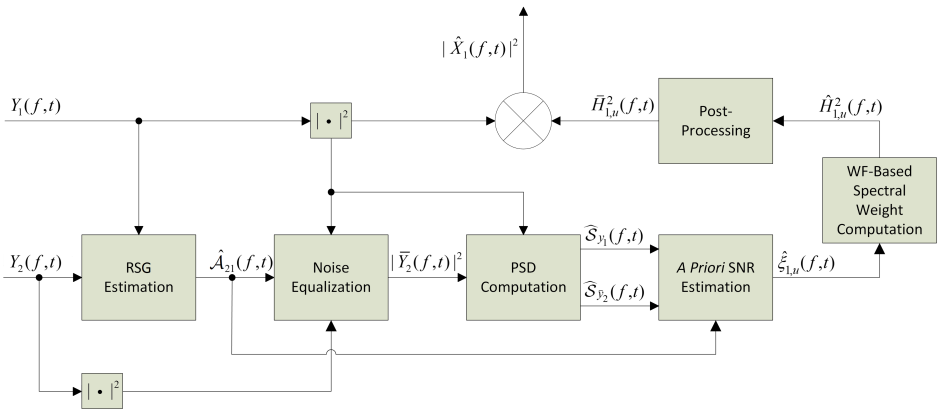

Figure 3.6: Block diagram of the full dual-channel spectral weighting system.

3.3.4 Post-processing and system overview

All the stages described above, plus the RSG estimation which will be formulated in Section 3.4 and a post-processing block, are accordingly interconnected to define the dual-channel spectral weighting system represented in Figure 3.6. This proposed post-processing block carries out a series of operations on either the WF-based spectral weights $\hat{H}^2_{1,b}(f,t)$ or $\hat{H}^2_{1,u}(f,t)$. For the sake of clarity let us consider only $\hat{H}^2_{1,u}(f,t)$ for the rest of this explanation. First, for speech recognition purposes, previous works have shown that better results can be achieved by leaving a small fraction of noise energy in the enhanced signal [18, 129]. Hence, $\hat{H}^2_{1,u}(f,t)$ is bounded below in accordance with

$$\bar{H}^2_{1,u}(f,t) = \max\left(\hat{H}^2_{1,u}(f,t), \eta\right), \tag{3.53}$$

where η is the same thresholding factor as in previous subsections (e.g. see Eq. (3.6)). Indeed, thresholding $\hat{H}^2_{1,u}(f,t)$ by η is equivalent to consider

$$\eta_\xi = \frac{\sqrt{\eta}}{1-\sqrt{\eta}} \tag{3.54}$$

in (3.39) in accordance with the WF definition of (3.26). So this thresholding enhancement is directly accomplished by including (3.54) into (3.39).

Second, as in [91], we exploit the spectro-temporal correlation of speech in order to refine $\bar{H}^2_{1,u}(f,t)$ by applying a couple of two-dimensional filters in the time-frequency domain. The first one is a median filter of size $M_f \times M_t$, that tries to remove those high-valued $\bar{H}^2_{1,u}(f,t)$ bins surrounded by low values. This procedure is justified by the fact that it is more likely that those bins constitute artifacts rather than actual isolated clean speech spectral bins. This kind of artifact often appears when the assumption $\mathcal{S}_{n_1}(f,t) \approx \mathcal{S}_{n_2}(f,t)$ does not hold but instead $\mathcal{S}_{n_1}(f,t)$ is significantly greater

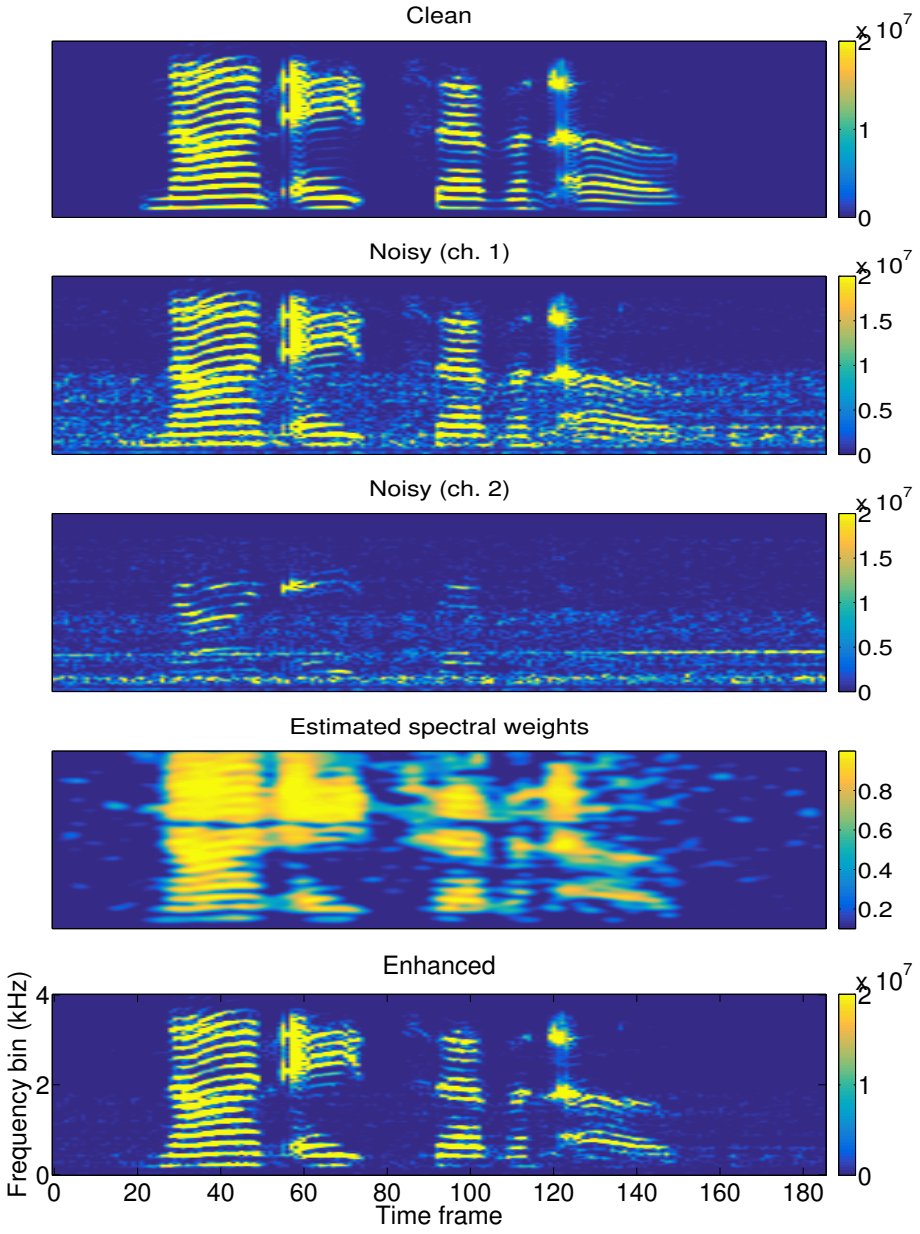

Figure 3.7: Example of spectral weighting by using the enhancement system of Fig. 3.6 for the utterance "nine eight seven oh" obtained from a dual-microphone smartphone in close-talk position. The utterance is contaminated with car noise at 0 dB in the primary channel. From top to bottom: clean speech power spectrum in the primary channel, noisy versions at the 1st and 2nd channels, estimated spectral weights and enhanced power spectrum.

than $\mathcal{S}_{n_2}(f,t)$. Then, in order to further increase the spectro-temporal coherence, the spectral weights are smoothed by convolving them with a Gaussian kernel of standard deviation σ_G and size $G_f \times G_t$.

Figure 3.7 shows an example of applying the estimated spectral weights by means of the full enhancement system in Figure 3.6 to the primary noisy power spectrum of an utterance captured by a dual-microphone smartphone used in close-talk position. In this example, the noise reduction capability of our system can be visually inspected.

3.4 MMSE-based relative speech gain estimation

The relative speech gain (RSG) $\mathcal{A}_{21}(f,t)$, required by all the enhancement contributions developed above (i.e. DCSS, P-MVDR and DSW), can be obtained through an MMSE-based estimation procedure, which is developed in this section. To this end, considering that $A_{21}(f,t)$ is the RSG in the short-time Fourier transform (STFT) domain (i.e. $\mathcal{A}_{21}(f,t) = |A_{21}(f,t)|^2$), let us define the following vectors with STFT coefficients for a particular time frame t and $k = 1, 2$:

$$\begin{aligned} \mathbf{a}_{21} &= (A_{21}(0,t), A_{21}(1,t), ..., A_{21}(\mathcal{M}-1,t))^\top; \\ \mathbf{y}_k &= (Y_k(0,t), Y_k(1,t), ..., Y_k(\mathcal{M}-1,t))^\top; \\ \mathbf{x}_k &= (X_k(0,t), X_k(1,t), ..., X_k(\mathcal{M}-1,t))^\top; \\ \mathbf{n}_k &= (N_k(0,t), N_k(1,t), ..., N_k(\mathcal{M}-1,t))^\top. \end{aligned} \tag{3.55}$$

It should be remarked that a reference to the time frame index t for the variables $\mathbf{a}_{21}$, $\mathbf{y}_k$, $\mathbf{x}_k$ and $\mathbf{n}_k$ has been omitted for the sake of clarity. All the variables in (3.55) can be decomposed into real and imaginary parts (e.g. $\mathbf{a}_{21} = \mathbf{a}_{21}^r + j\mathbf{a}_{21}^i$, where superscripts r and i denote real and imaginary parts, respectively). In order to develop a straightforward estimator for $\mathbf{a}_{21}$, we will apply an *ad-hoc* model of the secondary signal $\mathbf{y}_2$ for a specific primary one $\mathbf{y}_1$. Under this modeling, once $\mathbf{y}_1$ is observed, $\mathbf{y}_2$ is determined by the RSG vector $\mathbf{a}_{21}$ as well as by the noises affecting both channels (see Eq. (3.59)). Then, the MMSE estimate of $\mathbf{a}_{21}$ can be expressed as,

$$\begin{aligned} \hat{\mathbf{a}}_{21} &= \mathrm{E}\left[\mathbf{a}_{21}|\mathbf{y}_2\right] \\ &= \mathrm{E}\left[\mathbf{a}_{21}^r|\mathbf{y}_2^r\right] + j\mathrm{E}\left[\mathbf{a}_{21}^i|\mathbf{y}_2^i\right] \\ &= \hat{\mathbf{a}}_{21}^r + j\hat{\mathbf{a}}_{21}^i, \end{aligned} \tag{3.56}$$

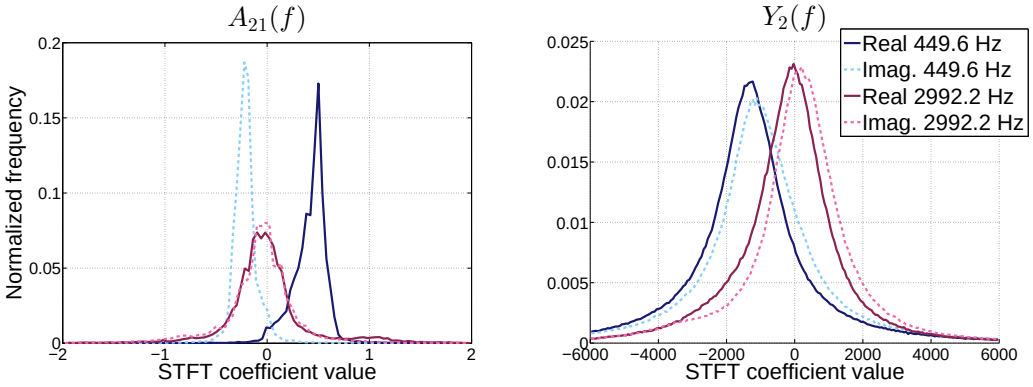

Figure 3.8: Example histograms of $A_{21}(f)$ (left) and $Y_2(f)$ (right) given a value of $\mathbf{y}_1$, at two different frequency bins, calculated from noisy speech data captured with a dual-microphone smartphone used in far-talk conditions.

where it has been assumed that the real and imaginary parts of $\mathbf{a}_{21}$ (i.e. $\mathbf{a}_{21}^r$ and $\mathbf{a}_{21}^i$) are statistically independent. It should be noticed that, for the sake of simplicity, a similar assumption was previously adopted in [96, 131] for the real and imaginary parts of the clean speech STFT coefficients. Since our estimator performs on a frame-by-frame basis, it is able to cope with variations over time on the relative location between the speaker and the microphones or the environment acoustics. It should also be noted that although phase information provided by our method is finally discarded in this Thesis, that could be exploited for different purposes, e.g. the definition of the steering vector for beamforming.

All the formulation below is intended to provide an estimation of $\mathbf{a}_{21}^r$. A similar procedure can be followed to obtain $\hat{\mathbf{a}}_{21}^i$, which is detailed in Appendix B. Variables $\mathbf{a}_{21}^r$ and $\mathbf{y}_2^r$ are assumed to be Gaussian-distributed (as well as $\mathbf{a}_{21}^i$ and $\mathbf{y}_2^i$), i.e. $\mathbf{a}_{21}^r \sim \mathcal{N}\left(\boldsymbol{\mu}_{A_{21}^r}, \boldsymbol{\Sigma}_{A_{21}^r}\right)$ and $\mathbf{y}_2^r \sim \mathcal{N}\left(\bar{\mathbf{y}}_2^r, \boldsymbol{\Sigma}_{Y_2^r}\right)$. The suitability of such an assumption is deduced from Figure 3.8, which depicts example histograms of the real and imaginary parts of two frequency components of $\mathbf{a}_{21}$ and $\mathbf{y}_2$. It should be noticed that a constant typical value of $\mathbf{y}_1$ was chosen to compute the $\mathbf{y}_2$ example histograms according to the aforementioned *ad-hoc* modeling (described below). Moreover, for the sake of simplicity we will assume that $\mathbf{a}_{21}^r$ and $\mathbf{y}_2^r$ are jointly Gaussian and, as a result, the conditional probability density function (PDF) $p(\mathbf{a}_{21}^r|\mathbf{y}_2^r)$ is also Gaussian [20]. Thus,

$$\begin{aligned} \hat{\mathbf{a}}_{21}^r &= \mathrm{E}\left[\mathbf{a}_{21}^r|\mathbf{y}_2^r\right] \\ &= \int \mathbf{a}_{21}^r p(\mathbf{a}_{21}^r|\mathbf{y}_2^r) d\mathbf{a}_{21}^r \\ &= \boldsymbol{\mu}_{A_{21}^r} + \boldsymbol{\Sigma}_{A_{21}^r Y_2^r} \boldsymbol{\Sigma}_{Y_2^r}^{-1} \left(\mathbf{y}_2^r - \bar{\mathbf{y}}_2^r\right). \end{aligned} \tag{3.57}$$

We consider that the parameters of the *a priori* distribution of $\mathbf{a}_{21}^r$, $\boldsymbol{\mu}_{A_{21}^r}$ and $\boldsymbol{\Sigma}_{A_{21}^r}$, are known. Their estimation from the available data is discussed in Subsection 6.2.2. Thus, we now focus on the estimation of the rest of required parameters, that is, $\boldsymbol{\Sigma}_{A_{21}^r Y_2^r}$, $\boldsymbol{\Sigma}_{Y_2^r}$ and $\bar{\mathbf{y}}_2^r$.

First, we will obtain the mean vector $\bar{\mathbf{y}}_2^r$ and the covariance matrix of the PDF $p(\mathbf{y}_2^r) = \mathcal{N}\left(\bar{\mathbf{y}}_2^r, \boldsymbol{\Sigma}_{Y_2^r}\right)$. In our *ad-hoc* modeling, $\mathbf{y}_2$ (and, therefore, $\mathbf{y}_2^r$) could be fully determined from the observation $\mathbf{y}_1$ of the primary channel if we had knowledge about $\mathbf{a}_{21}$, $\mathbf{n}_1$, and $\mathbf{n}_2$, that is, $\mathbf{y}_2 = \mathbf{h}(\mathbf{a}_{21}, \mathbf{n}_1, \mathbf{n}_2; \mathbf{y}_1)$. In order to obtain this function, we first adapt our additive distortion model for $\mathbf{y}_2$ to the STFT domain as,

$$\begin{aligned} \mathbf{y}_1 &= \mathbf{x}_1 + \mathbf{n}_1; \\ \mathbf{y}_2 &= \mathbf{x}_2 + \mathbf{n}_2 = \mathbf{a}_{21} \odot \mathbf{x}_1 + \mathbf{n}_2, \end{aligned} \tag{3.58}$$

where $\odot$ stands for element-wise multiplication. The combination of both expressions in (3.58) finally yields,

$$\begin{aligned} \mathbf{y}_2 &= \mathbf{h}(\mathbf{a}_{21}, \mathbf{n}_1, \mathbf{n}_2; \mathbf{y}_1) \\ &= \mathbf{a}_{21} \odot (\mathbf{y}_1 - \mathbf{n}_1) + \mathbf{n}_2, \end{aligned} \tag{3.59}$$

where $\mathbf{y}_1$ is observable and the variables $\mathbf{a}_{21}$, $\mathbf{n}_1$ and $\mathbf{n}_2$ are unknown. From (3.59), $\mathbf{y}_2^r = \mathrm{Re}(\mathbf{y}_2)$ can be modeled as

$$\begin{aligned} \mathbf{y}_2^r &= \mathbf{h}_r\left(\mathbf{a}_{21}^r, \mathbf{a}_{21}^i, \mathbf{n}_1^r, \mathbf{n}_1^i, \mathbf{n}_2^r; \mathbf{y}_1^r, \mathbf{y}_1^i\right) \\ &= \mathbf{a}_{21}^r \odot (\mathbf{y}_1^r - \mathbf{n}_1^r) - \mathbf{a}_{21}^i \odot (\mathbf{y}_1^i - \mathbf{n}_1^i) + \mathbf{n}_2^r. \end{aligned} \tag{3.60}$$

We will also assume that the variables $\mathbf{n}_k^r$ and $\mathbf{n}_k^i$ $(k = 1, 2)$ follow multivariate Gaussian distributions [44]. Since any linear combination of Gaussian variables follows another Gaussian distribution [150], we linearize the distortion model in (3.60) as a first step before describing $\mathbf{y}_2^r$ by means of a multivariate Gaussian distribution. This is carried out by means of the following first-order vector Taylor series (VTS) expansion of (3.60) around $(\boldsymbol{\mu}_{A_{21}^r}, \boldsymbol{\mu}_{A_{21}^i}, \boldsymbol{\mu}_{N_1^r}, \boldsymbol{\mu}_{N_1^i}, \boldsymbol{\mu}_{N_2^r})$:

$$\begin{aligned} \mathbf{y}_2^r &= \mathbf{h}_r\left(\mathbf{a}_{21}^r, \mathbf{a}_{21}^i, \mathbf{n}_1^r, \mathbf{n}_1^i, \mathbf{n}_2^r; \mathbf{y}_1^r, \mathbf{y}_1^i\right) \\ &\approx \mathbf{h}_r\left(\boldsymbol{\mu}_{A_{21}^r}, \boldsymbol{\mu}_{A_{21}^i}, \boldsymbol{\mu}_{N_1^r}, \boldsymbol{\mu}_{N_1^i}, \boldsymbol{\mu}_{N_2^r}; \mathbf{y}_1^r, \mathbf{y}_1^i\right) + \mathbf{J}_{A_{21}^r}^r\left(\mathbf{a}_{21}^r - \boldsymbol{\mu}_{A_{21}^r}\right) \\ &\quad + \mathbf{J}_{A_{21}^i}^r\left(\mathbf{a}_{21}^i - \boldsymbol{\mu}_{A_{21}^i}\right) + \mathbf{J}_{N_1^r}^r\left(\mathbf{n}_1^r - \boldsymbol{\mu}_{N_1^r}\right) + \mathbf{J}_{N_1^i}^r\left(\mathbf{n}_1^i - \boldsymbol{\mu}_{N_1^i}\right) \\ &\quad + \mathbf{J}_{N_2^r}^r\left(\mathbf{n}_2^r - \boldsymbol{\mu}_{N_2^r}\right), \end{aligned} \tag{3.61}$$

where the $\mathcal{M}\times\mathcal{M}$ Jacobian matrices $\mathbf{J}^r_{A^r_{21}}$, $\mathbf{J}^r_{A^i_{21}}$, $\mathbf{J}^r_{N^r_1}$, $\mathbf{J}^r_{N^i_1}$ and $\mathbf{J}^r_{N^r_2}$ have, respectively, the following definitions,

$$\begin{aligned}
\mathbf{J}^r_{A^r_{21}} &= \left.\frac{\partial \mathbf{y}^r_2}{\partial \mathbf{a}^r_{21}}\right|_{\boldsymbol{\mu}_{N^r_1}} &= \operatorname{diag}\left(\mathbf{y}^r_1 - \boldsymbol{\mu}_{N^r_1}\right);\\
\mathbf{J}^r_{A^i_{21}} &= \left.\frac{\partial \mathbf{y}^r_2}{\partial \mathbf{a}^i_{21}}\right|_{\boldsymbol{\mu}_{N^i_1}} &= -\operatorname{diag}\left(\mathbf{y}^i_1 - \boldsymbol{\mu}_{N^i_1}\right);\\
\mathbf{J}^r_{N^r_1} &= \left.\frac{\partial \mathbf{y}^r_2}{\partial \mathbf{n}^r_1}\right|_{\boldsymbol{\mu}_{A^r_{21}}} &= -\operatorname{diag}\left(\boldsymbol{\mu}_{A^r_{21}}\right);\\
\mathbf{J}^r_{N^i_1} &= \left.\frac{\partial \mathbf{y}^r_2}{\partial \mathbf{n}^i_1}\right|_{\boldsymbol{\mu}_{A^i_{21}}} &= \operatorname{diag}\left(\boldsymbol{\mu}_{A^i_{21}}\right);\\
\mathbf{J}^r_{N^r_2} &= \frac{\partial \mathbf{y}^r_2}{\partial \mathbf{n}^r_2} &= \mathbf{I}_{\mathcal{M}},
\end{aligned} \tag{3.62}$$

where $\mathbf{I}_{\mathcal{M}}$ is an $\mathcal{M}\times\mathcal{M}$ identity matrix and $\operatorname{diag}(\cdot)$ indicates a diagonal matrix whose main diagonal corresponds to that from its argument. Then, by considering the linearized distortion model of (3.61), the mean vector of the PDF $p(\mathbf{y}^r_2)$[1] can be approximated as,

$$\begin{aligned}
\bar{\mathbf{y}}^r_2 &= \mathrm{E}\left[\mathbf{y}^r_2\right]\\
&\approx \mathbf{h}_r\left(\boldsymbol{\mu}_{A^r_{21}}, \boldsymbol{\mu}_{A^i_{21}}, \boldsymbol{\mu}_{N^r_1}, \boldsymbol{\mu}_{N^i_1}, \boldsymbol{\mu}_{N^r_2}; \mathbf{y}^r_1, \mathbf{y}^i_1\right).
\end{aligned} \tag{3.63}$$

This mean can be considered as a predicted value for $\mathbf{y}^r_2$ obtained from $\boldsymbol{\mu}_{A^r_{21}}$, $\boldsymbol{\mu}_{A^i_{21}}$, $\boldsymbol{\mu}_{N^r_1}$, $\boldsymbol{\mu}_{N^i_1}$, $\boldsymbol{\mu}_{N^r_2}$ and $\mathbf{y}_1$.

The covariance matrix of $p(\mathbf{y}^r_2)$ can be approximated by following a similar procedure from (3.61) and assuming statistical independence between $\mathbf{a}_{21}$ and $(\mathbf{n}_1, \mathbf{n}_2)$, as well as between the real and imaginary parts of all the variables involved:

$$\begin{aligned}
\boldsymbol{\Sigma}_{Y^r_2} &= \mathrm{E}\left[(\mathbf{y}^r_2 - \bar{\mathbf{y}}^r_2)(\mathbf{y}^r_2 - \bar{\mathbf{y}}^r_2)^\top\right]\\
&\approx \mathbf{J}^r_{A^r_{21}}\boldsymbol{\Sigma}_{A^r_{21}}{\mathbf{J}^r_{A^r_{21}}}^\top + \mathbf{J}^r_{A^i_{21}}\boldsymbol{\Sigma}_{A^i_{21}}{\mathbf{J}^r_{A^i_{21}}}^\top + \mathbf{J}^r_{N^r_1}\boldsymbol{\Sigma}_{N^r_1}{\mathbf{J}^r_{N^r_1}}^\top\\
&\quad + \mathbf{J}^r_{N^r_1}\boldsymbol{\Sigma}_{N^r_1 N^r_2}{\mathbf{J}^r_{N^r_2}}^\top + \mathbf{J}^r_{N^i_1}\boldsymbol{\Sigma}_{N^i_1}{\mathbf{J}^r_{N^i_1}}^\top + \mathbf{J}^r_{N^r_2}\boldsymbol{\Sigma}_{N^r_2 N^r_1}{\mathbf{J}^r_{N^r_1}}^\top\\
&\quad + \mathbf{J}^r_{N^r_2}\boldsymbol{\Sigma}_{N^r_2}{\mathbf{J}^r_{N^r_2}}^\top,
\end{aligned} \tag{3.64}$$

[1]Notice that, according to our *ad-hoc* model, this PDF is specifically built for the given observation $\mathbf{y}_1$, although this dependency has been removed from our notation for the sake of simplicity.

with $\mathbf{\Sigma}_{N_1^r N_2^r} = \mathbf{\Sigma}_{N_2^r N_1^r}^\top = \mathrm{E}\left[(\mathbf{n}_1^r - \boldsymbol{\mu}_{N_1^r})(\mathbf{n}_2^r - \boldsymbol{\mu}_{N_2^r})^\top\right]$.

Finally, taking into account the statistical independence between $\mathbf{a}_{21}$ and $(\mathbf{n}_1, \mathbf{n}_2)$, as well as between $\mathbf{a}_{21}^r$ and $\mathbf{a}_{21}^i$, the covariance matrix $\mathbf{\Sigma}_{A_{21}^r Y_2^r}$ can be approximated as,

$$\begin{aligned} \mathbf{\Sigma}_{A_{21}^r Y_2^r} &= \mathrm{E}\left[(\mathbf{a}_{21}^r - \boldsymbol{\mu}_{A_{21}^r})(\mathbf{y}_2^r - \bar{\mathbf{y}}_2^r)^\top\right] \\ &\approx \mathbf{\Sigma}_{A_{21}^r} \mathbf{J}_{A_{21}^r}^{r\,\top}. \end{aligned} \tag{3.65}$$

In Subsection 6.2.2 the estimation of the parameters of the *a priori* PDFs $p(\mathbf{a}_{21}^r)$ and $p(\mathbf{a}_{21}^i)$ required to perform this method will be presented. In particular, $\boldsymbol{\mu}_{A_{21}^r}$, $\mathbf{\Sigma}_{A_{21}^r}$, $\boldsymbol{\mu}_{A_{21}^i}$ and $\mathbf{\Sigma}_{A_{21}^i}$ must be obtained in advance for every mobile device. On the other hand, the hyperparameters $\boldsymbol{\mu}_{N_k^r}$, $\mathbf{\Sigma}_{N_k^r}$, $\boldsymbol{\mu}_{N_k^i}$ and $\mathbf{\Sigma}_{N_k^i}$, $k = 1, 2$, as well as $\mathbf{\Sigma}_{N_1^r N_2^r}$ and $\mathbf{\Sigma}_{N_1^i N_2^i}$, are calculated on an utterance-by-utterance basis. Once again, it is considered that the first and last M frames from each utterance contain only noise energy. Then, an initial noise estimation is obtained by linear interpolation between the averages of the magnitude of the first and last M frames in the k-th channel of each utterance in the STFT domain. This noise magnitude estimate is then used along with the original noisy speech phase to shape an STFT noise estimate. Thus, the mean vectors of the PDFs $p(\mathbf{n}_k^r)$ and $p(\mathbf{n}_k^i)$, i.e. $\boldsymbol{\mu}_{N_k^r}$ and $\boldsymbol{\mu}_{N_k^i}$ $(k = 1, 2)$, are updated at every time frame t from the above complex noise estimate. The covariance matrices $\mathbf{\Sigma}_{N_k^r}$ and $\mathbf{\Sigma}_{N_k^i}$ $(k = 1, 2)$, and $\mathbf{\Sigma}_{N_1^r N_2^r}$ and $\mathbf{\Sigma}_{N_1^i N_2^i}$, are estimated per utterance as the sample covariance of the first and last M frames. Independence between frequency bins was also assumed in practice, such that all types of noise covariance matrices are diagonal. While for the sake of simplicity it is typical to find in the literature this independence assumption across frequency bins (e.g. [65, 157]), our preliminary experiments also showed that better speech recognition performance could be achieved by adopting this assumption for our MMSE-based RSG estimator. In addition, it is worth to notice that independence between frequency bins is just a practical assumption independent of our method, so that, according to its formulation, it can deal with correlations across frequency bins by just considering full covariance matrices.

3.5 Summary

The core of this chapter has been the presentation of three different power spectrum enhancement contributions, which are intended to exploit the dual-channel noisy speech signal coming from a mobile device for improved speech recognition: DCSS (Dual-Channel Spectral Subtraction), P-MVDR (Power-MVDR) and DSW (Dual-channel Spectral Weighting). While it is expected that a mobile device has no more than one

secondary sensor (i.e. a microphone with the main purpose of getting information about the acoustic environment which does not look towards the speaker), it is likely that such a device integrates various front sensors. Under this scenario, a virtual primary channel is defined by the application of beamforming from either the front sensors or all the microphones in the device (namely by also taking into account the secondary microphone). Then, this virtual primary channel can be used along with the secondary one by the proposed dual-channel power spectrum enhancement methods, in such a manner that they behave as post-filters.

The two basic approaches DCSS and P-MVDR have been presented in the first place. DCSS extends spectral subtraction to a dual-channel framework to outperform the single-channel spectral subtraction. Additionally, P-MVDR is based on MVDR beamforming but discarding the phase information in order to overcome the limitations when applied to a mobile device with a few microphones very close each other. Both DCSS and P-MVDR exploit the spatial properties of speech and noise by means of the relative speech gain factor and noise spatial correlation terms, respectively. Then, through a comparative study it was determined that P-MVDR is more robust than DCSS when a dual-microphone smartphone is used in far-talk conditions.

The third power spectrum enhancement contribution, DSW, consisted of a dual-channel spectral weighting based on Wiener filtering. DSW starts from a simple formulation in which it is assumed that the secondary microphone only captures noise and the existence of a homogeneous noise field. Since it is known that both assumptions are not accurate, the WF-based weighting is modified through *1)* a bias correction term (to rectify the resulting spectral weights when a non-negligible speech component is present at the secondary channel) and *2)* a noise equalization (inspired by MVDR beamforming) applied on the secondary channel before spectral weight computation.

Finally, since all DCSS, P-MVDR and DSW require knowledge of the relative speech gain between the secondary microphone and the primary one, a robust MMSE-based estimator was developed at the end of this chapter to obtain this parameter in an efficient way (as will be shown in our experimental evaluation).

CHAPTER 4

Dual-Channel Vector Taylor Series Feature Compensation

The vector Taylor series (VTS) approach for noise-robust automatic speech recognition (ASR) has been widely studied over the last two decades. As we saw in Chapter 2, the VTS strategy has traditionally been followed to accomplish either model adaptation or feature compensation. In this chapter, VTS feature compensation is extended to be performed on a dual-channel framework in a similar fashion to the power spectrum enhancement methods of Chapter 3. Of course, in the case of a mobile device with more than one front sensor, or more than two sensors of any type, the combinatorial strategy presented in 3.1 could be followed. It must be noted that the feature compensation scheme is preferred here over model adaptation due to the considerably lower computational complexity of the former.

In the following it is developed a minimum mean square error (MMSE)-based estimator of the log-Mel clean speech features that exploits the dual-channel noisy observations and relies on a VTS expansion of the dual-channel speech distortion model stated in Section 4.1. Through this approach, the noisy speech statistics, needed for the MMSE estimation, are easily derived in an analytical way from the clean speech, relative speech gain and noise statistics. Our dual-channel VTS contribution, explained in detail throughout Section 4.2, mainly follows a stacked formulation that exploits clean speech and noise correlations across the two available channels in order to achieve more accurate clean speech estimates than a single-channel VTS scheme.

The general overview of the dual-channel MMSE-based estimation is given in Subsection 4.2.1. As will be seen, such an estimation is composed of two differentiated components: a set of posterior probabilities and other set of clean speech partial es-

timates. Thus, dual-channel posterior computation is developed in Subsection 4.2.2, where, besides the posterior probabilities derived from the stacked formulation, an alternative approach is also explored. This alternative strategy, unlike the stacked formulation, models the conditional dependence of the noisy secondary channel given the primary one. This leads to a different derivation where the correlations between the two channels are exploited in a more robust way. Then, clean speech partial estimate computation is presented in Subsection 4.2.3. Finally, the contributions described in this chapter are summarized in Section 4.3.

Additionally, it should be mentioned that, as in Chapter 3, without loss of generality, different issues will be illustrated throughout the present chapter by considering again a dual-microphone smartphone employed in either close- or far-talk conditions.

4.1 Dual-channel distortion model for feature compensation

Let us consider the speech distortion model introduced in Section 2.1 in the log-Mel power spectral domain. It will be simplified by neglecting the convolutive distortion in first instance, although this type of distortion can be tackled as explained down below. Hence, if $\mathbf{y}_i$, $\mathbf{x}_i$ and $\mathbf{n}_i$ are noisy speech, clean speech and noise log-Mel feature vectors coming from the i-th channel of the mobile device at a particular time frame (where the time frame index t has been omitted in these variables for the sake of clarity), the considered additive noise distortion model from now onwards is

$$\mathbf{y}_i = \log\left(e^{\mathbf{x}_i} + e^{\mathbf{n}_i}\right), \tag{4.1}$$

where it should be reminded that the operators $\log(\cdot)$ and $e^{(\cdot)}$ are applied element-wise, as well as $i = 1$ and $i = 2$ correspond to the primary and secondary channels, respectively.

Besides the additive noise, we must also consider the acoustics involved in our problem. Thus, we assume that the clean speech signal $x_i(m)$ is the result of filtering the clean speech source $x(m)$ by the acoustics $h_i(m)$ that affect sensor i, that is, $x_i(m) = h_i(m) * x(m)$ or, in terms of log-Mel power spectra,

$$\begin{aligned} \mathbf{x}_i &= \mathbf{h}_i + \mathbf{x} \\ &= \mathbf{a}_{i1} + \mathbf{x}_1, \end{aligned} \tag{4.2}$$

where $\mathbf{h}_i$ and $\mathbf{x}$ are vectors of size $\mathcal{M}$ (namely the number of filterbank channels in this chapter) similarly defined as in Eq. (2.9), and $\mathbf{a}_{i1} = \mathbf{h}_i - \mathbf{h}_1$, $i = 1, 2$, represents the

clean speech acoustic path from the source to sensor i relative to that of the primary sensor. While $\mathbf{a}_{11} = \mathbf{0}_{\mathcal{M},1}$ is an $\mathcal{M}$-dimensional zero vector by definition, it must be observed that $\mathbf{a}_{21}$ is a relative speech gain vector in the log-Mel power spectral domain. In this chapter, $\mathbf{a}_{21}$ will be referred to as the relative acoustic path (RAP) vector.

For speech recognition purposes, we are interested in the estimation of the clean speech feature vector $\mathbf{x}_1$ derived from either the signal captured by the primary microphone or the virtual primary signal resulting from beamforming. As for the power spectrum enhancement techniques of Chapter 3, this is a reasonable choice since a clear line of sight between the source (i.e. speaker's mouth) and the front (primary) microphone/s can be assumed. Hence, we can expect that the primary signal $y_1(m)$ is less (or equally, in the worst case) affected by the noise than the secondary one, $y_2(m)$.

Under the described framework, we can estimate the clean speech feature vector $\mathbf{x}$ in two steps. First, $\mathbf{x}_1$ will be obtained by means of a dual-channel VTS estimation that benefits from the dual-channel noisy observation. Then, $\mathbf{x}$ can be estimated through the application of channel deconvolution on the clean speech estimate $\hat{\mathbf{x}}_1$. For simplicity, in this Thesis $h_1(m)$ is compensated by performing cepstral mean normalization (CMN) [10] on both training and test data. This way, we are able to cancel or, at least, mitigate the possible channel mismatches.

4.2 Dual-channel VTS feature compensation formulation

In this section we develop an MMSE-based estimator of $\mathbf{x}_1$ that exploits the dual-channel noisy observations and relies on a VTS expansion of the dual-channel speech distortion model introduced in the previous section. Our method, that performs on a frame-by-frame basis, follows a stacked formulation which exploits the spatial correlations of clean speech and noise across the two channels.

4.2.1 MMSE estimation

First, we assume that the clean speech statistics at the primary channel can be accurately modeled using a $\mathcal{K}$-component Gaussian mixture model (GMM) defined as,

$$p(\mathbf{x}_1) = \sum_{k=1}^{\mathcal{K}} P(k)\mathcal{N}\left(\mathbf{x}_1 \left| \boldsymbol{\mu}_{x_1}^{(k)}, \boldsymbol{\Sigma}_{x_1}^{(k)} \right.\right), \tag{4.3}$$

where $P(k)$ is the prior probability of the k-th multivariate Gaussian component $\mathcal{N}(\cdot)$ with mean vector and covariance matrix $\boldsymbol{\mu}_{x_1}^{(k)}$ and $\boldsymbol{\Sigma}_{x_1}^{(k)}$, respectively. By considering

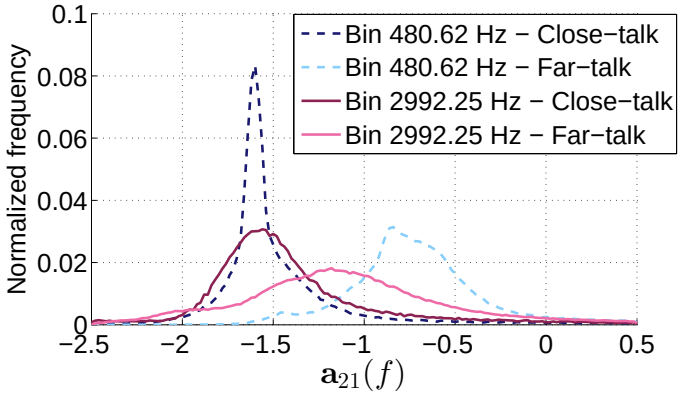

Figure 4.1: Example histograms of the variable $\mathbf{a}_{21}(f)$ at two different frequency bins for both close- and far-talk conditions. These histograms were obtained from clean speech recorded with a dual-microphone smartphone in a small and a medium-sized furnished rooms.

this speech model, the log-Mel clean speech features will be estimated at every time frame t under an MMSE approach as [64],

$$\hat{\mathbf{x}}_1 = \mathrm{E}\left[\mathbf{x}_1|\mathbf{y}\right] = \sum_{k=1}^{\mathcal{K}} P(k|\mathbf{y})\mathrm{E}\left[\mathbf{x}_1|\mathbf{y},k\right], \tag{4.4}$$

where $\mathbf{y}$ is a stacked vector defined as

$$\mathbf{y} = \begin{pmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{pmatrix}, \tag{4.5}$$

and the k-th clean speech partial estimate $\mathrm{E}\left[\mathbf{x}_1|\mathbf{y},k\right]$ is weighted by the posterior $P(k|\mathbf{y})$ to be linearly combined. In the following subsection the estimation of the posteriors $\{P(k|\mathbf{y});\ k = 1, 2, ..., \mathcal{K}\}$ is addressed while the computation of the clean speech partial estimates is detailed in Subsection 4.2.3.

4.2.2 Calculation of the posterior probabilities

Let us rewrite the speech distortion model of Eq. (4.1) by taking into account the relationship in (4.2) as

$$\begin{aligned} \mathbf{y}_i = \mathbf{f}(\mathbf{x}_1, \mathbf{a}_{i1}, \mathbf{n}_i) &= \log\left(e^{\mathbf{a}_{i1}+\mathbf{x}_1} + e^{\mathbf{n}_i}\right) \\ &= \mathbf{x}_1 + \mathbf{a}_{i1} + \log(\mathbf{1}_{\mathcal{M},1} + e^{\mathbf{n}_i-\mathbf{x}_1-\mathbf{a}_{i1}}), \end{aligned} \tag{4.6}$$

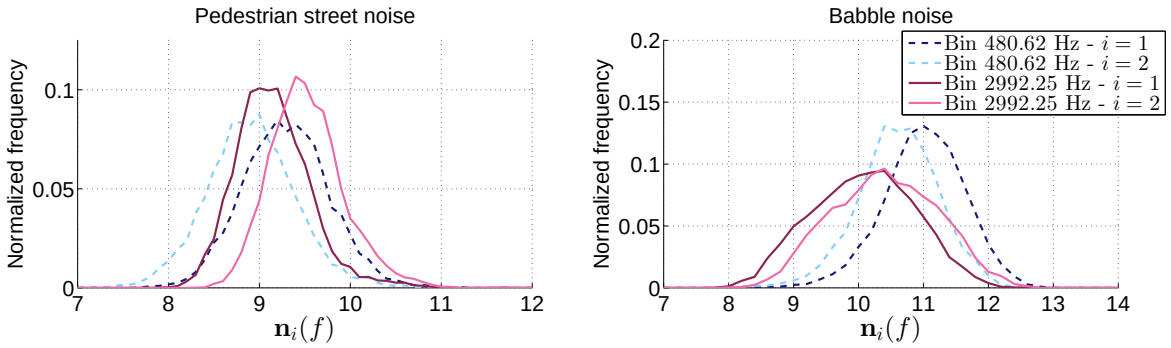

Figure 4.2: Example histograms of the variable $\mathbf{n}_i(f)$ $(i = 1, 2)$ at two different frequency bins. These histograms are calculated for two types of noise recorded with a dual-microphone smartphone: pedestrian street (left) and babble (right) noise.

where $\mathbf{f}(\mathbf{x}_1, \mathbf{a}_{i1}, \mathbf{n}_i) : \mathbb{R}^{\mathcal{M}\times\mathcal{M}\times\mathcal{M}} \rightarrow \mathbb{R}^{\mathcal{M}}$ and $\mathbf{y}_i$, $\mathbf{x}_1$, $\mathbf{a}_{i1}$ and $\mathbf{n}_i$ are the log-Mel feature vectors at time frame t introduced in the previous section, and $\mathbf{1}_{\mathcal{M},1}$ is an $\mathcal{M}$-dimensional vector filled with ones.

From now on, let $\mathbf{a} = (\mathbf{a}_{11}^\top, \mathbf{a}_{21}^\top)^\top = (\mathbf{0}_{\mathcal{M},1}^\top, \mathbf{a}_{21}^\top)^\top$ and $\mathbf{n} = (\mathbf{n}_1^\top, \mathbf{n}_2^\top)^\top$ be an augmented RAP vector and a stacked vector of noise, respectively, both of them $2\mathcal{M}$-dimensional. By taking into account the couple of sensors in the mobile device, the considered dual-channel distortion model is given by the following stacked vector:

$$\mathbf{y} = \begin{pmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{pmatrix} = \mathbf{F}\left(\mathbf{x}_1, \mathbf{a}, \mathbf{n}\right) = \begin{pmatrix} \mathbf{f}(\mathbf{x}_1, \mathbf{a}_{11}, \mathbf{n}_1) \\ \mathbf{f}(\mathbf{x}_1, \mathbf{a}_{21}, \mathbf{n}_2) \end{pmatrix}, \tag{4.7}$$

where $\mathbf{F}(\mathbf{x}_1, \mathbf{a}, \mathbf{n}) : \mathbb{R}^{\mathcal{M}\times 2\mathcal{M}\times 2\mathcal{M}} \rightarrow \mathbb{R}^{2\mathcal{M}}$. We assumed in (4.3) that the clean speech statistics at the primary channel are modeled by means of a $\mathcal{K}$-component GMM. To complete the generative model, we assume that the statistics for both the RAP and noise in each channel can be modeled by Gaussian distributions [49, 138], i.e. $p(\mathbf{a}_{21}) = \mathcal{N}\left(\mathbf{a}_{21} \left| \boldsymbol{\mu}_{a_{21}}, \boldsymbol{\Sigma}_{a_{21}}\right.\right)$ and $p(\mathbf{n}_i) = \mathcal{N}\left(\mathbf{n}_i \left| \boldsymbol{\mu}_{n_i}, \boldsymbol{\Sigma}_{n_i}\right.\right)$ $(i = 1, 2)$, respectively. To support this, Figures 4.1 and 4.2 plot example histograms of $\mathbf{a}_{21}(f)$ and $\mathbf{n}_i(f)$ $(i = 1, 2)$, respectively, at two different frequency bins. From those figures, we can state that the Gaussian assumption seems reasonable. Since any linear combination of Gaussian variables follows another Gaussian distribution [150], by linearizing the dual-channel distortion model in (4.7) we are able to describe the dual-channel noisy speech statistics (required to compute the posteriors $\{P(k|\mathbf{y});\, k = 1, 2, ..., \mathcal{K}\}$) by means of a GMM (at every time frame t) as

$$p(\mathbf{y}) = \sum_{k=1}^{\mathcal{K}} P(k)\mathcal{N}\left(\mathbf{y} \left| \boldsymbol{\mu}_y^{(k)}, \boldsymbol{\Sigma}_y^{(k)}\right.\right). \tag{4.8}$$

Then, we linearize $\mathbf{y} = \mathbf{F}(\mathbf{x}_1, \mathbf{a}, \mathbf{n})$ by means of a first-order VTS expansion around the point $\left(\boldsymbol{\mu}_{x_1}^{(k)}, \boldsymbol{\mu}_a, \boldsymbol{\mu}_n\right)$, where

$$\boldsymbol{\mu}_a = \begin{pmatrix} \boldsymbol{\mu}_{a_{11}} \\ \boldsymbol{\mu}_{a_{21}} \end{pmatrix} = \begin{pmatrix} \mathbf{0}_{\mathcal{M},1} \\ \boldsymbol{\mu}_{a_{21}} \end{pmatrix} \tag{4.9}$$

and

$$\boldsymbol{\mu}_n = \begin{pmatrix} \boldsymbol{\mu}_{n_1} \\ \boldsymbol{\mu}_{n_2} \end{pmatrix} \tag{4.10}$$

are $2\mathcal{M} \times 1$ vectors of stacked means. This procedure is accomplished by accordingly linearizing the speech distortion model for each channel, $\mathbf{f}(\mathbf{x}_1, \mathbf{a}_{i1}, \mathbf{n}_i)$ $(i = 1, 2)$, around the point $\left(\boldsymbol{\mu}_{x_1}^{(k)}, \boldsymbol{\mu}_{a_{i1}}, \boldsymbol{\mu}_{n_i}\right)$, that is,

$$\begin{aligned} \mathbf{f}(\mathbf{x}_1, \mathbf{a}_{i1}, \mathbf{n}_i) \;\approx\; & \mathbf{f}\left(\boldsymbol{\mu}_{x_1}^{(k)}, \boldsymbol{\mu}_{a_{i1}}, \boldsymbol{\mu}_{n_i}\right) + \mathbf{J}_x^{(i,k)}\left(\mathbf{x}_1 - \boldsymbol{\mu}_{x_1}^{(k)}\right) \\ & + \mathbf{J}_a^{(i,k)}\left(\mathbf{a}_{i1} - \boldsymbol{\mu}_{a_{i1}}\right) + \mathbf{J}_n^{(i,k)}\left(\mathbf{n}_i - \boldsymbol{\mu}_{n_i}\right), \end{aligned} \tag{4.11}$$

where $\mathbf{J}_x^{(i,k)}$, $\mathbf{J}_a^{(i,k)}$ and $\mathbf{J}_n^{(i,k)}$ are $\mathcal{M} \times \mathcal{M}$ Jacobian matrices, the calculation of which will be detailed later.

To finally characterize the probability density function (PDF) $p(\mathbf{y})$ we need to derive its mean vectors and covariance matrices. By taking into account (4.7) and (4.11), it is straightforward to show that the mean vectors $\left\{\boldsymbol{\mu}_y^{(k)} : k = 1, 2, ..., \mathcal{K}\right\}$ can be obtained as

$$\boldsymbol{\mu}_y^{(k)} = \begin{pmatrix} \mathrm{E}\left[\mathbf{y}_1 | k\right] \\ \mathrm{E}\left[\mathbf{y}_2 | k\right] \end{pmatrix} = \begin{pmatrix} \mathbf{f}\left(\boldsymbol{\mu}_{x_1}^{(k)}, \boldsymbol{\mu}_{a_{11}}, \boldsymbol{\mu}_{n_1}\right) \\ \mathbf{f}\left(\boldsymbol{\mu}_{x_1}^{(k)}, \boldsymbol{\mu}_{a_{21}}, \boldsymbol{\mu}_{n_2}\right) \end{pmatrix}. \tag{4.12}$$

On the other hand, the covariance matrices can be easily calculated in accordance to their definition as

$$\boldsymbol{\Sigma}_y^{(k)} = \mathrm{E}\left[\left(\mathbf{y} - \boldsymbol{\mu}_y^{(k)}\right)\left(\mathbf{y} - \boldsymbol{\mu}_y^{(k)}\right)^\top\right], \tag{4.13}$$

where $\mathbf{y} - \boldsymbol{\mu}_y^{(k)}$ is defined in the following manner by again considering the approximation in (4.11) as well as (4.12):

$$\mathbf{y} - \boldsymbol{\mu}_y^{(k)} = \begin{pmatrix} \mathbf{J}_x^{(1,k)}\left(\mathbf{x}_1 - \boldsymbol{\mu}_{x_1}^{(k)}\right) + \mathbf{J}_a^{(1,k)}\left(\mathbf{a}_{11} - \boldsymbol{\mu}_{a_{11}}\right) + \mathbf{J}_n^{(1,k)}\left(\mathbf{n}_1 - \boldsymbol{\mu}_{n_1}\right) \\ \mathbf{J}_x^{(2,k)}\left(\mathbf{x}_1 - \boldsymbol{\mu}_{x_1}^{(k)}\right) + \mathbf{J}_a^{(2,k)}\left(\mathbf{a}_{21} - \boldsymbol{\mu}_{a_{21}}\right) + \mathbf{J}_n^{(2,k)}\left(\mathbf{n}_2 - \boldsymbol{\mu}_{n_2}\right) \end{pmatrix}. \tag{4.14}$$

For notational convenience let us define the following block Jacobian matrices:

$$\mathbf{J}_x^{(k)} = \begin{pmatrix} \mathbf{J}_x^{(1,k)} \\ \mathbf{J}_x^{(2,k)} \end{pmatrix}; \tag{4.15}$$

$$\mathbf{J}_a^{(k)} = \begin{pmatrix} \mathbf{J}_a^{(1,k)} & \mathbf{0}_{\mathcal{M},\mathcal{M}} \\ \mathbf{0}_{\mathcal{M},\mathcal{M}} & \mathbf{J}_a^{(2,k)} \end{pmatrix}; \tag{4.16}$$

$$\mathbf{J}_n^{(k)} = \begin{pmatrix} \mathbf{J}_n^{(1,k)} & \mathbf{0}_{\mathcal{M},\mathcal{M}} \\ \mathbf{0}_{\mathcal{M},\mathcal{M}} & \mathbf{J}_n^{(2,k)} \end{pmatrix}, \tag{4.17}$$

where $\mathbf{J}_x^{(k)}$ is a $2\mathcal{M} \times \mathcal{M}$ matrix, $\mathbf{J}_a^{(k)}$ and $\mathbf{J}_n^{(k)}$ are $2\mathcal{M} \times 2\mathcal{M}$ matrices and $\mathbf{0}_{\mathcal{M},\mathcal{M}}$ is an $\mathcal{M} \times \mathcal{M}$ zero matrix. Then, (4.14) can be expressed in a more compact form as

$$\mathbf{y} - \boldsymbol{\mu}_y^{(k)} = \mathbf{J}_x^{(k)} \left(\mathbf{x}_1 - \boldsymbol{\mu}_{x_1}^{(k)} \right) + \mathbf{J}_a^{(k)} \left(\mathbf{a} - \boldsymbol{\mu}_a \right) + \mathbf{J}_n^{(k)} \left(\mathbf{n} - \boldsymbol{\mu}_n \right). \tag{4.18}$$

Finally, by combining (4.18) and (4.13), as well as considering independence between clean speech, the RAP and noise, an expression for the dual-channel noisy speech model covariance matrix can be obtained as

$$\boldsymbol{\Sigma}_y^{(k)} = \mathbf{J}_x^{(k)} \boldsymbol{\Sigma}_{x_1}^{(k)} \mathbf{J}_x^{(k)^\top} + \mathbf{J}_a^{(k)} \boldsymbol{\Sigma}_a \mathbf{J}_a^{(k)^\top} + \mathbf{J}_n^{(k)} \boldsymbol{\Sigma}_n \mathbf{J}_n^{(k)^\top}, \tag{4.19}$$

where

$$\boldsymbol{\Sigma}_a = \mathrm{E}\left[(\mathbf{a} - \boldsymbol{\mu}_a)(\mathbf{a} - \boldsymbol{\mu}_a)^\top \right] = \begin{pmatrix} \mathbf{0}_{\mathcal{M},\mathcal{M}} & \mathbf{0}_{\mathcal{M},\mathcal{M}} \\ \mathbf{0}_{\mathcal{M},\mathcal{M}} & \boldsymbol{\Sigma}_{a_{21}} \end{pmatrix} \tag{4.20}$$

and

$$\boldsymbol{\Sigma}_n = \mathrm{E}\left[(\mathbf{n} - \boldsymbol{\mu}_n)(\mathbf{n} - \boldsymbol{\mu}_n)^\top \right] = \begin{pmatrix} \boldsymbol{\Sigma}_{n_1} & \boldsymbol{\Sigma}_{n_{12}} \\ \boldsymbol{\Sigma}_{n_{21}} & \boldsymbol{\Sigma}_{n_2} \end{pmatrix} \tag{4.21}$$

are $2\mathcal{M} \times 2\mathcal{M}$ spatial covariance matrices of the RAP and noise, respectively. In addition, $\boldsymbol{\Sigma}_{n_{12}} = \boldsymbol{\Sigma}_{n_{21}}^\top = \mathrm{E}\left[(\mathbf{n}_1 - \boldsymbol{\mu}_{n_1})(\mathbf{n}_2 - \boldsymbol{\mu}_{n_2})^\top \right]$. The Jacobian matrices, which are diagonal in accordance to the speech distortion model described by Eq. (4.6) (independent frequency components), are easily calculated by employing the Jacobian matrix mathematical definition as,

$$\begin{aligned}
\mathbf{J}_x^{(i,k)} &= \left. \frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_1} \right|_{\boldsymbol{\mu}_{x_1}^{(k)}, \boldsymbol{\mu}_{a_{i1}}, \boldsymbol{\mu}_{n_i}} = \mathrm{diag}\left(\frac{\mathbf{1}_{\mathcal{M},1}}{\mathbf{1}_{\mathcal{M},1} + e^{\boldsymbol{\mu}_{n_i} - \boldsymbol{\mu}_{x_1}^{(k)} - \boldsymbol{\mu}_{a_{i1}}}} \right); \\
\mathbf{J}_a^{(i,k)} &= \left. \frac{\partial \mathbf{y}_i}{\partial \mathbf{a}_{i1}} \right|_{\boldsymbol{\mu}_{x_1}^{(k)}, \boldsymbol{\mu}_{a_{i1}}, \boldsymbol{\mu}_{n_i}} = \begin{cases} \mathbf{0}_{\mathcal{M},\mathcal{M}} & \text{if } i = 1 \\ \mathbf{J}_x^{(2,k)} & \text{if } i = 2 \end{cases}; \\
\mathbf{J}_n^{(i,k)} &= \left. \frac{\partial \mathbf{y}_i}{\partial \mathbf{n}_i} \right|_{\boldsymbol{\mu}_{x_1}^{(k)}, \boldsymbol{\mu}_{a_{i1}}, \boldsymbol{\mu}_{n_i}} = \mathbf{I}_{\mathcal{M}} - \mathbf{J}_x^{(i,k)},
\end{aligned} \tag{4.22}$$

where $\mathrm{diag}(\cdot)$ indicates a diagonal matrix whose main diagonal corresponds to its argument, division $\div$ operates element-wise and $\mathbf{I}_{\mathcal{M}}$ is an $\mathcal{M} \times \mathcal{M}$ identity matrix.

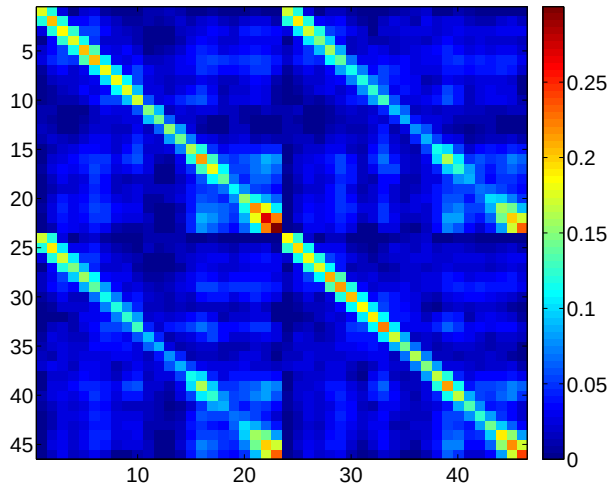

Figure 4.3: Example of noise spatial covariance matrix $\mathbf{\Sigma}_n$ estimated from 12 seconds of pedestrian street noise captured with a dual-microphone smartphone.

Finally, by using the Bayes' rule and the previous derivations, in the knowledge that $p(\mathbf{y}|k) = \mathcal{N}\left(\mathbf{y}\left|\boldsymbol{\mu}_y^{(k)}, \mathbf{\Sigma}_y^{(k)}\right.\right)$, the posteriors are obtained as

$$P(k|\mathbf{y}) = \frac{p(\mathbf{y}|k)P(k)}{\sum_{k'=1}^{\mathcal{K}} p(\mathbf{y}|k')P(k')}, \qquad k = 1, 2, ..., \mathcal{K}. \tag{4.23}$$

On the one hand, since the computation of the parameters of the PDFs $p(\mathbf{x}_1)$ and $p(\mathbf{a}_{21})$, required to perform the calculations above, depends on the experimental dataset (or mobile device), that issue is detailed later in Subsection 6.2.3. On the other hand, the parameters of $p(\mathbf{n}_i)$, $i = 1, 2$, along with $\mathbf{\Sigma}_{n_{12}}$ are obtained as follows. As for the noise statistics calculation procedures considered throughout Chapter 3, we assume again that the first and last M frames of each utterance contain only noise energy. In this way, the mean vector of the PDF $p(\mathbf{n}_i)$, $\boldsymbol{\mu}_{n_i}$ $(i = 1, 2)$, is computed for every time frame t from a linear interpolation between the averages of the first and last M frames in the i-th channel of each utterance in the log-Mel domain [65]. Additionally, the noise covariance matrices $\mathbf{\Sigma}_{n_i}$ $(i = 1, 2)$ and $\mathbf{\Sigma}_{n_{12}}$ are estimated per utterance as the sample covariance of the first and last M frames as well [65]. Independence across frequency bins is also assumed for the noise so that $\mathbf{\Sigma}_{n_i}$ $(i = 1, 2)$ and $\mathbf{\Sigma}_{n_{12}}$ are diagonal. Figure 4.3 shows an example of a noise spatial covariance matrix $\mathbf{\Sigma}_n$ estimated from 12 seconds of pedestrian street noise captured with a dual-microphone smartphone. This example illustrates the suitability of the diagonal assumption.

4.2.2.1 Alternative approach

A main feature of the stacked formulation described above is that the secondary channel is treated in a parallel manner to the primary one, using similar distortion models. Thus, as a result, the two-channel joint information is indirectly exploited by means of the spatial covariance matrix of noise and a term modeling the clean speech RAP between the two sensors of the device. However, as we know, clean speech will be easily masked by noise at the secondary channel, and, therefore, we can expect that the relation between the secondary noisy observation and the clean speech be more uncertain than that of the primary channel. We have found more robust conditioning this distortion model at the secondary channel to the certain noisy observation from the primary channel since both channels are heavily correlated, decreasing the influence of the clean speech variable. This is accomplished by replacing $P(k|\mathbf{y})$ in (4.4) by $P\left(k|\mathbf{y}_1, \mathbf{y}_2\right)$, which is further decomposed as the product of an *a priori* and a conditional PDF. This alternative posterior computation approach is formulated immediately below.

The posteriors $\{P\left(k|\mathbf{y}_1, \mathbf{y}_2\right); k = 1, 2, ..., \mathcal{K}\}$ can be calculated by employing again the Bayes' theorem as

$$P\left(k|\mathbf{y}_1, \mathbf{y}_2\right) = \frac{p\left(\mathbf{y}_1, \mathbf{y}_2|k\right) P(k)}{\sum_{k'=1}^{\mathcal{K}} p\left(\mathbf{y}_1, \mathbf{y}_2|k'\right) P(k')}, \tag{4.24}$$

where the PDF $p\left(\mathbf{y}_1, \mathbf{y}_2|k\right)$ can be factored as

$$p\left(\mathbf{y}_1, \mathbf{y}_2|k\right) = p(\mathbf{y}_1|k)p(\mathbf{y}_2|\mathbf{y}_1, k). \tag{4.25}$$

Then, by using a VTS approach [138], both $p(\mathbf{y}_1|k)$ and $p(\mathbf{y}_2|\mathbf{y}_1, k)$ will be similarly modeled as Gaussian PDFs and their parameters are obtained as described in the following.

First, the speech distortion model of (4.1) is adapted to the primary and secondary channels, respectively, as

$$\mathbf{y}_1 = \mathbf{x}_1 + \log\left(\mathbf{1}_{\mathcal{M},1} + e^{\mathbf{n}_1 - \mathbf{x}_1}\right); \tag{4.26}$$

$$\mathbf{y}_2 = \mathbf{x}_1 + \mathbf{a}_{21} + \log\left(\mathbf{1}_{\mathcal{M},1} + e^{\mathbf{n}_2 - \mathbf{x}_1 - \mathbf{a}_{21}}\right), \tag{4.27}$$

which are combined to define an alternative speech distortion model for the secondary channel given $\mathbf{y}_1$, as,

$$\mathbf{y}_2(\mathbf{y}_1) = \mathbf{y}_1 + \mathbf{a}_{21} + \log\left[\frac{\mathbf{1}_{\mathcal{M},1} + e^{\mathbf{n}_2 - \mathbf{x}_1 - \mathbf{a}_{21}}}{\mathbf{1}_{\mathcal{M},1} + e^{\mathbf{n}_1 - \mathbf{x}_1}}\right]. \tag{4.28}$$

Assuming again that all $\mathbf{a}_{21}$ and $\mathbf{n}_i$ $(i = 1, 2)$ can be modeled by Gaussian distributions [49, 138], Eqs. (4.26) and (4.28) are linearized by means of a first-order VTS

expansion to obtain the parameters (i.e. mean vectors and covariance matrices) of $p(\mathbf{y}_1|k) = \mathcal{N}\left(\boldsymbol{\mu}_{y_1}^{(k)}, \boldsymbol{\Sigma}_{y_1}^{(k)}\right)$ and $p(\mathbf{y}_2|\mathbf{y}_1, k) = \mathcal{N}\left(\boldsymbol{\mu}_{y_2|y_1}^{(k)}, \boldsymbol{\Sigma}_{y_2|y_1}^{(k)}\right)$, respectively. By following a procedure similar to that from the stacked case, it is straightforward to demonstrate that the $\mathcal{M}$-dimensional mean vectors are given by

$$\begin{aligned} \boldsymbol{\mu}_{y_1}^{(k)} &= \boldsymbol{\mu}_{x_1}^{(k)} + \log\left(\mathbf{1}_{\mathcal{M},1} + e^{\boldsymbol{\mu}_{n_1} - \boldsymbol{\mu}_{x_1}^{(k)}}\right); \\ \boldsymbol{\mu}_{y_2|y_1}^{(k)} &= \mathbf{y}_1 + \boldsymbol{\mu}_{a_{21}} + \log\left[\frac{\mathbf{1}_{\mathcal{M},1} + e^{\boldsymbol{\mu}_{n_2} - \boldsymbol{\mu}_{x_1}^{(k)} - \boldsymbol{\mu}_{a_{21}}}}{\mathbf{1}_{\mathcal{M},1} + e^{\boldsymbol{\mu}_{n_1} - \boldsymbol{\mu}_{x_1}^{(k)}}}\right]. \end{aligned} \tag{4.29}$$

Analogously, it is easy to show that the covariance matrix of $p(\mathbf{y}_1|k)$, considering independence between clean speech and noise, can be approximated as

$$\boldsymbol{\Sigma}_{y_1}^{(k)} = \mathbf{J}_{x_1}^{(1,k)} \boldsymbol{\Sigma}_{x_1}^{(k)} \mathbf{J}_{x_1}^{(1,k)^\top} + \mathbf{J}_{n_1}^{(1,k)} \boldsymbol{\Sigma}_{n_1} \mathbf{J}_{n_1}^{(1,k)^\top}. \tag{4.30}$$

The $\mathcal{M} \times \mathcal{M}$ Jacobian matrices have the following definitions:

$$\begin{aligned} \mathbf{J}_{x_1}^{(1,k)} &= \left.\frac{\partial \mathbf{y}_1}{\partial \mathbf{x}_1}\right|_{\boldsymbol{\mu}_{x_1}^{(k)}, \boldsymbol{\mu}_{n_1}} = \operatorname{diag}\left(\frac{\mathbf{1}_{\mathcal{M},1}}{\mathbf{1}_{\mathcal{M},1} + e^{\boldsymbol{\mu}_{n_1} - \boldsymbol{\mu}_{x_1}^{(k)}}}\right); \\ \mathbf{J}_{n_1}^{(1,k)} &= \left.\frac{\partial \mathbf{y}_1}{\partial \mathbf{n}_1}\right|_{\boldsymbol{\mu}_{x_1}^{(k)}, \boldsymbol{\mu}_{n_1}} = \mathbf{I}_{\mathcal{M}} - \mathbf{J}_{x_1}^{(1,k)}. \end{aligned} \tag{4.31}$$

Similarly, the covariance matrix of the conditional PDF $p\left(\mathbf{y}_2|\mathbf{y}_1, k\right)$, assuming independence between clean speech, the RAP and noise, is estimated as

$$\begin{aligned} \boldsymbol{\Sigma}_{y_2|y_1}^{(k)} &= \mathbf{J}_{x_1}^{(2,k)} \boldsymbol{\Sigma}_{x_1}^{(k)} \mathbf{J}_{x_1}^{(2,k)^\top} + \mathbf{J}_{a_{21}}^{(2,k)} \boldsymbol{\Sigma}_{a_{21}} \mathbf{J}_{a_{21}}^{(2,k)^\top} + \mathbf{J}_{n_1}^{(2,k)} \boldsymbol{\Sigma}_{n_1} \mathbf{J}_{n_1}^{(2,k)^\top} \\ &\quad + \mathbf{J}_{n_2}^{(2,k)} \boldsymbol{\Sigma}_{n_2} \mathbf{J}_{n_2}^{(2,k)^\top} + \mathbf{J}_{n_1}^{(2,k)} \boldsymbol{\Sigma}_{n_{12}} \mathbf{J}_{n_2}^{(2,k)^\top} + \mathbf{J}_{n_2}^{(2,k)} \boldsymbol{\Sigma}_{n_{21}} \mathbf{J}_{n_1}^{(2,k)^\top}. \end{aligned} \tag{4.32}$$

The corresponding $\mathcal{M} \times \mathcal{M}$ Jacobian matrices are calculated in a similar way as in (4.31) as follows:

$$\begin{aligned} \mathbf{J}_{x_1}^{(2,k)} &= \operatorname{diag}\left(\frac{e^{\boldsymbol{\mu}_{n_1} - \boldsymbol{\mu}_{x_1}^{(k)}} - e^{\boldsymbol{\mu}_{n_2} - \boldsymbol{\mu}_{x_1}^{(k)} - \boldsymbol{\mu}_{a_{21}}}}{\left(\mathbf{1}_{\mathcal{M},1} + e^{\boldsymbol{\mu}_{n_1} - \boldsymbol{\mu}_{x_1}^{(k)}}\right) \odot \left(\mathbf{1}_{\mathcal{M},1} + e^{\boldsymbol{\mu}_{n_2} - \boldsymbol{\mu}_{x_1}^{(k)} - \boldsymbol{\mu}_{a_{21}}}\right)}\right); \\ \mathbf{J}_{a_{21}}^{(2,k)} &= \operatorname{diag}\left(\frac{\mathbf{1}_{\mathcal{M},1}}{\mathbf{1}_{\mathcal{M},1} + e^{\boldsymbol{\mu}_{n_2} - \boldsymbol{\mu}_{x_1}^{(k)} - \boldsymbol{\mu}_{a_{21}}}}\right); \\ \mathbf{J}_{n_1}^{(2,k)} &= -\mathbf{J}_{n_1}^{(1,k)}; \\ \mathbf{J}_{n_2}^{(2,k)} &= \mathbf{I}_{\mathcal{M}} - \mathbf{J}_{a_{21}}^{(2,k)}. \end{aligned} \tag{4.33}$$

It should be reminded that, to perform the above calculations, the parameters of both $p(\mathbf{x}_1)$ and $p(\mathbf{a}_{21})$ depend on the experimental dataset (or mobile device), so their estimation is detailed in Subsection 6.2.3. Additionally, the parameters of $p(\mathbf{n}_i)$ $(i = 1, 2)$ plus $\mathbf{\Sigma}_{n_{12}}$ are obtained in the same manner as for the stacked case.

4.2.3 Clean speech partial estimate computation

The partial expected values in (4.4) are defined as

$$\mathrm{E}\left[\mathbf{x}_1|\mathbf{y}, k\right] = \int \mathbf{x}_1 p(\mathbf{x}_1|\mathbf{y}, k) d\mathbf{x}_1, \quad k = 1, 2, ..., \mathcal{K}. \tag{4.34}$$

In order to compute them, it is again necessary to linearize the non-linear speech distortion model of (4.6) to make inference feasible. In this regard, two different proposals for VTS feature compensation are considered in this Thesis.

In the first approach, we exploit the stacked dual-channel information as follows. If we assume that the joint PDF $p(\mathbf{x}_1, \mathbf{y}|k)$ is Gaussian then the conditional PDF $p(\mathbf{x}_1|\mathbf{y}, k)$ will also be Gaussian, so that the expected value of $p(\mathbf{x}_1|\mathbf{y}, k)$, $\mathrm{E}\left[\mathbf{x}_1|\mathbf{y}, k\right]$, can be approximated as [176]

$$\mathrm{E}\left[\mathbf{x}_1|\mathbf{y}, k\right] = \boldsymbol{\mu}_{x_1}^{(k)} + \mathbf{\Sigma}_{x_1 y}^{(k)} {\mathbf{\Sigma}_{y}^{(k)}}^{-1} \left(\mathbf{y} - \boldsymbol{\mu}_{y}^{(k)}\right), \tag{4.35}$$

where the cross-covariance matrix $\mathbf{\Sigma}_{x_1 y}^{(k)}$ is approximated by again considering a VTS approach. Thus, by using the result in (4.18),

$$\mathbf{\Sigma}_{x_1 y}^{(k)} = \mathrm{E}\left[\left(\mathbf{x}_1 - \boldsymbol{\mu}_{x_1}^{(k)}\right)\left(\mathbf{y} - \boldsymbol{\mu}_{y}^{(k)}\right)^{\top}\right] = \mathbf{\Sigma}_{x_1}^{(k)} {\mathbf{J}_{x}^{(k)}}^{\top}, \tag{4.36}$$

where it should be reminded that independence between clean speech, the RAP and noise was assumed.

In the second approach, only the information from the primary channel is used to compute the clean speech partial estimates. For this second strategy, Eq. (4.6) is rewritten as $\mathbf{y}_i = \mathbf{x}_1 + \mathbf{g}(\mathbf{x}_1, \mathbf{a}_{i1}, \mathbf{n}_i)$ [137, 159], where $\mathbf{g}(\mathbf{x}_1, \mathbf{a}_{i1}, \mathbf{n}_i) = \mathbf{a}_{i1} + \log(\mathbf{1}_{\mathcal{M},1} + e^{\mathbf{n}_i - \mathbf{x}_1 - \mathbf{a}_{i1}})$ is a distortion vector. Then, the k-th clean speech partial estimate is calculated as

$$\mathrm{E}\left[\mathbf{x}_1|\mathbf{y}, k\right] \approx \mathrm{E}\left[\mathbf{x}_1|\mathbf{y}_1, k\right] = \mathbf{y}_1 - \mathrm{E}[\mathbf{g}(\mathbf{x}_1, \mathbf{a}_{11}, \mathbf{n}_1)|\mathbf{y}_1, k], \tag{4.37}$$

where it is assumed that the function $\mathbf{g}(\mathbf{x}_1, \mathbf{a}_{i1}, \mathbf{n}_i)$ is smooth for each k such that [137, 159]

$$\mathrm{E}[\mathbf{g}(\mathbf{x}_1, \mathbf{a}_{11}, \mathbf{n}_1)|\mathbf{y}_1, k] = \mathbf{g}\left(\boldsymbol{\mu}_{x_1}^{(k)}, \boldsymbol{\mu}_{a_{11}}, \boldsymbol{\mu}_{n_1}\right). \tag{4.38}$$

Considering this latter strategy may be more appropriate in our case, since, generally, the secondary microphone of the mobile device captures a much noisier signal than the front (primary) one/s.

4.3 Summary

An extension towards a dual-channel framework has been developed in this chapter for VTS feature compensation. As for the previous dual-channel power spectrum enhancement techniques presented in Chapter 3, the beamforming-based combinatorial strategy is considered when this dual-channel VTS feature compensation method is applied on a mobile device with more than one front (primary) microphone, or more than two microphones of any type.

The overarching element of this dual-channel VTS method has been the stacked formulation. From this, an MMSE-based estimator for the log-Mel clean speech features, which relies on a VTS expansion of a dual-channel speech distortion model, has been developed. In particular, by taking advantage of the dual-channel information, this method estimates the log-Mel clean speech features at the primary channel, since it is expected that it is less affected by the ambient noise than the secondary one.

As we have seen, the MMSE-based estimator linearly combines a set of clean speech partial estimates which are weighted by other set of posterior probabilities. Two different approaches have been studied for the computation of each set of parameters. In the case of the posteriors, their VTS-based derivation from the stacked formulation has been carried out in the first place. As a result of this scheme, the two-channel joint information is indirectly exploited by means of the spatial covariance matrix of noise and a term modeling the clean speech relative acoustic path (RAP) between the two channels of the device. Then, a more robust strategy consisting of the modeling of the conditional dependence of the noisy secondary channel given the primary one was developed to explicitly exploit the correlations between the two channels. On the other hand, a simpler scheme has also been raised for clean speech partial estimate computation. In contrast to the original stacked formulation, it only takes into account the primary channel instead of the dual-channel information since it is expected that the secondary signal is usually noisier and may degrade the partial estimates.

CHAPTER 5

Dual-Channel Deep Learning-Based Techniques

It is difficult to agree on a single definition for deep learning. Nevertheless, we can consider deep learning as a branch of machine learning dealing with graph architectures called networks containing multiple layers of non-linear transformations which are used for high-level data modeling purposes. In particular, this kind of models are generally known as deep neural networks (DNN). In contrast to some other shallow architectures employed in signal processing such as hidden Markov models (HMMs) or artificial neural networks (ANNs) with only one hidden layer [206], it is implicitly established that a neural network must have at least three hidden layers to be considered as deep. Nonetheless, we can find some examples in the literature where ANNs with only two hidden layers are referred to as DNNs, e.g. [139, 193]. Anyway, the most important thing is that a DNN architecture, along with the set of related methods (e.g. supervised and unsupervised training methods), is able to learn complex non-linear dependencies among the underlying data to overcome the modeling capabilities of the classical analytical approaches.

The classical signal processing solutions have inherent constraints because they are based on analytical functions and statistical models, which do their best to approximate the underlying natural phenomena. For instance, classical noise-robust automatic speech recognition (ASR) and speech enhancement techniques make assumptions in order to make the problem analytically tractable at the expense of a drop in performance. As we know, it is usual that this kind of techniques relies on additive and convolutive distortion models [149] which are not able to correctly model the complex relationships between the human voice and the ambient distortions. Even so, those types

of distortion models are non-linear in different domains where the signal processing methods can be applied (e.g. in the log-Mel power spectral domain), in such a way that it is required to carry out linearization procedures to make the problem tractable [138]. Indeed, this was the case of the dual-channel vector Taylor series (VTS) feature compensation method developed in the previous chapter. Another typical assumption is time or frequency bin independence, which highly constraints the performance of the signal processing techniques. For instance, the spectral reconstruction technique in [65] assumes independence between frequency bins in order to provide with an analytically tractable algorithm. Moreover, those assumptions were also made in several ways when developing the dual-channel contributions presented in this Thesis throughout Chapters 3 and 4. As an alternative to the use of simplifications, numerical methods can be considered to tackle with analytically intractable problems. Thus, in [47], the Monte Carlo methods were studied to perform feature compensation in replacement of the VTS approach. Unfortunately, the use of numerical methods presents a twofold disadvantage: the techniques still rely on imprecise analytical approximations while the computational cost dramatically increases [195].

Unlike the classical signal processing solutions above mentioned, a main feature of deep learning is that no assumptions on the problem to be addressed are required. The powerful modeling capabilities of DNNs will be applied in this chapter to complex tasks (from an analytical point of view) by also taking advantage of the available dual-channel information. This synergy will allow us to achieve very accurate missing-data masks and noise estimates, with no assumptions and in an efficient manner, to be used for noise-robust ASR. Both the missing-data mask and noise estimation approaches here developed follow a similar DNN-based scheme in the log-Mel power spectral domain which exploits the power level difference (PLD) between the primary and secondary channels of a mobile device. While missing-data masks and noise estimates can be employed in multiple ways in order to provide robustness in ASR, we will use them during the experimental evaluation for spectral reconstruction and feature compensation, respectively. It should be noticed that these are hybrid DNN/signal processing architectures which will be extensively and successfully explored in the near future as it can be foreseen [186]. Before describing these dual-channel deep learning-based techniques, a brief overview of deep learning is given immediately below. This overview is focused on the theoretical fundamentals of deep feedforward neural networks, as it is the architecture considered in the following by our methods. Additionally, a brief review of the literature covering the use of deep learning for signal processing applications, especially noise-robust ASR and speech enhancement applications, is also given in the next section.

5.1 A brief overview of deep learning

Over the last years, deep learning has become very popular mainly due to two landmarks: the appearing of the so-called restricted Boltzmann machines (RBMs) in 2006 [83] and the possibility of exploiting the great computing capacities of the graphic processing units (GPUs) [37]. In particular, RBMs have been a revolution since they allow us having feature learning structures with multiple non-linear processing layers (i.e. DNNs), the modeling capabilities of which are impressive. With these tools, the great amount of parameters of a DNN can be properly set for modeling complex problems when a lot of data are available. Thus, several types of deep learning structures are being successfully applied to problems that have traditionally been addressed by the signal processing paradigm. One of them is acoustic modeling in ASR. In this regard, acoustic modeling has clearly moved towards the use of DNNs, since they outperform the modeling capabilities of traditional Gaussian mixture models (GMMs) [82, 154, 196]. As it was introduced in Subsection 1.2.2, in DNN-HMM-based ASR systems, DNNs are employed to estimate the posterior probabilities of the HMM states from several frames of speech feature coefficients [82]. Indeed, deep learning is applied to manifold tasks such as classification, prediction or regression problems. It is also applied to the obtainment of features [193] (as seen in Subsection 2.2.1) as well as it is used in image (e.g. optical character and object recognition or inpainting [200]), audio (e.g. transcription or music composition [98]) or text applications (e.g. semantic object parsing [111]).

Deep learning architectures for signal processing applications can be used for the resolution of either the whole problem or one of several of its parts, and in the literature we can find examples of both types of approaches. In the following, let us take a look at some deep learning-based methods for speech enhancement to illustrate both types of approaches. Thus, in [124, 201, 202], deep architectures implement the whole speech enhancement system by directly estimating the clean speech spectrum from its noisy version within a single-channel context. While a denoising auto-encoder (DA) [188] is used in [124] for this purpose, deep feedforward neural networks are considered in [201, 202]. It is worth to note that in these latter works, apart from the noisy speech spectrum, a noise estimate is also used as input to the DNN to improve the performance. This noise-aware training (NAT) strategy first appeared in [163] for noise-robust acoustic modeling purposes. In fact, we should emphasize the importance of the features chosen as input for the deep learning approaches, since their success many times depends on them. Additionally, [201, 202] also demonstrate how the modeling is improved in terms of PESQ (Perceptual Evaluation of Speech Quality) when increasing

the depth of the ANN. On the contrary, a quite interesting speech enhancement method using deep learning to partially solve the problem is the one reported in [19]. In this work, a weighted denoising auto-encoder (WDA) estimates the clean speech power spectral density (PSD). Since one of the constraints of this kind of deep architectures is their generalization ability to unknown data distributions, distinct WDAs are trained under different noise conditions. Then, a GMM-based noise classifier selects the WDA that fits the best to the inferred environmental condition. The clean speech PSD estimated from this procedure is used along a noise PSD independently estimated by means of a classical algorithm to finally define a Wiener filter (WF) for enhancement. The philosophy of this work is based on keeping the knowledge framework provided by the signal processing paradigm while integrating a machine learning approach for the more complex system stages.

As was introduced in Chapter 2, a critical aspect of a number of ASR systems is the calculation of time-frequency (T-F) masks to classify every T-F bin of a noisy spectrum in one of two categories: one where speech dominates and another where noise prevails. The computation of this kind of masks is highly difficult, and a great amount of signal processing techniques depends on the accuracy of these masks to achieve an appropriate performance. Hence, it is no wonder that a number of deep learning-based solutions can be found in the literature to address T-F mask estimation. Let us take a look at the following works which deal with mask estimation in a single-channel context. In [106], the noisy spectrum at the output of a filterbank is used as features for DNN-based soft-mask estimation. Then, this soft-mask weights the noisy power spectrum to enhance it for noise-robust ASR. A similar strategy is considered in [139], while a different set of features is employed. Of course, the choice of the type of features is a fundamental issue in deep learning-based approaches. In this respect, the mask estimation method reported in [193] is of special interest since a DNN is only used to generate more discriminative and linearly separable features to be used by an SVM (Support Vector Machine) which classifies every T-F bin as speech or noise dominant. To improve the generalization ability of the network when exposed to unseen conditions during training, this is fed with pitch-based features. Thus, the constraints of the linear SVMs classifiers are overcome by integrating a deep learning architecture. This hybrid strategy tries to exploit the best features of both paradigms to achieve an outstanding system performance. This is the philosophy followed regarding the contributions presented in this chapter as well, where two complex stages (from an analytical perspective) of a noise-robust ASR system (i.e. missing-data mask and noise estimation), are addressed by taking benefit from the powerful modeling capabilities of DNNs.

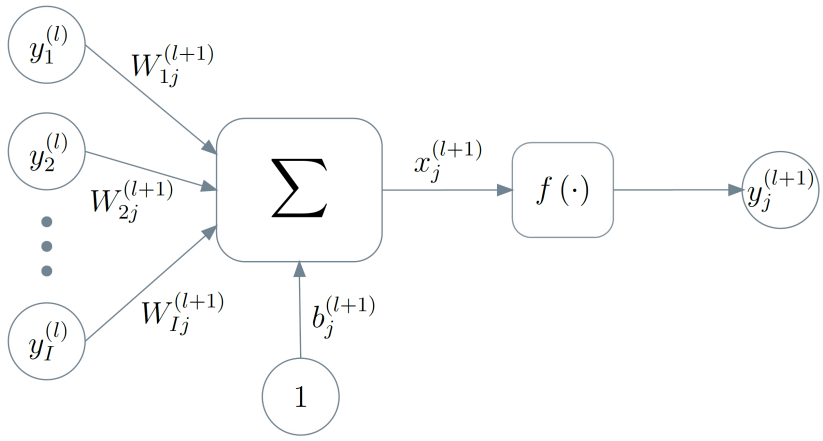

Figure 5.1: Structure of a neuron of an artificial neural network.

As it was reviewed throughout Section 2.4, multi-channel information can be exploited in synergy with deep learning for noise-robust ASR purposes. In this regard, let us recall here the winner of the 3rd CHiME Speech Separation and Recognition Challenge [203], who employed for acoustic modeling a deep learning architecture called NIN-CNN trained on the six available channels. NIN-CNN is a convolutional neural network (CNN) based on the concept "network-in-network" proposed in [116] to improve the performance of image classification applications. Additionally, a recurrent neural network (RNN) was used for language modeling as well as, on the front-end side, MVDR beamforming was applied for enhancing the multi-channel noisy signal. A different approach was that reported in [125], where the output of a filter-and-sum beamformer was enhanced by a DNN-based post-filtering to feed the recognition engine. As we discussed in Section 2.4, these combinations produce excellent results in terms of recognition accuracy.

The rest of this section is mainly devoted to describe the fundamentals of the deep feedforward neural networks, as this is the architecture considered by our dual-channel deep learning-based techniques. To conclude, a glimpse on both recurrent neural networks (RNNs) and convolutional neural networks (CNNs) is also provided.

5.1.1 Deep feedforward neural networks

A deep feedforward neural network is a multi-layer perceptron with multiple hidden layers. To understand what this means let us start by presenting the basic deep learning unit: the neuron. Its structure is drawn in Figure 5.1, and it simply consists of a parametric non-linear transformation of the weighted sum of its input $\mathbf{y}^{(l)} = \left(y_1^{(l)}, ..., y_I^{(l)}\right)^\top$

plus an offset, as,

$$y_j^{(l+1)} = f\left(x_j^{(l+1)}\right) = f\left(\sum_{i=1}^{I} y_i^{(l)} W_{ij}^{(l+1)} + b_j^{(l+1)}\right), \tag{5.1}$$

where $W_{ij}^{(l+1)}$ is a weight and $b_j^{(l+1)}$ is a bias coefficient. Furthermore, $f(\cdot)$ is the so-called activation function. A set of neurons sharing the same input is referred to as a layer, and multiple layers can be concatenated to shape an ANN as in Figure 5.2. In this example, we have an ANN with an input, a hidden and an output layer with three, four and two units, respectively. As aforementioned, that ANN would be deep, i.e. a DNN, if it had at least two or three hidden layers (depending on the criterion considered). On the basis of Eq. (5.1), if $l = 1$ corresponds to the input layer, $l + 1$ would refer to the first hidden layer. Thus, for $l = 1$, $I = 3$ in the example of Figure 5.2 and (5.1) would have to be computed for $j = 1, ..., 4$ in order to obtain the output vector of the hidden layer. Typical activation functions, plotted in Figure 5.3, are the sigmoid and rectifier functions, which are respectively defined as,

$$\begin{aligned} y_j^{(l+1)} &= \frac{1}{1 + e^{-x_j^{(l+1)}}}; \\ y_j^{(l+1)} &= \max\left(0, x_j^{(l+1)}\right). \end{aligned} \tag{5.2}$$

It should be noticed that neurons using the rectifier activation function are also known as ReLUs (Rectified Linear Units). The rectifier activation function is preferred over the sigmoid mainly due to the vanishing gradient problem [88]. An activation function of particular interest is the softmax function, which is used at the output layer of ANNs intended to multiclass classification. For an ANN with a total of L layers and an output layer with J neurons (i.e. classes), the softmax function can be expressed as,

$$y_j^{(L)} = \frac{e^{x_j^{(L)}}}{\sum_{j'=1}^{J} e^{x_{j'}^{(L)}}}, \tag{5.3}$$

where $y_j^{(L)}$ $(j = 1, ..., J)$ represents a categorical distribution. A popular example of use of the softmax activation function is in DNN-based acoustic modeling, where those are employed to estimate the posterior probabilities of the HMM states [82].

As mentioned above, one of the landmarks in deep learning was the integration of restricted Boltzmann machines (RBMs), introduced by Hinton *et al.* in 2006 [83], to overcome the weaknesses of discriminative training when trying to optimize deep structures. Instead of randomly initializing the parameters of the DNN, these are

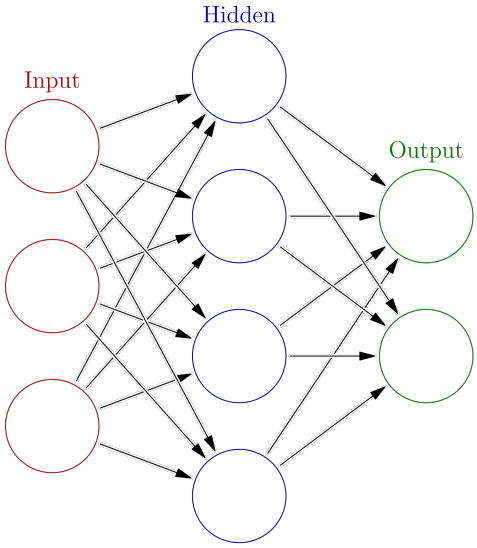

Figure 5.2: Example of a neural network with one hidden layer [1].

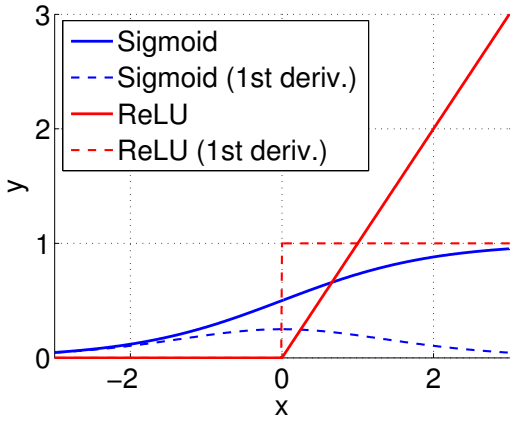

Figure 5.3: Comparison between the sigmoid and rectifier (ReLU) activation functions.

greedily set up by performing an unsupervised generative pre-training by considering each pair of layers as RBMs. This way, a much better starting point is achieved before supervised training by application of backpropagation learning to jointly optimize all the layers [84]. A similar alternative that can also be found in the literature is to pre-train the DNN by considering each pair of layers as denoising auto-encoders [17].

To discriminatively train the DNN, a cost function $\mathcal{C}$ must be defined first. This cost function measures the discrepancy between the output produced by the network from the input training data and the corresponding target data. The derivatives of $\mathcal{C}$ are

backpropagated through the network to optimize its parameters, namely weights and biases. To do this, a stochastic gradient descent (SGD) method is iteratively applied to random small sets of training examples called minibatches in order to minimize the cost function. Given a minibatch t, the parameters of the DNN are updated proportionally to the gradient by means of the rule

$$\Delta\left(W_{ij}^{(l)}(t+1), b_j^{(l)}(t+1)\right) = \omega\Delta\left(W_{ij}^{(l)}(t), b_j^{(l)}(t)\right) - \epsilon\frac{\partial\mathcal{C}}{\partial\left(W_{ij}^{(l)}(t), b_j^{(l)}(t)\right)} \tag{5.4}$$

at every layer $2 \leq l \leq L$, where ϵ is the learning rate and $\omega \in (0,\ 1)$ is the momentum coefficient. Momentum smooths the gradient calculated for minibatch t, thereby damping oscillations across ravines and speeding progress down them [82]. An epoch is completed every time the parameters of the network are updated from all the minibatches composing the whole training dataset. After a number of epochs, this discriminative training finishes in accordance with the chosen stopping criterion, e.g. the cost function stops decreasing. Typical cost functions are mean square error (MSE) and cross-entropy. While the former is especially suitable for regression purposes, the cross-entropy cost function is usually considered when designing a DNN using the softmax activation function at its output layer for classification. Thus, if $\mathbf{d} = (d_1, ..., d_J)^\top$ is a vector of target probabilities (normally binary) and the output of the network, $\left\{y_j^{(L)}; j = 1, ..., J\right\}$, is as in (5.3), the cross-entropy cost function can be expressed as

$$\mathcal{C} = -\sum_{j=1}^{J} d_j \log y_j^{(L)}. \tag{5.5}$$

Once the DNN is trained, it is ready to estimate new outputs by forward pass of new input data through the network.

As aforementioned, the rectifier activation function is preferred over the sigmoid one as a result of the vanishing gradient problem [87]. This problem occurs when adjusting the parameters of the DNN by backpropagation. Since backpropagation calculates gradients of the cost function according to the chain rule to update the values of the parameters in each iteration (see Eq. (5.4)), these gradients can exponentially decrease (i.e. vanish) at deep layers if the derivative of the activation function is typically near zero, as it is the case of the sigmoid function. This makes very difficult to update the values of the parameters at the deepest layers at each iteration, and, therefore, to properly train the network. While this problem could be overcome by selecting a trade-off value for the learning rate parameter, it might be more robust to use ReLUs, since the derivative of the rectifier activation function is one as long as the unit is active (see Figure (5.3)).

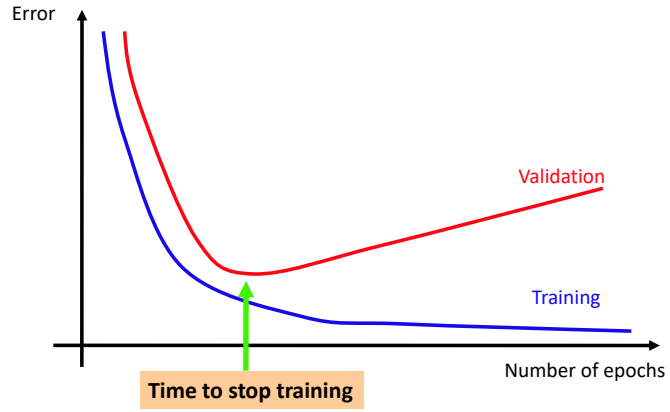

Figure 5.4: The early-stopping strategy to avoid overfitting during backpropagation learning [128].

Since the DNNs, with their large number of parameters, are able to learn very complex non-linear relationships between the input training and the output target data, they can easily be affected by overfitting to the training data. This is an important issue because the generalization ability of the DNNs to unseen examples during the training phase is severely reduced, resulting in a drop in performance of the network. While a solution to this fact may be to make use of very large training datasets [27], this is not always possible, in such a way that other strategies might be followed to avoid overfitting. Two non-exclusive popular strategies to do this are early-stopping and dropout [85]. Early-stopping simply consists of stopping backpropagation learning when the error on a validation dataset increases after successive training iterations, as exemplified in Figure 5.4. On the other hand, dropout works by randomly deactivating a percentage of neurons in each hidden layer, ρ, for each sample during the training phase [163, 202]. This is similar to adding random noise to the training data so that the DNN parameters are more robust to noise, namely the DNN improves its generalization ability. During the test phase, neither dropping out any neurons nor using a random combination of them is required, but only to properly scale the weights at every layer by $(1 - \rho)$ while employing all the neurons.

5.1.1.1 Unsupervised pre-training by RBMs

A diagram of a restricted Boltzmann machine (RBM), that can be seen as a two-layer neural network, is shown in Figure 5.5. RBMs are mainly used to initialize the set of parameters of a DNN to avoid falling into local minima during backpropagation learning. This could happen because of the complex error surface derived from the

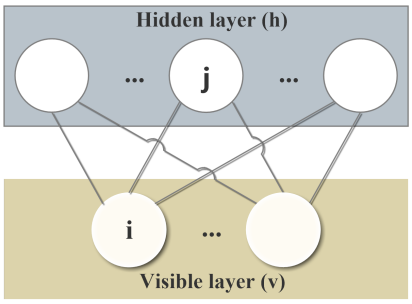

Figure 5.5: Example of a restricted Boltzmann machine.

large number of hidden layers [84]. An RBM consists of a visible layer with stochastic units, that represent input data, which are only connected to the stochastic units in the hidden layer. Hidden units are usually modeled by Bernoulli distributions. On the other hand, visible units can be modeled with either Bernoulli or Gaussian distributions. In the first case the resulting model is referred to as Bernoulli-Bernoulli RBM (BRBM), while the second as Gaussian-Bernoulli RBM (GRBM). GRBMs are very useful to model real-valued input data (e.g. input features), so that they are often used as the first level of a multi-layer generative model built with stacked RBMs, also known as deep belief network (DBN) [82].

Let $\mathbf{v}$, $\mathbf{h}$ and θ be the visible units, the hidden units and the set of parameters (namely weights and biases) of an RBM, respectively. The probability of a visible vector given the set of parameters is obtained by summing over all hidden vectors as

$$P(\mathbf{v}|\theta) = \frac{1}{Z}\sum_{\mathbf{h}} e^{-E(\mathbf{v},\mathbf{h}|\theta)}, \tag{5.6}$$

where $Z = \sum_{\mathbf{v}}\sum_{\mathbf{h}} e^{-E(\mathbf{v},\mathbf{h}|\theta)}$ is known as the partition function and $E(\mathbf{v},\mathbf{h}|\theta)$ is an energy function that defines the joint configuration of the visible and hidden units. For a BRBM, the energy function is

$$E_B(\mathbf{v},\mathbf{h}|\theta) = -\sum_{i=1}^{V}\sum_{j=1}^{H} w_{ij}v_i h_j - \sum_{i=1}^{V} a_i v_i - \sum_{j=1}^{H} b_j h_j, \tag{5.7}$$

and in the case of a GRBM,

$$E_G(\mathbf{v},\mathbf{h}|\theta) = -\sum_{i=1}^{V}\sum_{j=1}^{H} w_{ij}v_i h_j + \frac{1}{2}\sum_{i=1}^{V}(v_i - a_i)^2 - \sum_{j=1}^{H} b_j h_j, \tag{5.8}$$

where w_{ij} represents the symmetric weight between the visible, v_i, and hidden, h_j, units, and a_i and b_j their respective bias terms. The total number of visible and hidden units are V and H, respectively.

The set of parameters θ is estimated by maximizing $\log P(\mathbf{v}|\theta)$ from training data. This approach yields the following simple updating equation for the set of weights:

$$\Delta w_{ij} = \epsilon \cdot (\mathrm{E}_{data}[v_i h_j] - \mathrm{E}_{model}[v_i h_j]), \tag{5.9}$$

where ϵ is the learning rate and $\mathrm{E}[\cdot]$ indicates expectation under the corresponding distribution. To overcome the difficulties in getting samples of $\mathrm{E}_{model}[v_i h_j]$, Hinton proposed in [80] a fast algorithm called contrastive divergence (CD). Briefly, this algorithm performs alternating Gibbs sampling from visible units initialized to a training data vector [82]. In order to perform the CD algorithm, the following conditional probabilities are employed in the case of a BRBM:

$$P_B(h_j = 1|\mathbf{v}, \theta) = \sigma\left(\sum_{i=1}^{V} w_{ij} v_i + b_j\right) \tag{5.10}$$

and

$$P_B(v_i = 1|\mathbf{h}, \theta) = \sigma\left(\sum_{j=1}^{H} w_{ij} h_j + a_i\right), \tag{5.11}$$

where $\sigma(x) = 1/(1 + \exp(-x))$ is the sigmoid function. For the case of a GRBM, conditional probabilities can be calculated as

$$P_G(h_j = 1|\mathbf{v}, \theta) = \sigma\left(\sum_{i=1}^{V} w_{ij} v_i + b_j\right) \tag{5.12}$$

and

$$P_G(v_i = 1|\mathbf{h}, \theta) = \mathcal{N}\left(\sum_{j=1}^{H} w_{ij} h_j + a_i, 1\right), \tag{5.13}$$

where v_i is real-valued in this case and $\mathcal{N}(\cdot)$ denotes a normal distribution with mean $\sum_j w_{ij} h_j + a_i$ and unit variance. Before pre-training, input data should be normalized in such a way that each coefficient has zero mean and unit variance according to the assumptions of (5.13) and the energy function of Eq. (5.8). Such assumptions are required to be adopted due to the difficulties in learning the standard deviation of a GRBM as reported in [81].

5.1.2 Other deep architectures

Two types of deep learning architectures that are rapidly increasing their popularity over the last years among the community of speech researchers are recurrent neural networks (RNNs) and convolutional neural networks (CNNs). They are briefly introduced down below.

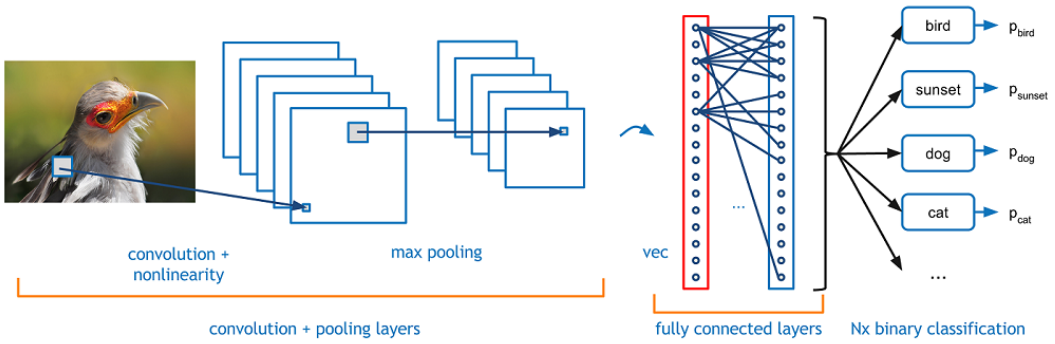

Figure 5.6: Example of a CNN used for image classification [38].

5.1.2.1 Recurrent neural networks

A recurrent neural network (RNN) is a type of ANN which, unlike the classical DNNs presented above, is able to model temporal dynamics. It can be considered that an RNN has memory through time as its current internal state depends on all previous internal states. This feature makes RNNs very appropriate for sequential data modeling. Indeed, it is shown that better recognition performance can be achieved by using RNNs instead of deep feedforward neural networks for ASR acoustic modeling [68]. This makes sense since RNNs, that are also "deep in time", are able to exploit the temporal correlations present in the speech signal. Because of this distinguishing characteristic, it is also possible to directly substitute a DNN-HMM-based acoustic modeling architecture by an RNN just trained end-to-end for speech recognition [67]. While this approach still has little impact in the literature, it is better able to exploit the temporal dynamics than HMMs, avoiding at the same time the problem of possibly using incorrect alignments as target during the training phase [68]. Similarly to the backpropagation algorithm to train classical DNNs, RNNs can be trained in a supervised manner by using a variant of that algorithm called backpropagation through time (BPTT) [198] in order to estimate the parameters of the recurrent network that minimize the cost function. Nowadays, the most popular type of RNN is the so-called long short-term memory (LSTM) recurrent network [89] which solves the vanishing gradient problem as one of its main features.

5.1.2.2 Convolutional neural networks

A convolutional neural network (CNN) is another type of deep learning architecture consisting of the stack of a convolutional layer followed by a subsampling step, often known as pooling layer, and a series of fully-connected layers as in a typical DNN. This

architecture is exemplified in Figure 5.6, where a CNN is employed for image classification. CNNs are very useful to take advantage of the two-dimensional structure of the input data, e.g. images, since they are locally connected networks. More precisely, connections are restricted in the convolutional layer in the sense that each hidden neuron is just connected to a few input units. This makes the processing of large inputs becomes feasible in relation with using DNNs. The convolutional layer performs the convolution operation on feature maps through the use of filters or kernels [154], the size of which determines the locally connected structure. A subsequent pooling layer implementing the mean or max activation function is employed to reduce the dimension of the features coming from the convolutional layer. The concatenation of the convolutional and pooling layers provides us with translation invariant features [141]. These pooled convolved features are then normally used for classification. CNNs can be similarly trained in a supervised manner using the backpropagation algorithm. It must be remarked that a CNN with the same amount of hidden neurons as a DNN has a smaller number of parameters to be adjusted, which can be a benefit in terms of computational complexity. CNNs have also been successfully applied to a number of tasks related with speech processing, such as acoustic modeling [154], speaker identification and speaker gender and phone classification [206].

5.2 DNN-based missing-data mask estimation

In this section we propose taking advantage of the learning capabilities of DNNs to efficiently estimate missing-data masks from dual-channel noisy speech. This rather simple and straightforward approach will supply quite accurate missing-data masks for the primary channel by exploiting the PLD between the two available channels in accordance with the dual-microphone set-up considered throughout this Thesis. Under this scenario, it is clear that a missing-data mask can be easily derived from a comparison between the noisy speech power present in both channels, where the secondary one is a good noise reference since speech is much attenuated. While missing-data masks can be employed in several ways for noise-robust ASR purposes (e.g. for marginalization [31]), our method will be oriented to a spectral reconstruction technique, in particular, the truncated-Gaussian based imputation (TGI) already introduced in Subsection 2.2.4. Unlike other related works such as [139, 193], where a wide set of features (i.e. amplitude modulation spectrogram, relative spectral transform and perceptual linear prediction, pitch-based features, etc.) is extracted to feed the DNN, in our system the DNN directly provides an estimation of the missing-data mask by just using dual-channel log-Mel spectral features. Through this approach we can obtain a competitive performance with less computation.

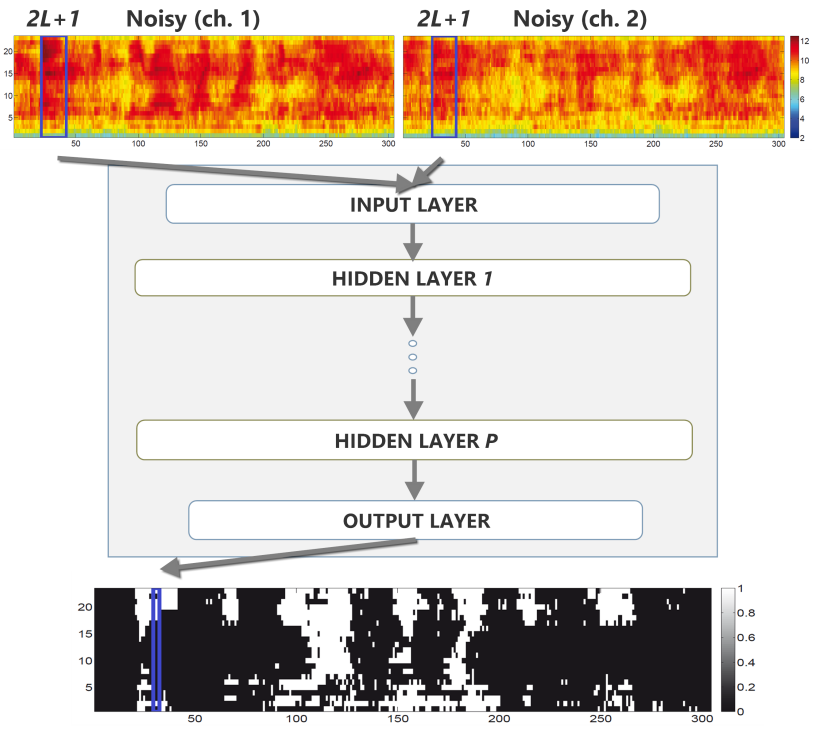

Figure 5.7: An outline of the DNN as used for missing-data mask estimation purposes.

Particularly, a feedforward neural network with two hidden layers is employed as in [139, 193]. An outline of the DNN as used for missing-data mask estimation purposes is shown in Figure 5.7. Missing-data mask estimation is performed in the log-Mel domain, where many of the spectral reconstruction algorithms operate (as TGI does). The proposed DNN works on a frame-by-frame basis, i.e. the DNN returns a missing-data mask for each frame in the utterance. Let the dual-channel noisy speech log-Mel features at time frame t be

$$\mathbf{y}(t) = \begin{pmatrix} \mathbf{y}_1(t) \\ \mathbf{y}_2(t) \end{pmatrix}, \tag{5.14}$$

where $\mathbf{y}_i(t) = (y_i(0,t), y_i(1,t), ..., y_i(\mathcal{M}-1,t))^\top$, $i = 1, 2$, is the noisy speech log-Mel feature vector obtained from the signal acquired by the i-th microphone of the device (channel i). Then, the input for the DNN at time t is the stacked vector

$$\boldsymbol{\mathcal{Y}}(t) = \begin{pmatrix} \mathbf{y}(t-L) \\ \vdots \\ \mathbf{y}(t+L) \end{pmatrix}, \tag{5.15}$$

where $L \geq 0$ determines the size of the temporal window around frame t, that is $2L+1$.

Thus, the dimensionality of the input vector is

$$\dim\left(\boldsymbol{\mathcal{Y}}(t)\right) = 2\mathcal{M}(2L+1), \tag{5.16}$$

where, as before, $\mathcal{M}$ is the number of filterbank channels. On the other hand, the target is an oracle missing-data mask vector corresponding to feature vector $\mathbf{y}_1(t)$. In this case, the size of each output vector is $\mathcal{M} \times 1$. It must be noticed that oracle masks are obtained by direct comparison between the clean and noisy utterances using a threshold of η_O dB signal-to-noise ratio (SNR).

As aforementioned, the DNN training consists of an unsupervised generative pre-training, where it is considered each pair of layers as RBMs, followed by a supervised fine-tuning step. In particular, the input and first hidden layers form a GRBM (i.e. a visible layer of Gaussian variables connected to binary units in a hidden layer) since the input vector is real-valued. The successive pairs of layers form BRBMs (i.e. two layers with connections between their binary units). Input data, i.e. $\boldsymbol{\mathcal{Y}}(t)$, are used to train the GRBM and the inferred states of its hidden units are employed to train the following BRBM, and so on. Since the units in the visible layer of a GRBM are assumed to be standard-normally distributed [81], input data are properly normalized, per filterbank channel, to zero mean and unit variance. Indeed, to maintain the power ratio between the two channels, both of them are jointly normalized. The parameters resulting from this generative model consisting of the stack of RBMs (also known as deep belief net) are used to initialize the DNN, which is then fine-tuned by performing a supervised training by means of the backpropagation algorithm. The cross-entropy criterion was chosen for backpropagation learning. Finally, it must be specified that all the hidden and output layers employ sigmoid units. Therefore, the output of the DNN is rounded to get the final mask values, i.e. 0's and 1's.

An example of the TGI spectral reconstruction of a dual-channel noisy utterance by using this dual-channel DNN-based system is shown in Figure 5.8. The considered utterance, contaminated with bus noise, was recorded by means of a dual-microphone smartphone employed in close-talk conditions. As we can see, the DNN is able to faithfully distinguish between T-F bins where speech dominates and those where noise prevails. In this way, the resulting spectral reconstruction is clearly more similar to the clean spectrum than the noisy spectrum at the primary channel.

In Subsection 6.2.4, the rest of practical details on the DNN setup as well as the parameter values will be described.

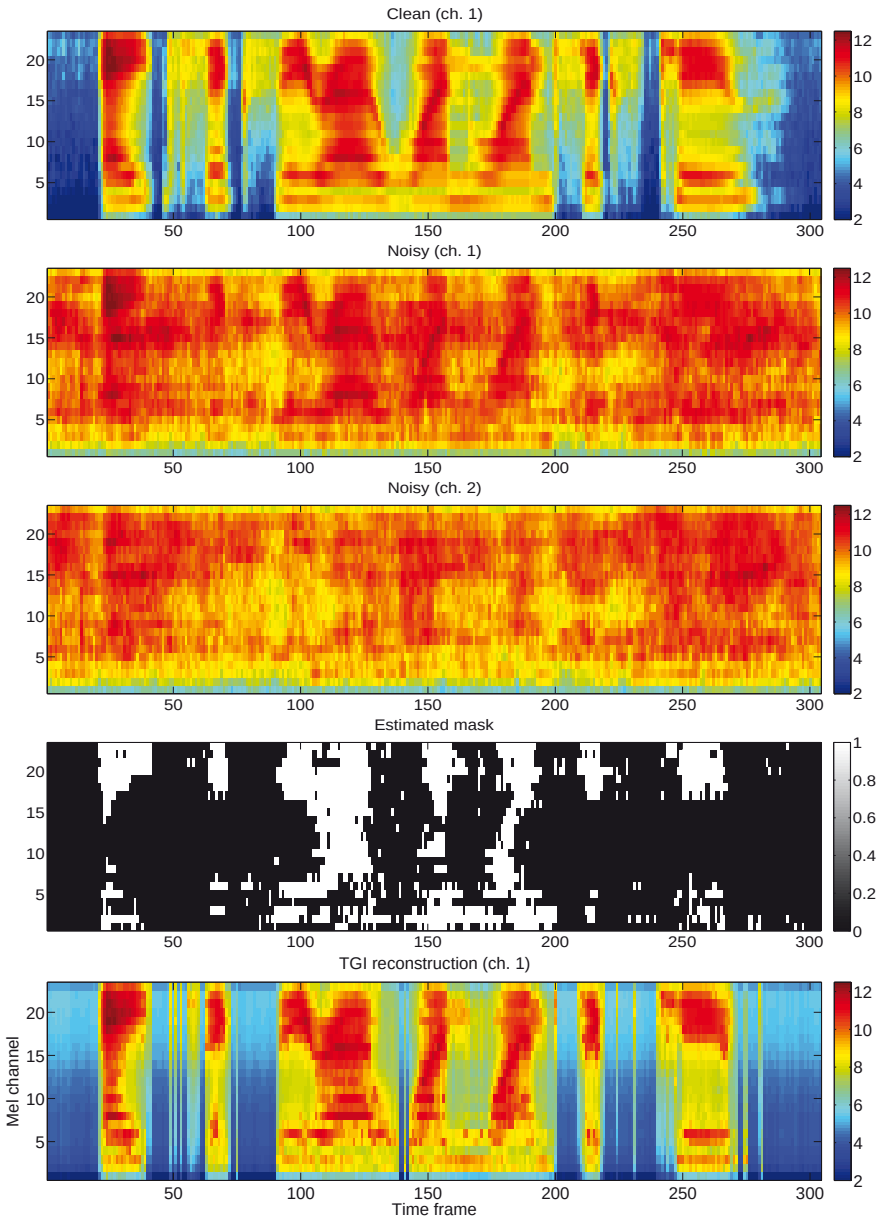

Figure 5.8: Example of the TGI reconstruction of an utterance recorded with a dual-microphone smartphone in close-talk position. All the spectrograms are in the log-Mel domain. From top to bottom: clean utterance (1st ch.), corrupted by bus noise at 0 dB (1st & 2nd chs.), mask estimated by the dual-channel DNN-based system and the resulting reconstruction (over the 1st ch.).

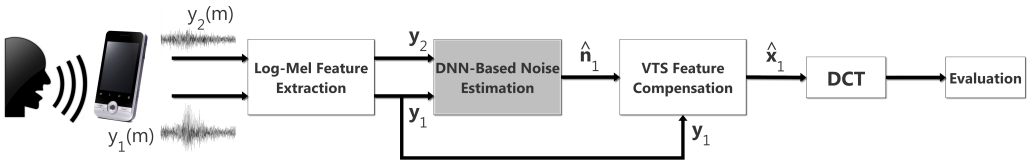

Figure 5.9: Block diagram of the noise-robust ASR framework considered to test the performance of the dual-channel DNN-based noise estimation method.

5.3 DNN-based noise estimation

A parallel approach to that developed in the above section is considered to efficiently estimate noise from dual-channel noisy speech. Now, a DNN is used to find a mapping function between the dual-channel noisy observation and the noise that contaminates speech at the primary channel. While DNNs have been employed for many different tasks from noise-robust ASR such as missing-data mask estimation [120, 139, 193], surprisingly they had not yet been applied to directly estimate noise. Similarly to the missing-data mask case, noise estimates can be employed in various ways for noise-robust ASR purposes. While the quality of these estimates will be evaluated in Subsection 6.2.4 when combined with VTS feature compensation, in this section we will focus on obtaining the estimates themselves. In this respect, the noise-robust ASR framework considered to test the performance of this dual-channel DNN-based noise estimation method is depicted in Figure 5.9. A dual-microphone smartphone is considered in this example. The noisy speech signal captured by the primary microphone of the smartphone is denoted as $y_1(m)$. Similarly, $y_2(m)$ refers to the noisy speech signal recorded by the secondary microphone of the device. As we can expect for our dual-channel set-up, the noise components in $y_1(m)$ and $y_2(m)$ are assumed to be quite similar while speech is much attenuated at the secondary sensor with respect to the primary one since the former is placed in an acoustic shadow regarding the speaker's mouth. Then, log-Mel spectral features $\mathbf{y}_i$ are extracted from the noisy signals $y_i(m)$, $i = 1, 2$, which are employed by a DNN-based stage in order to provide a noise estimate of the primary channel, $\hat{\mathbf{n}}_1$. To obtain the clean speech log-Mel features at the primary channel, $\hat{\mathbf{x}}_1$, this noise estimate is used along with $\mathbf{y}_1$ by a VTS feature compensation method. Finally, $\hat{\mathbf{x}}_1$ is transformed into the cepstral domain by application of the discrete cosine transform (DCT) prior to be used by the speech recognizer.

As in Section 5.2, a DNN is considered in the following to find a non-linear mapping function between dual-channel noisy speech and noise log-Mel features at the primary channel of the mobile device. This DNN-based method exploits the PLD between the two microphones of the device to effectively provide accurate noise estimates. An

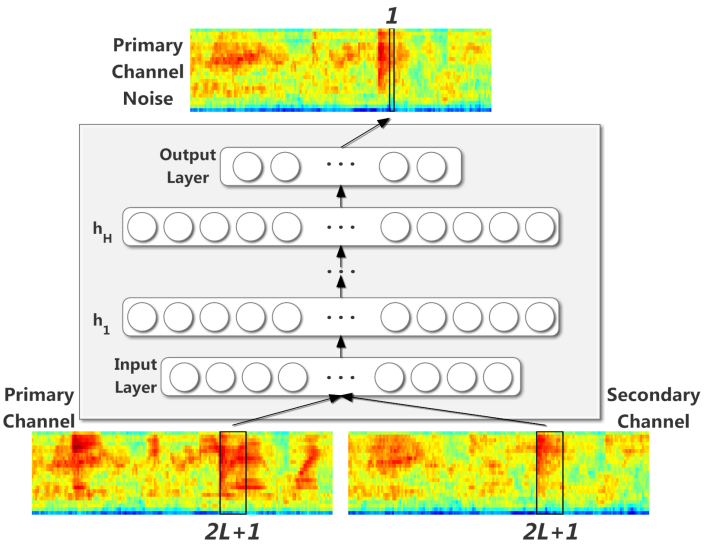

Figure 5.10: An outline of the DNN as used for noise estimation purposes.

illustration on how the DNN is used to this end can be seen in Figure 5.10. Along with $\mathbf{y}_i(t)$, $i = 1, 2$, let

$$\mathbf{n}_1(t) = (n_1(0,t), n_1(1,t), ..., n_1(\mathcal{M}-1,t))^\top \tag{5.17}$$

be a noise log-Mel feature vector at time frame t coming from the primary channel. Our DNN works on a frame-by-frame basis so that it gives a noise frame estimate at each time t from the same input features as used for DNN-based missing-data mask estimation (see Eq. (5.15)). Additionally, as expected, the corresponding $\mathcal{M}$-dimensional target vector is that of Eq. (5.17).

The DNN training is performed in the same way as for the missing-data mask estimation of Section 5.2. In this case, the MSE criterion was chosen for backpropagation learning. Furthermore, the activation function type considered for the hidden layers is sigmoid while that it is chosen linear for the output layer, as could be expected for regression purposes. It should be noticed that all input and target data are normalized to zero mean and unit variance. Hence, the output of the DNN is properly denormalized when used to perform the regression during the test phase. As for the missing-data mask case, both channels are jointly normalized per frequency bin to keep the power ratio between the two channels unchanged.

A comparative example between this dual-channel DNN-based noise estimation approach and linear interpolation noise estimation, which demonstrates a very competitive performance [122, 159], is shown in Figure 5.11. The latter noise estimation approach, applied on the primary channel, consists of the linear interpolation between

the averages of the first and last $M = 20$ frames of the log-Mel utterance. The first and last M frames are directly taken as part of the final noise estimate and this is bounded above by the noisy spectrum for more consistency. From top to bottom the figure shows the primary log-Mel spectrum of a noisy utterance captured with a dual-microphone smartphone in close-talk conditions, the actual bus noise that contaminates it at 0 dB, the corresponding dual-channel DNN-based noise estimation and the noise estimated by linear interpolation. As we can observe, the noise spectrum estimated by our method better resembles the actual one than that from linear interpolation.

The values chosen for the DNN hyperparameters and the rest of details about the DNN setup can be found in Subsection 6.2.4.

5.3.1 Noise-aware training

DNN noise-aware training (NAT) is a method first appeared in [163] to strengthen the DNN-based acoustic modeling for ASR. It basically consists of appending a noise estimate to the network's input vector containing the noisy speech features in order to improve word recognition rates when employing multi-style acoustic modeling. Since then, NAT has been successfully applied to different tasks such as, for instance, DNN-based speech enhancement [202]. We want to explore if the DNN-based noise estimation approach presented above can be improved by increasing the awareness of the DNN about the noise that contaminates speech in each case.

As mentioned above, a simple noise estimator, which has demonstrated to be quite accurate [159], consists of the linear interpolation between the averages of the first and last M frames of an utterance in the log-Mel domain. Inspired by this method, and assuming that an utterance is T frames long, we propose an alternative NAT scheme in which the initial input vector $\boldsymbol{\mathcal{Y}}(t)$ is augmented by appending the aforementioned averages,

$$\bar{\mathbf{n}}_1^{(0)} = \frac{1}{M}\sum_{t=0}^{M-1}\mathbf{y}_1^{\top}(t); \qquad \bar{\mathbf{n}}_1^{(1)} = \frac{1}{M}\sum_{t=T-M}^{T-1}\mathbf{y}_1^{\top}(t), \tag{5.18}$$

as well as a time index to indicate the frame's relative position within the utterance:

$$\tau(t) = \frac{t}{(T-1)}. \tag{5.19}$$

Additionally, we also include noise variance information by computing and appending the following sample quantities:

$$\boldsymbol{\sigma}_1^{(0)} = \frac{1}{M-1}\sum_{t=0}^{M-1}\left(\mathbf{y}_1^{\top}(t) - \bar{\mathbf{n}}_1^{(0)}\right)^2; \qquad \boldsymbol{\sigma}_1^{(1)} = \frac{1}{M-1}\sum_{t=T-M}^{T-1}\left(\mathbf{y}_1^{\top}(t) - \bar{\mathbf{n}}_1^{(1)}\right)^2, \tag{5.20}$$

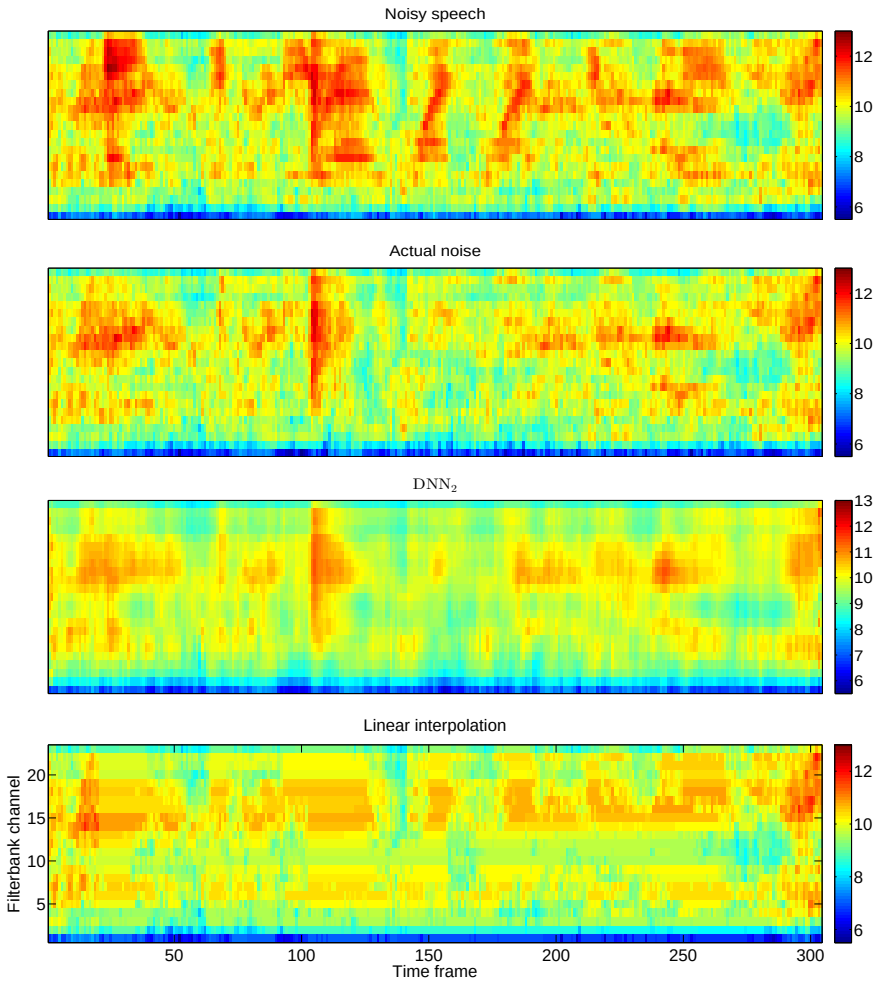

Figure 5.11: A comparative noise estimation example generated from an utterance captured with a dual-microphone smartphone used in close-talk position. From top to bottom: primary channel log-Mel noisy spectrum, actual bus noise that contaminates it at 0 dB, dual-channel DNN-based noise estimation and noise estimated by linear interpolation.

where $(\cdot)^2$ is applied element-wise. Thus, the final DNN input vector is

$$\boldsymbol{\mathcal{Y}}_{NAT}(t) = \left(\boldsymbol{\mathcal{Y}}^{\top}(t),\ \bar{\mathbf{n}}_1^{(0)},\ \bar{\mathbf{n}}_1^{(1)},\ \boldsymbol{\sigma}_1^{(0)},\ \boldsymbol{\sigma}_1^{(1)},\ \tau(t)\right)^{\top}, \tag{5.21}$$

with dimension

$$\dim(\boldsymbol{\mathcal{Y}}_{NAT}(t)) = \dim(\boldsymbol{\mathcal{Y}}(t)) + 4\mathcal{M} + 1 = 4\mathcal{M}\left(L + \frac{3}{2}\right) + 1. \tag{5.22}$$

5.4 Summary

In this chapter we have explored the use of deep learning applied to noise-robust ASR on mobile devices with several sensors. In the first instance, we have tried to give a definition of deep learning while mentioning its advantages when applied to the resolution of problems that have traditionally been addressed from the classical analytical signal processing paradigm. Among these advantages, we have highlighted the powerful modeling capabilities of the deep learning architectures without the need for approximations or assumptions on the underlying problem. Then, a brief review of the related literature was made. More precisely, we focused on those deep learning-based approaches devised for both ASR and speech enhancement purposes as well as emphasis was placed on the hybrid DNN/signal processing architectures. Such architectures, instead of proposing an end-to-end DNN-based solution, try to exploit the best features of both the deep learning and signal processing paradigms to achieve outstanding system performances. This philosophy, which is expected to be successfully explored in the near future, is the one followed by our deep learning-based proposals. In particular, such contributions make use of deep feedforward neural networks (DNNs), so the theoretical fundamentals of this architecture was explained. Since backpropagation can get stuck into local minima during the supervised training, a proper initialization of the DNN parameters is required to avoid this issue. It was argued that such an initialization can be carried out by performing an unsupervised generative pre-training of the DNN from considering each pair of layers as restricted Boltzmann machines (RBMs), so these were also outlined. To complete our theoretical review, two types of deep learning architectures that are rapidly increasing their popularity over the last years among the community of speech researchers were briefly introduced: the recurrent neural networks (RNNs) and the convolutional neural networks (CNNs).

To conclude the chapter, two dual-channel deep learning-based contributions were presented to deal with the development of two complex (from an analytical point of view) tasks of a noise-robust ASR system. These tasks are missing-data mask and noise estimation, which are tackled by taking benefit from the powerful modeling capabilities of DNNs. More specifically, these DNNs exploit the power level difference

(PLD) between the two available channels to efficiently obtain the corresponding estimates with good generalization ability. While missing-data mask and noise estimates can be employed in various ways for noise-robust ASR purposes, in this Thesis they will be applied to spectral reconstruction by imputation and feature compensation, respectively. Additionally, a noise-aware training (NAT) strategy was also developed to explore if this dual-channel DNN-based noise estimation method can be improved by increasing its awareness about the noise that contaminates speech in each case.

CHAPTER 6

Experimental Evaluation

This chapter addresses the experimental evaluation of the different proposed noise-robust methods to be performed on intelligent mobile devices (IMDs) with several sensors. Such an evaluation is mainly carried out in terms of word recognition accuracy and/or word error rate of the automatic speech recognition (ASR) system integrating our contributions when employed in noisy environments. Related techniques are also tested and properly compared and analyzed along with ours from the obtained experimental results. The chapter comprises three differentiated sections. Section 6.1 is devoted to describe the experimental framework considered for evaluation. The experimental results are shown and discussed in Section 6.2. Finally, a summary is presented in Section 6.3.

6.1 Experimental framework

In this section we depict the experimental framework considered for evaluation, including the description of the different noisy corpora employed in Subsection 6.1.1 along with their related set-up particularities, namely the feature extraction process (Subsection 6.1.2) and the back-end configuration (Subsection 6.1.3).

6.1.1 Databases

The AURORA2-2C-CT/FT and the CHiME-3 corpora are the noisy speech databases considered during our experimental evaluation. All of them are multi-channel corpora reflecting the acquisition of speech with different types of IMDs employed in a variety of realistic noisy locations where the use of mobile devices is quite probable. In

	AURORA2-2C-CT/FT	CHiME-3
Device	Smartphone	Tablet
# of mics	2	6
Secondary mic?	Yes	Yes
Task	Connected digits (small-vocabulary)	WSJ0 5k (medium-vocabulary)
Type of data	Synthetic noisy speech (real noise)	Synthetic and real noisy speech
Type of distortion	Additive noise	Additive and convolutive noise
# of environments	8	4
# of test speakers	104	12

Table 6.1: Comparative overview between the AURORA2-2C-CT/FT and CHiME-3 corpora.

particular, while the CHiME-3 corpus is the result of the third edition of the CHiME Speech Separation and Recognition Challenge series [15], the AURORA2-2C-CT/FT corpora have been developed in our research group and must be highlighted as another contribution of this Thesis. A comparative overview between these databases is shown in Table 6.1. They are described with more detail in the following.

6.1.1.1 AURORA2-2C-CT/FT

The AURORA2-2C-CT (Aurora-2 - 2 Channels - Close-Talk) and the AURORA2-2C-FT (Aurora-2 - 2 Channels - Far-Talk) databases are synthetic dual-channel noisy speech databases generated from the well-known Aurora-2 corpus [148]. Aurora-2 is a well-established and widely used standard for research and development in noise-robust speech recognition. This framework, which the speech scientific community has employed for years to evaluate and compare its noise-robust developments, serves as the basis for the creation of the AURORA2-2C-CT/FT corpora. Aurora-2 was released by the working group STQ-AURORA, belonging to the European Telecommunications Standards Institute (ETSI), in order to evaluate the DSR (Distributed Speech Recognition) standards [2]. This is a synthetic single-channel noisy speech database of connected digits (small-vocabulary task) spoken by American English talkers. A total of 8 different noisy environments are considered in such a way that noise signals from them are artificially added to the clean speech signals coming from the TIDigits corpus [105]. Before summation, all the noise and clean speech signals are filtered to simulate the average response of a telecommunication terminal.

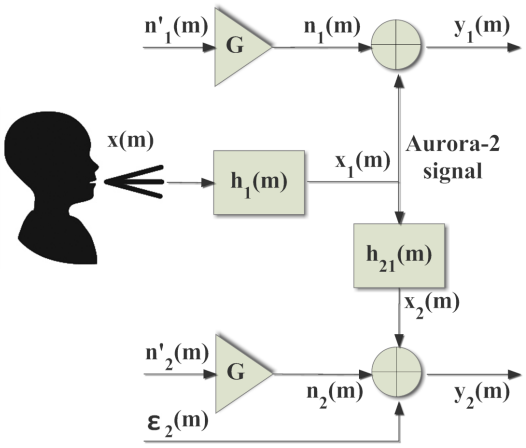

Figure 6.1: Generation block diagram of the AURORA2-2C-CT/FT databases, where $x_1(m)$ represents to the clean speech signals provided by the Aurora-2 database.

On the one hand, the AURORA2-2C-CT database tries to emulate the acquisition of dual-channel noisy speech data by using a smartphone with a dual-microphone in close-talk conditions (i.e. when the loudspeaker of the smartphone is placed at the ear of the user). On the other hand, the AURORA2-2C-FT database is generated in a similar way but emulating a far-talk scenario (i.e. when the user holds the device in one hand at a certain distance from her/his face). Both corpora have been defined in a similar manner, so that the rest of the description here presented is common to both of them and their differences explicitly remarked.

The generation scheme is outlined in Figure 6.1. The clean speech produced by the speaker, $x(m)$, is received at the primary microphone of the device transformed by the channel $h_1(m)$ as $x_1(m) = h_1(m) * x(m)$. Similarly, $x(m)$ is captured by the secondary sensor once transformed by the corresponding acoustic path: $x_2(m) = h_{21}(m) * x_1(m) = (h_{21}(m) * h_1(m)) * x(m)$ where $h_2(m) = h_{21}(m) * h_1(m)$. A noise gain factor G scales a recorded stereo noise signal, $\{n'_i(m);\ i = 1, 2\}$, in order to obtain a certain signal-to-noise ratio (SNR) at the primary channel, $y_1(m)$. Moreover, a small component of noise, $\varepsilon_2(m)$, is added to the secondary channel to account for the baseline noise, due to hiss noise from the circuitry and other factors. This is necessary since the acoustic path $h_{21}(m)$ may excessively attenuate the original baseline noise present in $x_1(m)$. The noise component $\varepsilon_2(m)$ is properly generated from the statistical distribution of the real baseline noise present in the primary channel.

It is assumed that the clean speech signals provided by Aurora-2 are the ones captured by the primary microphone of the device, $x_1(m)$, as indicated in Figure 6.1. Then, the clean speech relative acoustic path between the two sensors of the device,

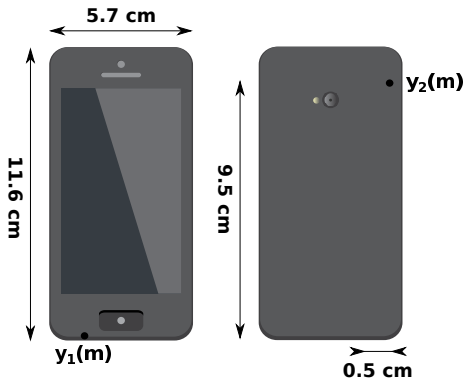

Figure 6.2: Characteristics of the device used for the generation of the AURORA2-2C-CT/FT databases.

$h_{21}(m)$, is estimated from stereo clean speech recorded in an anechoic chamber according to the following discussion. Let us assume that the environment acoustics corresponding to the acquisition of the Aurora-2 signals can be fully described by a set of acoustic parameters $\mathcal{A}$, so we will write $x_1(m;\mathcal{A}) = h_1(m;\mathcal{A}) * x(m)$, where $x(m)$ is the original speech signal as uttered by the speaker. In order to generate a speech signal for the secondary sensor acoustically coherent with that of the primary one, the former one should be acquired in the same acoustic conditions $\mathcal{A}$, that is, $x_2(m;\mathcal{A}) = h_2(m;\mathcal{A}) * x(m) = h_{21}(m;\mathcal{A}) * x_1(m;\mathcal{A})$. That is, filter $h_{21}(m)$ should reflect the same acoustic conditions $\mathcal{A}$ in which Aurora-2 was recorded. According to reference [105], which describes the recording conditions of the TIDigits corpus, this can be approximately accomplished by estimating $h_{21}(m)$ in an anechoic chamber.

For the AURORA2-2C-CT, it is assumed that the speaker holds the device virtually fixed with respect to herself/himself (although they can move as a whole). In this way, the acoustic path $h_{21}(m)$ was modeled as a time-invariant finite impulse response (FIR) filter with 1000 coefficients. Such a filter, $\hat{h}_{21}(m)$, was obtained from stereo clean speech $\{x_1^{(tr)}(m), x_2^{(tr)}(m)\}$ recorded in an anechoic chamber with a dual-microphone smartphone in close-talk condition (see Figure 6.2) by means of the minimization of the mean square error (MSE) between $x_2^{(tr)}(m)$ and $\hat{h}_{21}(m) * x_1^{(tr)}(m)$.

On the other hand, for the AURORA2-2C-FT it is assumed that the speaker can hold the device at different positions but fixed with respect to herself/himself during an utterance. This simplification is reasonable since, for short periods of use, it can be expected that the speaker does not modify substantially the position of the smartphone with respect to herself/himself. In this case, a total of $\mathcal{Q} = 35$ time-invariant FIR filters, $\{\hat{h}_{21}^{(q)}(m);\ q = 1, ..., \mathcal{Q}\}$, were obtained also from stereo clean speech recorded in an anechoic chamber with the same smartphone but in $\mathcal{Q}$ different far-talk positions.

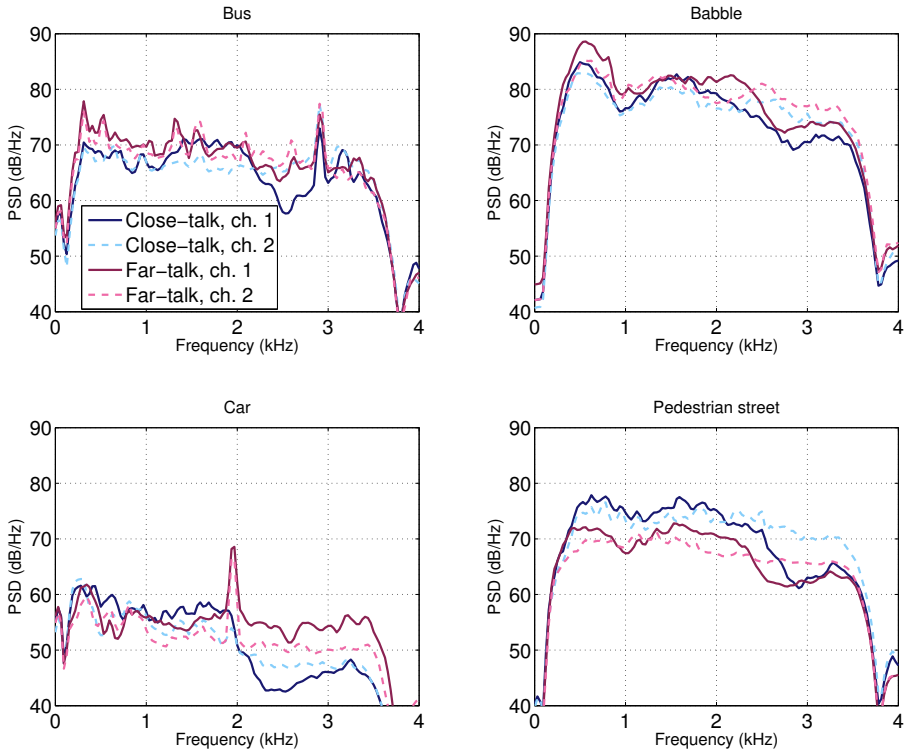

Figure 6.3: Power spectral densities of the noises used for the generation of the test set A of the AURORA2-2C-CT/FT databases. Noise PSDs for both the primary and secondary channels are plotted.

Every channel response $\hat{h}_{21}^{(q)}(m)$ was estimated in a similar fashion to the AURORA2-2C-CT database. Thus, given an utterance $x_1(m)$, its form in the secondary channel is obtained as $x_2(m) = \hat{h}_{21}^{(q)}(m) * x_1(m)$, where q is randomly chosen following a discrete uniform distribution $\mathcal{U}(1, \mathcal{Q})$.

Actual stereo noise signals, $\{n_1'(m), n_2'(m)\}$, were recorded (using the same dual-microphone smartphone) at different noisy places in close-talk position for the AURORA2-2C-CT database and far-talk conditions for the AURORA2-2C-FT corpus. The recorded noise signals correspond to bus, babble, car, pedestrian street, café, street and bus and train stations. Their respective power spectral densities (PSDs) are depicted in Figures 6.3 and 6.4 for both close- and far-talk conditions. As can be seen from these figures, in general, noise PSDs are similar at both the primary and secondary channels given a particular type of noise. This similarity is greater in far- than in close-talk conditions due to the acoustic shadow produced by the head of the speaker in the latter case. Finally, the noise gain factor G was computed by employing the application FaNT (Filtering and Noise Adding Tool) [86].

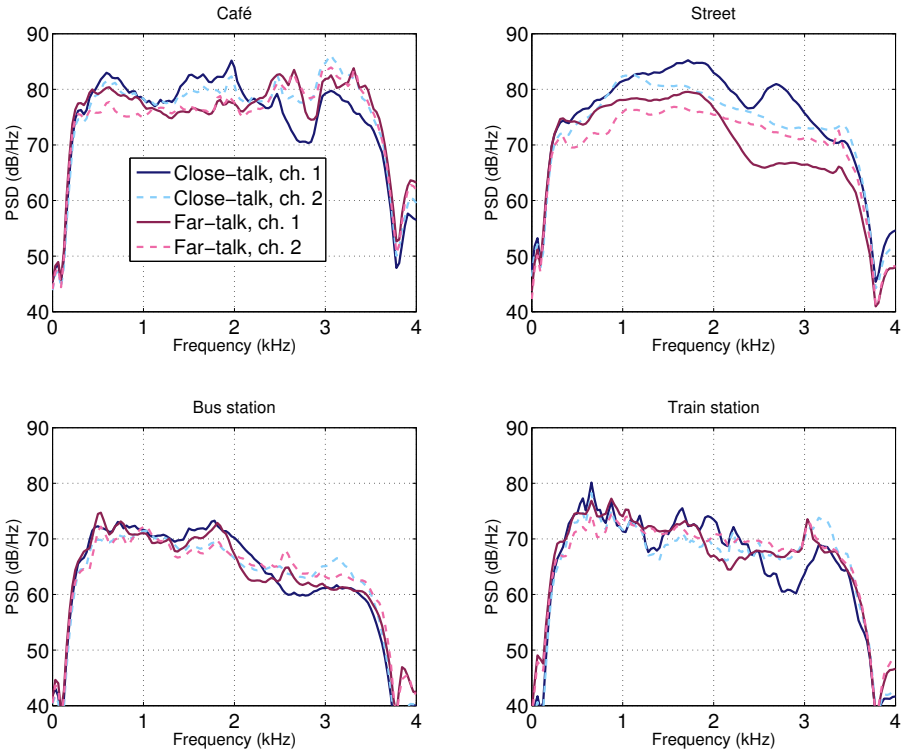

Figure 6.4: Power spectral densities of the noises used for the generation of the test set B of the AURORA2-2C-CT/FT databases. Noise PSDs for both the primary and secondary channels are plotted.

Two new test sets (A and B) were created for each database using the recorded noise signals. Following the Aurora-2 structure, test set A is comprised of the noises bus, babble, car and pedestrian street while test set B is comprised of the noises café, street, bus station and train station. Signals of each kind of noise were used to contaminate each Aurora-2 test subset at the SNRs (referred to the primary channel) -5 dB, 0 dB, 5 dB, 10 dB, 15 dB and 20 dB. In addition, the clean case is included as a seventh condition. Also, to enforce that speech and noise were recorded with a similar equipment, both of them are filtered with the G.712 characteristic as Aurora-2 does [148]. Each test set contains 28028 utterances, namely 1001 utterances per subset $\times$ 4 subsets $\times$ 7 SNRs, and the only difference between them is the type of noises in each set.

For clean acoustic model training the clean training dataset of Aurora-2 comprising 8440 utterances is used. Finally, the multi-condition training datasets of AURORA2-2C-CT/FT are also composed of 8440 utterances and created from the clean training dataset of Aurora-2. Similarly to [148], these multi-condition training datasets consist

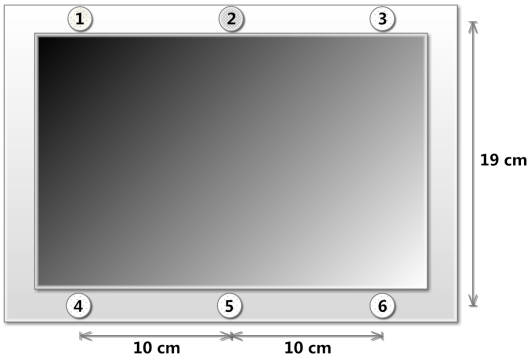

Figure 6.5: Characteristics of the device used for the generation of the CHiME-3 database. Microphone number 2 faces backwards while the rest of them face forward.

of dual-channel utterances contaminated with the types of noise in test set *A* at the SNRs (referred again to the primary channel) of 5 dB, 10 dB, 15 dB and 20 dB as well as the clean condition. As for the test sets, each combination of type of noise and SNR value defines a multi-condition training subset comprising 422 dual-channel utterances.

6.1.1.2 CHiME-3

CHiME-3 [15] is a novel framework (part of the well-known CHiME challenge series) specially intended for researching on multi-channel noise-robust speech recognition that includes ASR baseline software which uses the Kaldi ASR toolkit [152]. CHiME-3 database is comprised of both simulated and real noisy speech data. Real data were recorded in noisy environments by 12 US English talkers using a tablet with six microphones, the geometry of which is represented in Figure 6.5. Five of these microphones face forward and one faces backwards (that numbered as 2 in Figure 6.5). Simultaneously along with the tablet, a close-talk microphone was used to capture speech. The usual distance between every speaker and the tablet was around 40 cm (far-talk position). Similarly, simulated data were created by mixing clean speech utterances with background noise recordings. In particular, the speech data in this case correspond to utterances from the well-known speaker-independent medium (5k) vocabulary subset of the Wall Street Journal (WSJ0) corpus [147].

Training data are composed by 8738 noisy utterances (1600 real plus 7138 simulated from the standard WSJ0 training dataset) in the four different noisy environments considered: public transport (BUS), café (CAF), pedestrian area (PED) and street junction (STR). Furthermore, development and evaluation datasets are also defined separately for the simulated and real cases. Thus, each development dataset contains

	Training	Development	Evaluation
Simulated	7138	1640	1320
Real	1600	1640	1320

Table 6.2: Number of utterances per dataset in CHiME-3.

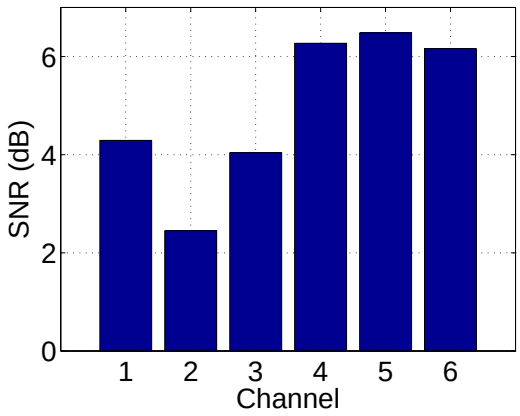

Figure 6.6: Average SNR as a function of the CHiME-3 channel estimated from the real development dataset.

1640 utterances (410 from each noisy environment) while each evaluation dataset is comprised of 1320 utterances (330 per noisy environment). The number of utterances per dataset in CHiME-3 is summarized in Table 6.2. For a more detailed description on the CHiME-3 framework the reader is referred to [15].

According to preliminary experiments, speech captured by the microphone that faces backwards is quite attenuated and yields a lower SNR with respect to the rest of sensors. This is reasonable as that microphone is placed in an acoustic shadow regarding the speaker's mouth. To show this, we have estimated the average SNR of every channel from the real development dataset by also taking advantage of the close-talk microphone data provided with CHiME-3 (see Figure 6.6). As can be observed, the microphone numbered as 2, or secondary microphone, yields the lowest SNR.

6.1.2 Feature extraction

In this Thesis, the European Telecommunications Standards Institute front-end (ETSI FE), ETSI ES 201 108, has been used to extract acoustic features from the speech signals [2, 149]. This front-end is intended to the extraction of Mel-cepstral features and briefly consists of the following steps. First, a signal coming from either the AURORA2-2C-CT/FT databases (with a sampling rate of 8 kHz) or the CHiME-3

corpus (with a sampling rate of 16 kHz) is filtered to remove its offset. Second, this filtered signal is divided into overlapping frames of 25 ms each with a shift interval of 10 ms. A pre-emphasis filtering with a filter coefficient value of $\mu = 0.97$ is applied to the framed signal, which is then windowed by using a Hamming window. After increasing the length of each windowed frame by zero-padding, the magnitude spectrum of the signal is computed by application of the fast Fourier transform (FFT).

A Mel filterbank covering the frequency range from 64 Hz to 4 kHz or 8 kHz (depending on the sampling frequency of the input signal) is employed to transform the magnitude spectrum to the Mel-frequency domain. In particular, the frequency range is divided into 23 equidistant channels (according to the Mel scale) with triangular-shaped frequency windows each. A non-linear operation consisting of the natural logarithm is applied to the Mel-filtered outputs and, then, 13 Mel-frequency cepstral coefficients (MFCCs) are computed per frame through the discrete cosine transform (DCT).

For the case of the AURORA2-2C-CT/FT corpora, velocity and acceleration coefficients are calculated from the above 13 MFCCs in such a way that all of them are stacked to form the 39-dimensional feature vector used by the recognizer. Finally, to improve the robustness of the system against channel mismatches, cepstral mean normalization (CMN) is applied.

In the case of CHiME-3, we should first distinguish between GMM- and DNN-based acoustic models, since both types can be used for evaluation as detailed below. For GMM-based acoustic modeling, three frames from the left and right temporal context are appended to each frame, which defines an augmented 91-dimensional MFCC feature vector. Then, a linear discriminant analysis (LDA) procedure (to reduce the number of components of the augmented feature vector to only 40) as well as maximum likelihood linear transformation (MLLT) and feature-space maximum likelihood linear regression (fMLLR) with speaker adaptive training (SAT) are applied [15]. The result is then used to train the GMM-HMM-based recognizer. Finally, for DNN-based acoustic modeling, five frames from the left and right temporal context are appended to each frame, which generates an augmented 143-dimensional MFCC feature vector that is directly used as input to the DNN.

6.1.3 Back-end

The back-ends for all the AURORA2-2C-CT/FT and CHiME-3 corpora are based on hidden Markov models (HMMs) trained with either clean or multi-style data. It should be reminded that to train multi-style acoustic models (intended to strengthen the ASR system against noisy conditions), the corresponding multi-condition training dataset is first enhanced by means of the particular technique that shall be evaluated. The

characteristics of each back-end depending on the database are explained immediately below.

6.1.3.1 AURORA2-2C-CT/FT

For these databases, each digit is modeled by a left-to-right (i.e. Bakis) continuous density HMM with 16 states plus beginning and ending states emitting no output. Skips over states are avoided. Only GMM-based acoustic models are considered such that a mixture of 3 Gaussians (with diagonal covariance matrices) per state is defined. The pauses at the beginning and end of an utterance are similarly modeled by an HMM silence model with 3 emitting states and 6 Gaussians per state. Moreover, the short pauses between digits are modeled by an HMM with 1 emitting state with 6 Gaussians which is tied to the middle state of the silence model. Acoustic model training is performed by means of several iterations of the Baum-Welch algorithm. The implementation of this back-end is done by employing the HTK 3.4 toolkit [204].

6.1.3.2 CHiME-3

For CHiME-3 we employ the Kaldi toolkit [152]-based ASR system of [15], where both GMM- and DNN-based acoustic models are considered for this task. Initially, 2500 different tied triphone HMM states, which are modeled by a total of 15000 Gaussians [15], are trained. Then, for DNN-based acoustic modeling, the Kaldi recipe for Track 2 of the 2nd CHiME Challenge is followed [196]. A DNN with 7 hidden layers with 2048 neurons each is employed. A generative pre-training using restricted Boltzmann machines (RBMs) as well as cross-entropy and sequence-discriminative training employing the state-level minimum Bayes risk (sMBR) criterion [184] are performed on the DNN [15]. It should be noted that, in first instance, the DNN is trained from the alignments generated by the above GMM-HMM-based ASR system. Then, once the DNN is trained, realignments are done and the DNN is re-trained from these new alignments. This procedure is repeated until completing four iterations.

6.2 Experiments and results

Different types of quality measures can be employed to determine the goodness or usefulness of an ASR system in both absolute and relative (when compared with other systems) terms. For sure, the most important performance metric is the accuracy of the recognizer, which is usually measured through the word accuracy (WAcc) and/or

the word error rate (WER) metrics. These are the ones chosen to evaluate our noise-robust contributions when integrated in an ASR system, and they are presented in the following subsection along with their confidence intervals. Furthermore, another issue that may be interesting to be controlled in this kind of systems is the computational complexity of the algorithms. However, we have found that this aspect is not so critical as in the past, since it seems that the preferred mobile ASR architecture nowadays is the NSR (Network-based Speech Recognition) one in contrast to the DSR architecture or embedded systems. That is, unlike in DSR, signal processing is carried out on the server side in NSR, where there are available more computational resources than in the mobile terminals. Thus, Subsections 6.2.2, 6.2.3 and 6.2.4 present, by following the same sequence as in the previous chapters, the word accuracy and word error rate results obtained by our techniques and those for comparison, as well as an analysis on such results.

6.2.1 Recognition accuracy and confidence intervals

As it is standard for the Aurora-2 corpus and the CHiME-3 database, the noise-robust methods will be evaluated in terms of word recognition accuracy or word error rate when considering the AURORA2-2C-CT/FT corpora or the CHiME-3 database, respectively. Let us suppose that a reference sentence, the correct transcription of which is known, contains a total of N_T words. If an ASR system, when trying to recognize and transcribe such a reference sentence, substitutes N_S words, deletes N_D words and inserts N_I words, the word recognition accuracy from that system is

$$\text{WAcc} = \frac{N_T - N_S - N_D - N_I}{N_T}. \tag{6.1}$$

For its part, the word error rate metric is defined as

$$\text{WER} = 1 - \text{WAcc} = \frac{N_S + N_D + N_I}{N_T}. \tag{6.2}$$

In order to find out the WAcc or WER given the transcription of the reference sentence and that recognized by the ASR system, dynamic programming is used. It is common to provide the WAcc or WER measures in terms of percentage. Moreover, due to the insertion errors, WAcc (WER) may be even negative (over 100%).

When comparing two different ASR systems, it is very important to determine if their differences in terms of the considered performance metric, e.g. WAcc or WER, are statistically significant. For every database we are able to calculate a confidence interval in such a way that we can assure that we are $100(1-\alpha)\%$ confident that

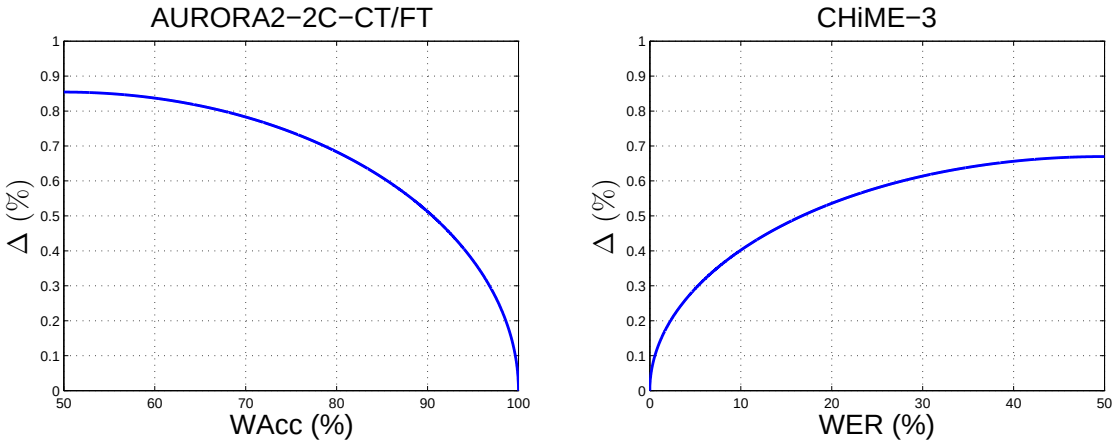

Figure 6.7: Amplitude of the 95% confidence interval as a function of WAcc for the test sets of the AURORA2-2C-CT/FT databases (left) and WER for the real data evaluation set of the CHiME-3 corpus (right).

the true value of either WAcc or WER is in that interval, where α is the significance level. The amplitude of this confidence interval in terms of WAcc (it can be expressed analogously for WER) is

$$\Delta = z_{1-\alpha/2}\sqrt{\frac{\text{WAcc}(1-\text{WAcc})}{N}}, \tag{6.3}$$

where N is the total number of different realizations of words to be tested, i.e. 13159 and 21409 for the AURORA2-2C-CT/FT and the CHiME-3 corpora, respectively. Furthermore, $z_{1-\alpha/2}$ is the z-score for the $100(1-\alpha/2)\%$ percentile point, which is approximately 1.96 for a typical significance level of $\alpha = 0.05$ (equivalently, a 95% confidence value). From (6.3), we can observe that the greater the number of tested words, N, the smaller the amplitude of the confidence interval. Moreover, the greater (smaller) the WAcc (WER), the smaller the amplitude of the confidence interval as well. In order to serve as a reference along with the results presented throughout the following subsections, Figure 6.7 depicts the amplitude of the 95% confidence interval as a function of WAcc, for the test sets of the AURORA2-2C-CT/FT databases, and WER, for the real data evaluation set of the CHiME-3 corpus. In addition, some of these 95% confidence interval amplitudes are shown in Table 6.3 for typical WAcc and WER values.

6.2.2 Power spectrum enhancement techniques

We should recall here that the primary channel in the AURORA2-2C-CT/FT databases is identified with the primary microphone of the smartphone. On the contrary, in

AURORA2-2C-CT/FT (WAcc±Δ, %)	95±0.37	90±0.51	85±0.61	80±0.68	75±0.74	70±0.78	65±0.82
CHiME-3 (WER±Δ, %)	5±0.29	10±0.40	15±0.48	20±0.54	25±0.58	30±0.61	35±0.64

Table 6.3: 95% confidence interval amplitudes for some typical WAcc (for the test sets of the AURORA2-2C-CT/FT databases) and WER (for the real data evaluation set of the CHiME-3 corpus) values.

CHiME-3, a virtual primary channel is obtained by means of MVDR beamforming from all the six microphones in the tablet. With this arrangement, our dual-channel power spectrum enhancement techniques act as beamformer post-filters which help to mitigate the beamformer weak points such as its poor performance at low frequencies or the effect of noise sources placed along the steering direction [103]. The use of MVDR in particular will be justified in the CHiME-3 results section.

In addition to our power spectrum enhancement contributions, DCSS, P-MVDR and DSW, other single-channel and multi-channel noise-robust techniques are evaluated for comparison purposes. First, the following three single-channel noise-robust methods are tested on the primary channel: a soft-mask weighting (SMW) technique in the log-Mel domain [91], the ETSI advanced front-end (AFE) [3] and a classical Wiener filtering with the same post-processing as in Subsection 3.3.4 (Wiener+Int). For Wiener+Int, the noise PSD $\mathcal{S}_{n_1}(f,t)$ in Eq. (3.28) is approximated from noise estimates obtained again by means of linear interpolation in the log-power spectral domain. This interpolation uses the averages of the first and last M frames in each utterance. Noise estimation by linear interpolation is selected for a fair comparison, as the noise statistical parameters required by our techniques are obtained by following this same approach (as explained throughout Chapter 3).

Moreover, two beamforming techniques are tested for comparison as well: delay-and-sum (D&S) [211] and MVDR [77]. On the one hand, time difference of arrival (TDoA) estimation for D&S is performed as explained in [15] and [21]. On the other hand, the tested MVDR applies an eigenvalue decomposition-based technique, where the clean speech spatial covariance matrix is derived from complex GMM-based time-frequency (T-F) masks, to estimate the steering vector [77]. In addition, two post-filtering methods are evaluated in CHiME-3 when applied after MVDR beamforming. The first one consists of a multi-channel Wiener post-filter (Lefkimmiatis) [103] using a noise coherence matrix estimated per utterance and frequency bin from the noise spatial correlation matrix as computed for MVDR beamforming [77]. The second considered post-filter is a multi-channel noise reduction post-filter as in [210] (MCNR-like) which employs the steering vector from [77] for a fair evaluation. The baseline system uses

Parameter	M	η	η_ξ	η_n	M_f	M_t	G_f	G_t	σ_G
Value	20	0.1	-3.35 dB	10^3	3	5	5	5	1

Table 6.4: Values chosen for the parameters used by our dual-channel power spectrum enhancement contributions.

noisy speech features from the primary channel in the case of the AURORA2-2C-CT/FT databases and from the fifth microphone in the case of CHiME-3 (as in [15]). Finally, for comparison purposes, not only our MMSE-based RSG (Relative Speech Gain) estimation method is considered but also the aforementioned method based on eigenvalue decomposition (ED) [77]. The latter will only be combined with DSW as this technique performs the best among our dual-channel power spectrum enhancement developments. Thus, our biased WF-based spectral weighting approach (DSW-B) is evaluated along with its unbiased version with noise equalization (DSW-(U+Eq)$_{MMSE}$ and DSW-(U+Eq)$_{ED}$) and without it (DSW-U$_{MMSE}$ and DSW-U$_{ED}$).

The parameters of the probability density functions (PDFs) $p(\mathbf{a}_{21}^r) = \mathcal{N}\left(\boldsymbol{\mu}_{A_{21}^r}, \boldsymbol{\Sigma}_{A_{21}^r}\right)$ and $p(\mathbf{a}_{21}^i) = \mathcal{N}\left(\boldsymbol{\mu}_{A_{21}^i}, \boldsymbol{\Sigma}_{A_{21}^i}\right)$ used to estimate the variable $\mathcal{A}_{21}(f,t)$ in Section 3.4 are computed in advance for the AURORA2-2C-CT and AURORA2-2C-FT databases (separately), and the CHiME-3 corpus. Furthermore, in the case of CHiME-3, such parameters are obtained for the simulated and real cases independently. In this work both $p(\mathbf{a}_{21}^r)$ and $p(\mathbf{a}_{21}^i)$ are assumed to be stationary distributions. Thus, the mean vectors $\boldsymbol{\mu}_{A_{21}^r}$ and $\boldsymbol{\mu}_{A_{21}^i}$, and the covariance matrices $\boldsymbol{\Sigma}_{A_{21}^r}$ and $\boldsymbol{\Sigma}_{A_{21}^i}$, are obtained as the sample means and sample covariances, respectively, from $\mathbf{a}_{21}$ samples. That is, $\mathbf{a}_{21}^r$ and $\mathbf{a}_{21}^i$ at every time frame t are considered realizations of the variables. For all the AURORA2-2C-CT/FT and CHiME-3 corpora, $\mathbf{a}_{21}$ samples are obtained from their corresponding development datasets in the knowledge that $\mathbf{a}_{21} = \mathbf{x}_2 \oslash \mathbf{x}_1$, where $\oslash$ symbolizes element-wise division. In addition, statistical independence between frequency bins was assumed so that $\boldsymbol{\Sigma}_{A_{21}^r}$ and $\boldsymbol{\Sigma}_{A_{21}^i}$ are diagonal covariance matrices.

The values of the parameters employed by our different dual-channel power spectrum enhancement contributions can be seen in Table 6.4. Thus, M corresponds to the first and last 200 ms of the utterance, where it was considered that speech is absent. Furthermore, η was selected by means of preliminary speech recognition experiments over development datasets, what also fixes the value of η_ξ in accordance with (3.54). The value of η_n roughly corresponds to an SNR of 40 dB. Finally, the values for the parameters of the filters intended to improve the spectro-temporal coherence of the spectral weights in Subsection 3.3.4 were similarly chosen as in [91].

		SNR (dB)								
Method	**Test set**	-5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	21.14	38.19	64.60	87.71	95.99	98.13	99.13	67.63	-
	B	15.15	25.50	47.61	77.84	93.44	97.38	99.13	59.49	-
	Avg.	18.15	31.85	56.11	82.78	94.72	97.76	99.13	63.56	-
AFE	*A*	**44.31**	**70.63**	**87.71**	94.93	97.33	**98.53**	**99.24**	**82.24**	**14.61**
	B	**27.31**	**60.29**	**82.61**	92.67	96.58	98.13	**99.24**	**76.27**	**16.78**
	Avg.	**35.81**	**65.46**	**85.16**	93.80	96.96	**98.33**	**99.24**	**79.25**	**15.69**
SMW	*A*	33.44	58.61	80.30	90.60	94.73	96.53	98.40	75.70	8.07
	B	19.02	44.91	73.75	88.37	93.64	95.64	98.40	69.22	9.73
	Avg.	26.23	51.76	77.03	89.49	94.19	96.09	98.40	72.47	8.91
Wiener+Int	*A*	33.65	58.59	80.77	92.29	96.41	98.03	99.08	76.62	8.99
	B	19.72	40.04	70.16	88.00	95.11	97.39	99.08	68.40	8.91
	Avg.	26.69	49.32	75.47	90.15	95.76	97.71	99.08	72.52	8.96
D&S	*A*	14.38	25.89	46.83	75.74	92.71	97.39	99.04	58.82	-8.81
	B	10.14	17.46	32.15	59.72	87.28	96.30	99.04	50.51	-8.98
	Avg.	12.26	21.68	39.49	67.73	90.00	96.85	99.04	54.67	-8.89
MVDR	*A*	16.86	34.30	64.47	89.24	96.35	98.31	99.05	66.59	-1.04
	B	10.86	20.88	49.82	82.33	94.63	97.65	99.05	59.36	-0.13
	Avg.	13.86	27.59	57.15	85.79	95.49	97.98	99.05	62.98	-0.58
DCSS	*A*	29.88	55.22	79.69	92.52	96.61	97.85	98.63	75.30	7.67
	B	18.42	37.02	69.14	87.93	95.11	97.40	98.63	67.50	8.01
	Avg.	24.15	46.12	74.42	90.23	95.86	97.63	98.63	71.40	7.84
P-MVDR	*A*	29.47	54.50	79.20	92.41	96.62	98.03	98.98	75.04	7.41
	B	18.34	36.64	68.71	87.74	95.13	97.47	98.98	67.34	7.85
	Avg.	23.91	45.57	73.96	90.08	95.88	97.75	98.98	71.19	7.63
DSW-B	*A*	35.62	62.47	83.94	93.76	96.97	98.17	99.05	78.49	10.86
	B	22.98	45.40	75.31	90.34	96.01	97.73	99.05	71.30	11.81
	Avg.	29.30	53.94	79.63	92.05	96.49	97.95	99.05	74.89	11.33
DSW-U$_{MMSE}$	*A*	35.76	62.50	83.69	93.63	96.89	98.13	98.99	78.43	10.80
	B	22.84	45.81	75.15	90.22	95.82	97.61	98.99	71.24	11.75
	Avg.	29.30	54.16	79.42	91.93	96.36	97.87	98.99	74.84	11.28
DSW-(U+Eq)$_{MMSE}$	*A*	39.46	67.07	87.28	**95.28**	**97.42**	98.44	99.02	80.83	13.20
	B	24.00	49.31	80.22	**92.72**	**97.01**	**98.19**	99.02	73.58	14.09
	Avg.	31.73	58.19	83.75	**94.00**	**97.22**	98.32	99.02	77.20	13.64

Table 6.5: Word accuracy results (in terms of percentage and for different SNR values) obtained for our power spectrum enhancement proposals and comparison techniques evaluated on the AURORA2-2C-CT (close-talk) database when using clean acoustic models.

6.2.2.1 AURORA2-2C-CT/FT results

Tables 6.5 and 6.6 summarize the word accuracy results obtained for the AURORA2-2C-CT database (close-talk) when clean and multi-style acoustic models are employed, respectively. Results are broken down by SNR and averaged across all types of noise in test sets *A* and *B*. As can be observed, the best result is obtained with multi-style acoustic models by our unbiased spectral weighting with noise equalization considering our RSG estimation method (DSW-(U+Eq)$_{MMSE}$), yielding a relative average improvement regarding the baseline of 6.73%. Moreover, this approach also presents the best behavior at the most adverse acoustic condition tested (-5 dB) with an absolute word accuracy of 53.95% and a relative improvement of 17.02% with respect to the baseline.

The word accuracy results obtained for the AURORA2-2C-FT database (far-talk), when clean and multi-style acoustic models are employed, are shown in Tables 6.7

		SNR (dB)								
Method	**Test set**	-5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	47.64	76.99	92.36	96.94	97.98	98.49	98.77	85.07	-
	B	26.22	56.39	85.33	94.52	97.14	98.12	98.77	76.29	-
	Avg.	36.93	66.69	88.85	95.73	97.56	98.31	98.77	80.68	-
AFE	*A*	57.36	82.98	94.13	97.08	98.38	**98.82**	**99.07**	88.13	3.06
	B	39.06	73.74	90.35	95.99	97.83	98.49	**99.07**	82.58	6.29
	Avg.	48.21	78.36	92.24	96.54	98.11	98.66	**99.07**	85.35	4.67
SMW	*A*	47.95	75.81	90.90	96.02	97.52	98.09	98.77	84.38	-0.69
	B	26.77	60.92	83.14	92.92	96.30	97.81	98.77	76.31	0.02
	Avg.	37.36	68.37	87.02	94.47	96.91	97.95	98.77	80.35	-0.33
Wiener+Int	*A*	58.79	82.85	94.06	96.95	98.00	98.37	98.90	88.17	3.10
	B	35.70	69.70	89.30	95.60	97.33	98.20	98.90	80.97	4.68
	Avg.	47.25	76.28	91.68	96.28	97.67	98.29	98.90	84.58	3.90
D&S	*A*	30.67	57.76	83.43	94.16	97.08	97.83	98.44	76.82	-8.25
	B	14.93	34.77	71.87	90.26	95.79	97.54	98.44	67.53	-8.76
	Avg.	22.80	46.27	77.65	92.21	96.44	97.69	98.44	72.18	-8.50
MVDR	*A*	34.00	67.85	90.33	97.05	98.23	98.79	98.79	81.04	-4.03
	B	16.01	48.61	83.92	94.70	97.34	98.40	98.79	73.16	-3.13
	Avg.	25.01	58.23	87.13	95.88	97.79	98.60	98.79	77.11	-3.57
DCSS	*A*	57.29	84.51	95.08	97.78	**98.47**	98.63	98.49	88.63	3.56
	B	36.04	70.95	90.90	96.24	97.99	**98.72**	98.49	81.81	5.52
	Avg.	46.67	77.73	92.99	97.01	98.23	**98.68**	98.49	85.22	4.54
P-MVDR	*A*	56.56	84.12	94.87	97.71	98.40	98.76	98.71	88.40	3.33
	B	35.30	70.08	90.52	96.03	97.96	98.46	98.71	81.39	5.10
	Avg.	45.93	77.10	92.70	96.87	98.18	98.61	98.71	84.90	4.22
DSW-B	*A*	63.05	87.00	95.47	97.75	98.27	98.70	98.87	90.04	4.97
	B	41.19	73.98	90.90	96.04	97.97	98.51	98.87	83.10	6.81
	Avg.	52.12	80.49	93.19	96.90	98.12	98.61	98.87	86.57	5.89
DSW-U$_{MMSE}$	*A*	62.89	86.87	95.46	97.78	98.21	98.62	98.59	89.97	4.90
	B	41.28	74.48	90.71	95.99	97.90	98.40	98.59	83.13	6.84
	Avg.	52.09	80.68	93.09	96.89	98.06	98.51	98.59	86.55	5.87
DSW-(U+Eq)$_{MMSE}$	*A*	**65.02**	**88.15**	**95.84**	**97.88**	98.43	98.60	98.65	**90.65**	**5.58**
	B	**42.87**	**76.50**	**92.24**	**96.62**	**98.15**	98.58	98.65	**84.16**	**7.87**
	Avg.	**53.95**	**82.33**	**94.04**	**97.25**	**98.29**	98.59	98.65	**87.41**	**6.73**

Table 6.6: Word accuracy results (in terms of percentage and for different SNR values) obtained for our power spectrum enhancement proposals and comparison techniques evaluated on the AURORA2-2C-CT (close-talk) database when using multi-style acoustic models.

and 6.8, respectively. With a relative average improvement of 5.18% with respect to the baseline system, DSW-(U+Eq)$_{MMSE}$ using multi-style models is again the best approach according to the results. In addition, with an absolute word accuracy of 53.11% and a relative improvement of 14.69% regarding the baseline, DSW-(U+Eq)$_{MMSE}$ with multi-style acoustic models is the best option at -5 dB as well.

While DSW-(U+Eq)$_{MMSE}$ with multi-style acoustic models achieves on average the highest results, the AFE performs the best on AURORA2-2C-CT/FT when clean acoustic models are employed. We should take into account that the AFE is shown only as a reference as it involves multiple state-of-the-art strategies (e.g., a sophisticated two stage Mel-warped Wiener filter approach which uses a voice activity detector (VAD), and waveform processing and blind equalization stages [3, 149]), which are not incompatible with our developments.

It is worth noticing that beamforming techniques do not provide a successful perfor-

		SNR (dB)								
Method	**Test set**	-5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	26.01	43.52	66.77	88.41	96.23	98.18	99.10	69.85	-
	B	16.24	26.54	51.14	81.06	94.43	97.81	99.10	61.20	-
	Avg.	21.13	35.03	58.96	84.74	95.33	98.00	99.10	65.53	-
AFE	*A*	**48.36**	**72.71**	**87.95**	**95.39**	97.61	98.29	**99.24**	**83.39**	**13.54**
	B	**30.38**	**63.40**	**85.37**	**93.79**	**96.99**	**98.38**	**99.24**	**78.05**	**16.85**
	Avg.	**39.37**	**68.06**	**86.66**	**94.59**	**97.30**	98.34	**99.24**	**80.72**	**15.19**
SMW	*A*	37.44	61.04	81.18	91.04	94.97	96.59	98.40	77.04	7.19
	B	20.67	45.62	74.36	87.99	93.32	95.60	98.40	69.59	8.39
	Avg.	29.06	53.33	77.77	89.52	94.15	96.10	98.40	73.32	7.79
Wiener+Int	*A*	39.65	62.47	81.64	92.82	96.78	98.14	99.08	78.58	8.73
	B	21.78	44.83	74.13	90.50	96.16	98.11	99.08	70.92	9.72
	Avg.	30.72	53.65	77.89	91.66	96.47	98.13	99.08	74.75	9.22
D&S	*A*	21.80	39.13	62.98	86.64	96.14	98.10	99.07	67.47	-2.38
	B	12.79	23.05	46.73	78.01	94.11	97.87	99.07	58.76	-2.44
	Avg.	17.30	31.09	54.86	82.33	95.13	97.99	99.07	63.12	-2.41
MVDR	*A*	24.86	48.15	77.26	93.32	**97.73**	**98.63**	99.17	73.33	3.48
	B	13.59	27.95	66.09	89.54	96.58	98.37	99.17	65.35	4.15
	Avg.	19.23	38.05	71.68	91.43	97.16	**98.50**	99.17	69.34	3.81
DCSS	*A*	33.80	55.74	76.15	89.10	94.63	97.01	97.92	74.41	4.56
	B	18.64	36.42	65.13	85.56	93.39	96.48	97.92	65.94	4.74
	Avg.	26.22	46.08	70.64	87.33	94.01	96.75	97.92	70.17	4.64
P-MVDR	*A*	32.01	54.28	76.24	90.24	95.70	97.79	98.88	74.38	4.53
	B	18.01	34.09	63.38	85.80	94.54	97.29	98.88	65.52	4.32
	Avg.	25.01	44.19	69.81	88.02	95.12	97.54	98.88	69.95	4.42
DSW-B	*A*	36.87	59.27	79.95	92.27	96.52	97.92	99.07	77.13	7.28
	B	21.00	39.16	69.05	88.59	95.50	97.79	99.07	68.52	7.32
	Avg.	28.94	49.22	74.50	90.43	96.01	97.86	99.07	72.83	7.30
DSW-U$_{MMSE}$	*A*	39.18	61.18	80.19	91.97	96.25	97.76	98.99	77.76	7.91
	B	21.87	41.15	70.35	88.67	95.36	97.71	98.99	69.19	7.99
	Avg.	30.53	51.17	75.27	90.32	95.81	97.74	98.99	73.47	7.94
DSW-(U+Eq)$_{MMSE}$	*A*	39.20	62.12	81.42	93.17	96.79	98.16	98.99	78.48	8.63
	B	22.17	43.83	73.73	90.55	95.84	97.86	98.99	70.66	9.46
	Avg.	30.69	52.98	77.58	91.86	96.32	98.01	98.99	74.57	9.04

Table 6.7: Word accuracy results (in terms of percentage and for different SNR values) obtained for our power spectrum enhancement proposals and comparison techniques evaluated on the AURORA2-2C-FT (far-talk) database when using clean acoustic models.

mance. Indeed, D&S yields a drop in performance since it only aligns the target signals from each channel leading to the primary channel being combined with a much noisier secondary one. On the other hand, while MVDR beamforming additionally manages both the speech gains (through the steering vector) and the noise signals, it is only able to achieve a modest improvement, regarding the baseline, under clean acoustic modeling in far-talk conditions. These results are coherent with the fact that poor performance of the classical beamforming techniques can be expected with only two microphones very close each other, one of them also placed in an acoustic shadow with respect to the target signal (i.e. the secondary sensor) [179, 180]. Indeed, beamforming results are especially poor in close-talk conditions and are, at the same time, coherent with the fact that speech is much more attenuated at the secondary sensor in close- than in far-talk position.

On average, DSW-(U+Eq)$_{MMSE}$ always outperforms both DCSS and P-MVDR,

		SNR (dB)								
Method	**Test set**	-5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	49.66	76.86	92.68	97.08	98.06	98.57	98.76	85.49	-
	B	27.18	58.76	86.93	95.32	97.53	98.35	98.76	77.35	-
	Avg.	38.42	67.81	89.81	96.20	97.80	98.46	98.76	81.42	-
AFE	*A*	59.61	83.33	94.03	97.52	98.41	98.69	**99.06**	88.60	3.11
	B	**41.66**	**75.59**	91.05	96.18	98.02	**98.60**	**99.06**	**83.52**	**6.17**
	Avg.	50.64	79.46	92.54	96.85	98.22	98.65	**99.06**	86.06	4.64
SMW	*A*	47.67	75.96	91.38	96.16	97.67	98.40	97.23	84.54	-0.95
	B	25.12	57.58	83.99	93.28	96.62	97.72	97.23	75.72	-1.63
	Avg.	36.40	66.77	87.69	94.72	97.15	98.06	97.23	80.13	-1.29
Wiener+Int	*A*	60.38	82.65	93.93	97.20	98.10	98.36	98.81	88.44	2.95
	B	37.27	71.11	89.98	95.92	97.48	98.27	98.81	81.67	4.32
	Avg.	48.83	76.88	91.96	96.56	97.79	98.32	98.81	85.06	3.64
D&S	*A*	39.16	68.87	89.83	96.57	98.01	98.51	98.60	81.83	-3.66
	B	20.11	46.77	79.51	94.36	97.56	98.57	98.60	72.81	-4.54
	Avg.	29.64	57.82	84.67	95.47	97.79	98.54	98.60	77.32	-4.10
MVDR	*A*	42.20	76.51	93.66	**97.60**	**98.52**	**98.76**	98.85	84.54	-0.95
	B	21.28	59.89	89.49	**96.33**	**98.03**	98.58	98.85	77.27	-0.08
	Avg.	31.74	68.20	91.58	**96.97**	**98.28**	**98.67**	98.85	80.91	-0.51
DCSS	*A*	59.27	83.25	94.35	97.37	98.14	98.51	98.45	88.48	2.99
	B	35.63	70.13	89.74	95.75	97.45	98.28	98.45	81.16	3.81
	Avg.	47.45	76.69	92.05	96.56	97.80	98.40	98.45	84.83	3.41
P-MVDR	*A*	59.05	82.97	94.15	97.30	98.04	98.44	98.54	88.33	2.84
	B	34.77	69.77	89.58	95.85	97.43	98.28	98.54	80.95	3.60
	Avg.	46.91	76.37	91.87	96.57	97.74	98.36	98.54	84.64	3.22
DSW-B	*A*	60.22	83.49	94.38	97.51	98.25	98.65	98.79	88.75	3.26
	B	36.18	67.72	89.10	95.69	97.56	98.46	98.79	80.79	3.44
	Avg.	48.20	75.61	91.74	96.60	97.91	98.56	98.79	84.77	3.35
DSW-U$_{MMSE}$	*A*	62.05	84.45	94.49	97.35	98.21	98.73	98.69	89.21	3.72
	B	38.13	69.56	89.57	95.79	97.64	98.44	98.69	81.52	4.17
	Avg.	50.09	77.01	92.03	96.57	97.93	98.59	98.69	85.37	3.95
DSW-(U+Eq)$_{MMSE}$	*A*	**65.11**	**86.19**	**95.00**	97.59	98.27	98.70	98.57	**90.14**	**4.65**
	B	41.10	73.44	**91.16**	96.27	97.76	98.55	98.57	83.05	5.70
	Avg.	**53.11**	**79.82**	**93.08**	96.93	98.02	98.63	98.57	**86.60**	**5.18**

Table 6.8: Word accuracy results (in terms of percentage and for different SNR values) obtained for our power spectrum enhancement proposals and comparison techniques evaluated on the AURORA2-2C-FT (far-talk) database when using multi-style acoustic models.

which can also be interpreted as spectral weighting techniques. As can be observed, on average, P-MVDR outperforms MVDR in all conditions. It is interesting to see that, as opposed to MVDR beamforming, the greatest relative improvements achieved by P-MVDR are in close-talk conditions. Thus, we can consider P-MVDR as an *ad-hoc* MVDR to be used with small microphone arrays. We also confirm that discarding the phase information as used by MVDR beamforming is positive. In accordance with the analysis presented in Subsection 3.2.3, DCSS and P-MVDR in general provide very similar results since the term $\mathcal{A}_{21}(f,t)$ has a relatively small magnitude. As can be seen, this statement is valid not only for close-talk position, but also for far-talk conditions. In this respect, we should recall that the clean speech secondary signals for the AURORA2-2C-FT corpus were generated in an anechoic chamber environment, where, theoretically, there are no signal reflections but only diffraction. Indeed, in this context, $\mathcal{A}_{21}(f,t)$ is still relatively small.

		SNR (dB)								
Method	**Test set**	-5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	21.14	38.19	64.60	87.71	95.99	98.13	**99.13**	67.63	-
	B	15.15	25.50	47.61	77.84	93.44	97.38	**99.13**	59.49	-
	Avg.	18.15	31.85	56.11	82.78	94.72	97.76	**99.13**	63.56	-
DSW-B	*A*	35.62	62.47	83.94	93.76	96.97	98.17	99.05	78.49	10.86
	B	22.98	45.40	75.31	90.34	96.01	97.73	99.05	71.30	11.81
	Avg.	29.30	53.94	79.63	92.05	96.49	97.95	99.05	74.89	11.33
DSW-U_{ED}	*A*	31.95	59.38	83.16	93.57	96.88	98.12	98.98	77.18	9.55
	B	20.05	42.94	74.27	90.00	95.80	97.65	98.98	70.12	10.63
	Avg.	26.00	51.16	78.72	91.79	96.34	97.89	98.98	73.65	10.09
DSW-U_{MMSE}	*A*	35.76	62.50	83.69	93.63	96.89	98.13	98.99	78.43	10.80
	B	22.84	45.81	75.15	90.22	95.82	97.61	98.99	71.24	11.75
	Avg.	29.30	54.16	79.42	91.93	96.36	97.87	98.99	74.84	11.28
DSW-$(U+Eq)_{ED}$	*A*	33.01	61.26	85.54	95.06	**97.47**	**98.48**	99.04	78.47	10.84
	B	19.13	42.87	77.78	92.35	96.97	**98.25**	99.04	71.23	11.74
	Avg.	26.07	52.07	81.66	93.71	**97.22**	**98.37**	99.04	74.85	11.29
DSW-$(U+Eq)_{MMSE}$	*A*	**39.46**	**67.07**	**87.28**	**95.28**	97.42	98.44	99.02	**80.83**	**13.20**
	B	**24.00**	**49.31**	**80.22**	**92.72**	**97.01**	98.19	99.02	**73.58**	**14.09**
	Avg.	**31.73**	**58.19**	**83.75**	**94.00**	**97.22**	98.32	99.02	**77.20**	**13.64**

Table 6.9: Comparison between an ED-based steering vector computation method and our MMSE-based RSG estimation technique in terms of word accuracy (%) when combined with our dual-channel spectral weighting evaluated on the AURORA2-2C-CT (close-talk) database when using clean acoustic models.

		SNR (dB)								
Method	**Test set**	-5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	47.64	76.99	92.36	96.94	97.98	98.49	98.77	85.07	-
	B	26.22	56.39	85.33	94.52	97.14	98.12	98.77	76.29	-
	Avg.	36.93	66.69	88.85	95.73	97.56	98.31	98.77	80.68	-
DSW-B	*A*	63.05	87.00	95.47	97.75	98.27	**98.70**	**98.87**	90.04	4.97
	B	41.19	73.98	90.90	96.04	97.97	98.51	**98.87**	83.10	6.81
	Avg.	52.12	80.49	93.19	96.90	98.12	98.61	**98.87**	86.57	5.89
DSW-U_{ED}	*A*	55.35	84.26	95.07	97.70	98.20	98.62	98.68	88.20	3.13
	B	34.09	70.59	90.14	95.87	97.81	98.39	98.68	81.15	4.86
	Avg.	44.72	77.43	92.61	96.79	98.01	98.51	98.68	84.68	4.00
DSW-U_{MMSE}	*A*	62.89	86.87	95.46	97.78	98.21	98.62	98.59	89.97	4.90
	B	41.28	74.48	90.71	95.99	97.90	98.40	98.59	83.13	6.84
	Avg.	52.09	80.68	93.09	96.89	98.06	98.51	98.59	86.55	5.87
DSW-$(U+Eq)_{ED}$	*A*	52.93	83.68	95.24	97.70	98.37	98.64	98.57	87.76	2.69
	B	30.59	69.44	90.82	96.13	98.03	98.48	98.57	80.58	4.29
	Avg.	41.76	76.56	93.03	96.92	98.20	98.56	98.57	84.17	3.49
DSW-$(U+Eq)_{MMSE}$	*A*	**65.02**	**88.15**	**95.84**	**97.88**	**98.43**	98.60	98.65	**90.65**	**5.58**
	B	**42.87**	**76.50**	**92.24**	**96.62**	**98.15**	**98.58**	98.65	**84.16**	**7.87**
	Avg.	**53.95**	**82.33**	**94.04**	**97.25**	**98.29**	**98.59**	98.65	**87.41**	**6.73**

Table 6.10: Comparison between an ED-based steering vector computation method and our MMSE-based RSG estimation technique in terms of word accuracy (%) when combined with our dual-channel spectral weighting evaluated on the AURORA2-2C-CT (close-talk) database when using multi-style acoustic models.

		SNR (dB)								
Method	**Test set**	-5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	26.01	43.52	66.77	88.41	96.23	98.18	**99.10**	69.85	-
	B	16.24	26.54	51.14	81.06	94.43	97.81	**99.10**	61.20	-
	Avg.	21.13	35.03	58.96	84.74	95.33	98.00	**99.10**	65.53	-
DSW-B	*A*	36.87	59.27	79.95	92.27	96.52	97.92	99.07	77.13	7.28
	B	21.00	39.16	69.05	88.59	95.50	97.79	99.07	68.52	7.32
	Avg.	28.94	49.22	74.50	90.43	96.01	97.86	99.07	72.83	7.30
DSW-U_{ED}	*A*	35.95	58.95	79.39	91.88	96.22	97.73	98.97	76.69	6.84
	B	19.17	37.89	68.49	90.63	95.19	97.65	98.97	68.17	6.97
	Avg.	27.56	48.42	73.94	91.26	95.71	97.69	98.97	72.43	6.90
DSW-U_{MMSE}	*A*	39.18	61.18	80.19	91.97	96.25	97.76	98.99	77.76	7.91
	B	21.87	41.15	70.35	88.67	95.36	97.71	98.99	69.19	7.99
	Avg.	30.53	51.17	75.27	90.32	95.81	97.74	98.99	73.47	7.94
DSW-(U+Eq)$_{ED}$	*A*	35.29	59.28	**81.79**	**93.71**	**97.17**	**98.28**	99.00	77.59	7.74
	B	18.31	39.45	72.92	**90.83**	**96.19**	**97.97**	99.00	69.28	8.08
	Avg.	26.80	49.37	77.36	**92.27**	**96.68**	**98.13**	99.00	73.44	7.91
DSW-(U+Eq)$_{MMSE}$	*A*	**39.20**	**62.12**	81.42	93.17	96.79	98.16	98.99	**78.48**	**8.63**
	B	**22.17**	**43.83**	**73.73**	90.55	95.84	97.86	98.99	**70.66**	**9.46**
	Avg.	**30.69**	**52.98**	**77.58**	91.86	96.32	98.01	98.99	**74.57**	**9.04**

Table 6.11: Comparison between an ED-based steering vector computation method and our MMSE-based RSG estimation technique in terms of word accuracy (%) when combined with our dual-channel spectral weighting evaluated on the AURORA2-2C-FT (far-talk) database when using clean acoustic models.

		SNR (dB)								
Method	**Test set**	-5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	49.66	76.86	92.68	97.08	98.06	98.57	98.76	85.49	-
	B	27.18	58.76	86.93	95.32	97.53	98.35	98.76	77.35	-
	Avg.	38.42	67.81	89.81	96.20	97.80	98.46	98.76	81.42	-
DSW-B	*A*	60.22	83.49	94.38	97.51	98.25	98.65	**98.79**	88.75	3.26
	B	36.18	67.72	89.10	95.69	97.56	98.46	**98.79**	80.79	3.44
	Avg.	48.20	75.61	91.74	96.60	97.91	98.56	**98.79**	84.77	3.35
DSW-U_{ED}	*A*	55.67	81.87	94.22	97.35	98.23	**98.73**	98.68	87.68	2.19
	B	31.16	64.49	88.46	95.68	97.51	98.44	98.68	79.29	1.94
	Avg.	43.42	73.18	91.34	96.52	97.87	98.59	98.68	83.49	2.07
DSW-U_{MMSE}	*A*	62.05	84.45	94.49	97.35	98.21	**98.73**	98.69	89.21	3.72
	B	38.13	69.56	89.57	95.79	97.64	98.44	98.69	81.52	4.17
	Avg.	50.09	77.01	92.03	96.57	97.93	98.59	98.69	85.37	3.95
DSW-(U+Eq)$_{ED}$	*A*	53.56	81.78	94.64	97.58	**98.43**	98.70	98.63	87.45	1.96
	B	29.46	66.48	90.01	96.07	**97.82**	98.50	98.63	79.72	2.37
	Avg.	41.51	74.13	92.33	96.83	**98.13**	98.60	98.63	83.59	2.17
DSW-(U+Eq)$_{MMSE}$	*A*	**65.11**	**86.19**	**95.00**	**97.59**	98.27	98.70	98.57	**90.14**	**4.65**
	B	**41.10**	**73.44**	**91.16**	**96.27**	97.76	**98.55**	98.57	**83.05**	**5.70**
	Avg.	**53.11**	**79.82**	**93.08**	**96.93**	98.02	**98.63**	98.57	**86.60**	**5.18**

Table 6.12: Comparison between an ED-based steering vector computation method and our MMSE-based RSG estimation technique in terms of word accuracy (%) when combined with our dual-channel spectral weighting evaluated on the AURORA2-2C-FT (far-talk) database when using multi-style acoustic models.

The word accuracy results from the comparison between the ED-based steering vector computation method of [77] and our MMSE-based RSG estimation technique are presented, for close-talk conditions, in Tables 6.9 and 6.10 when using clean and multi-style acoustic models, respectively. Both approaches are evaluated in combination with DSW-(U+Eq), which is our best dual-channel power spectrum enhancement contribution according to the results discussed above. As expected, since the speech component at the secondary channel of a dual-microphone smartphone can be safely neglected in close-talk conditions, both DSW-U_{MMSE} and DSW-U_{ED} do not perform better than DSW-B. However, while our MMSE-based RSG estimation approach barely affects in this case, the ED-based one introduces a harmful mismatch. This is confirmed when adding noise equalization, which also uses the RSG term. Thus, DSW-$(U+Eq)_{MMSE}$ and DSW-$(U+Eq)_{ED}$ perform better and worse, respectively, than DSW-B.

Similarly, Tables 6.11 and 6.12 show, for far-talk position, the results from the comparison between the same ED- and MMSE-based techniques when employing clean and multi-style acoustic models, respectively. In this case, where the speech component at the secondary channel is not negligible, the usefulness of our MMSE-based RSG estimation procedure is confirmed (DSW-$(U+Eq)_{MMSE}$ performs better than DSW-U_{MMSE} which is also better than DSW-B) while the ED-based one still introduces a harmful mismatch due to estimation errors.

6.2.2.2 CHiME-3 results

Tables 6.13 and 6.14 report, for all the methods indicated, the word error rates (WERs) obtained on the CHiME-3 real data evaluation set when using multi-style GMM- and DNN-based acoustic models, respectively. In all cases, WERs are broken down by type of noise. Beamforming methods were also tested by using both the signals from only the five microphones facing forward (5 ch.) and the total six microphones (6 ch.) in the tablet (i.e. by also including the sensor that faces backwards). Similarly to what happened with the AURORA2-2C-CT/FT corpora, considering the secondary sensor for D&S yields a drop in performance while MVDR modestly improves with respect to use only the five sensors facing forward. In this way, our best beamforming choice is an MVDR with all the six microphones in the tablet. This justifies its use to obtain our virtual primary channel for CHiME-3 as aforementioned. Of course, even better results may be obtained using a more sophisticated beamforming technique (e.g. a generalized sidelobe canceller (GSC) [151] or a post-filtered beamformer [127]) for the virtual primary channel. It must be noted that, unlike for the AURORA2-2C-CT/FT corpora, classical beamforming obtains significant improvements over the baseline in this scenario due to the greater number of sensors more separated each other.

	Type of noise					
Method	BUS	CAF	PED	STR	*Average*	*Rel. improv.*
Baseline	49.64	32.72	27.30	21.03	32.67	-
AFE	35.91	16.96	17.84	15.75	21.62	11.05
SMW	33.09	21.39	22.33	18.27	23.77	8.90
Wiener+Int	32.04	14.92	**15.64**	14.01	19.15	13.52
D&S (5 ch.)	32.08	22.60	25.82	15.13	23.91	8.76
D&S (6 ch.)	35.13	25.03	27.50	16.25	25.98	6.69
MVDR (5 ch.)	34.08	17.35	18.16	15.02	21.15	11.52
MVDR (6 ch.)	32.90	16.83	17.40	14.72	20.46	12.21
Lefkimmiatis	42.31	14.79	19.04	17.48	23.41	9.26
MCNR-like	33.60	15.92	17.31	15.14	20.49	12.18
DCSS	34.19	14.90	16.95	15.58	20.41	12.26
P-MVDR	33.18	15.58	16.78	14.76	20.08	12.59
DSW-B	34.51	17.24	17.69	14.85	21.07	11.60
DSW-U$_{MMSE}$	34.42	16.70	17.73	14.70	20.89	11.78
DSW-(U+Eq)$_{MMSE}$	**29.07**	**12.63**	**15.64**	**13.39**	**17.68**	**14.99**

Table 6.13: Word error rate results (in terms of percentage and per type of noise) for our power spectrum enhancement proposals and comparison techniques evaluated with CHiME-3 when multi-style acoustic models are employed. Results are from the real data evaluation set when considering GMMs for acoustic modeling.

	Type of noise					
Method	BUS	CAF	PED	STR	*Average*	*Rel. improv.*
Baseline	51.13	35.06	28.31	21.48	34.00	-
AFE	31.00	14.77	16.69	14.10	19.14	14.86
SMW	31.13	17.99	19.08	17.09	21.32	12.68
Wiener+Int	31.20	13.13	15.88	14.19	18.60	15.40
D&S (5 ch.)	30.90	21.16	25.52	14.61	23.05	10.95
D&S (6 ch.)	33.82	22.81	26.49	15.09	24.55	9.45
MVDR (5 ch.)	29.89	14.66	16.54	14.62	18.93	15.07
MVDR (6 ch.)	29.50	14.79	16.37	13.88	18.64	15.36
Lefkimmiatis	35.02	15.26	17.73	16.29	21.08	12.92
MCNR-like	29.40	14.82	16.95	14.96	19.03	14.97
DCSS	29.52	13.02	15.70	13.99	18.06	15.94
P-MVDR	29.50	13.09	16.12	13.39	18.03	15.97
DSW-B	30.57	13.95	16.52	13.78	18.71	15.29
DSW-U$_{MMSE}$	30.72	15.05	16.91	14.96	19.41	14.59
DSW-(U+Eq)$_{MMSE}$	**26.96**	**11.88**	**15.06**	**13.02**	**16.73**	**17.27**

Table 6.14: Word error rate results (in terms of percentage and per type of noise) for our power spectrum enhancement proposals and comparison techniques evaluated with CHiME-3 when multi-style acoustic models are employed. Results are from the real data evaluation set when considering DNNs for acoustic modeling.

As expected, similar trends are obtained by employing GMMs and DNNs for acoustic modeling. Moreover, the baseline WER from GMM-based acoustic modeling is 1.33% lower than that from DNN-based acoustic modeling as a result of the more sophisticated front-end used in the former case as explained in Subsection 6.1.2. Nevertheless, all the tested techniques perform better by using DNN-based acoustic models than GMM-based ones. As can be seen, the best result is achieved by DSW-(U+Eq)$_{MMSE}$ under DNN-based acoustic modeling, with an absolute WER of 16.73% and a relative average improvement of 17.27% and 1.91% regarding the baseline and MVDR (6 ch.), respectively. Though the secondary channel has already been (successfully) used to define the virtual primary one, these results reveal the convenience of treating the secondary signal in a differentiated manner since it can be further exploited to provide useful information about the acoustic environment. This is confirmed by the results in both absolute and relative terms since, on average and considering DNN-based acoustic models, the percentage change between MVDR (5 ch.) and MVDR (6 ch.) is 0.36% while the percentage change between MVDR (6 ch.) and DSW-(U+Eq)$_{MMSE}$ is 2.35%.

While there are meaningful improvements between DSW-(U+Eq)$_{MMSE}$ and DSW-U$_{MMSE}$, this latter approach slightly enhances the results of DSW-B under GMM-based acoustic modeling (in fact, DSW-U$_{MMSE}$ worsens both DSW-B when using DNN-based acoustic models and MVDR (6 ch.)). This is because MVDR beamforming yields a strong dehomogenization of the noise at the virtual primary and secondary channels. Under this circumstance, the homogeneity assumption underlying DSW-U$_{MMSE}$ is not accomplished, so the substantial improvement only comes when bias correction and noise equalization, which also relies on $\mathcal{A}_{21}(f,t)$, are applied together.

Wiener+Int, which can also be considered a single-channel post-filter and unlike both the multi-channel post-filters Lefkimmiatis and MCNR-like, is able to modestly improve MVDR (6 ch.). Even in this case, DSW-(U+Eq)$_{MMSE}$ yields a relative average improvement of 1.87% under DNN-based acoustic modeling with respect to Wiener+Int as well, which confirms that the secondary microphone can provide more valuable information about the ambient noise than a single-channel noise estimation. Furthermore, unlike for the synthetic AURORA2-2C-CT/FT corpora, it should be noticed that the AFE does not present a competitive performance on the CHiME-3 real data. In addition, DSW-(U+Eq)$_{MMSE}$ is again clearly superior to both DCSS and P-MVDR. Nevertheless, both DCSS and P-MVDR, which again exhibit a similar performance in accordance with the analysis of Subsection 3.2.3, are ranked just behind DSW-(U+Eq)$_{MMSE}$, also outperforming the other post-filters tested.

Unlike for the AURORA2-2C-CT/FT corpora, in CHiME-3 the performance of the

	Type of noise					
Method	BUS	CAF	PED	STR	*Average*	*Rel. improv.*
Baseline	49.64	32.72	27.30	21.03	32.67	-
DSW-B	34.51	17.24	17.69	14.85	21.07	11.60
DSW-U_{ED}	33.95	16.14	17.17	15.13	20.60	12.07
DSW-U_{MMSE}	34.42	16.70	17.73	14.70	20.89	11.78
DSW-$(U+Eq)_{ED}$	29.48	13.02	**14.82**	13.86	17.80	14.87
DSW-$(U+Eq)_{MMSE}$	**29.07**	**12.63**	15.64	**13.39**	**17.68**	**14.99**

Table 6.15: Comparison between an ED-based steering vector computation method and our MMSE-based RSG estimation technique in terms of word error rate (%) when combined with our dual-channel spectral weighting evaluated with CHiME-3 when multi-style acoustic models are employed. Results are from the real data evaluation set when considering GMMs for acoustic modeling.

	Type of noise					
Method	BUS	CAF	PED	STR	*Average*	*Rel. improv.*
Baseline	51.13	35.06	28.31	21.48	34.00	-
DSW-B	30.57	13.95	16.52	13.78	18.71	15.29
DSW-U_{ED}	29.73	13.78	16.27	14.27	18.51	15.49
DSW-U_{MMSE}	30.72	15.05	16.91	14.96	19.41	14.59
DSW-$(U+Eq)_{ED}$	**26.94**	**11.80**	**14.63**	13.69	16.77	17.23
DSW-$(U+Eq)_{MMSE}$	26.96	11.88	15.06	**13.02**	**16.73**	**17.27**

Table 6.16: Comparison between an ED-based steering vector computation method and our MMSE-based RSG estimation technique in terms of word error rate (%) when combined with our dual-channel spectral weighting evaluated with CHiME-3 when multi-style acoustic models are employed. Results are from the real data evaluation set when considering DNNs for acoustic modeling.

MMSE- and ED-based RSG estimation methods is comparable, as can be seen from the WER results shown in Tables 6.15 and 6.16. Nevertheless, it should be borne in mind that for the CHiME-3 real data evaluation it was needed to derive the RSG prior statistics from estimated multi-channel clean speech, which still presents distortions, by using the close-talk microphone data provided with CHiME-3. In addition, the ED-based approach involves an expectation-maximization (EM) iterative procedure while our MMSE-based method is applied straightforward, which makes it a much more efficient approach. Thus, the execution time of both RSG estimation methods, programmed in MatLab, was measured on a quad-core CPU with a clock speed of 3.4 GHz as a function of the utterance duration in seconds. This way it was determined that the time complexity of both methods is linear and the slopes of these curves are 0.008 s/s and 27.208 s/s for the MMSE- and ED-based methods, respectively.

6.2.3 Vector Taylor series feature compensation

Dual-channel vector Taylor series (VTS) feature compensation based on the stacked (S) formulation presented in Chapter 4 is evaluated by considering the two different approaches to compute the clean speech partial estimates of Subsection 4.2.3. These two variants will be referred to as 2-VTS$_a^S$ and 2-VTS$_b^S$, when using the dual-channel MMSE approach of Eq. (4.35) and the straightforward single-channel strategy of Eq. (4.37), respectively. As will be seen, this latter strategy leads to better performance. Because of this, the alternative posterior computation scheme based on modeling the conditional (C) dependence of the noisy secondary channel given the primary one is only tested in combination with the clean speech partial estimates coming from (4.37). We will refer to this scheme as 2-VTS$_b^C$. In addition to these dual-channel VTS techniques, for convenience as well as for comparison purposes, the results of DSW-(U+Eq)$_{MMSE}$ are again shown as this was the dual-channel power spectrum enhancement method exhibiting the best performance in the previous subsection. Furthermore, the AFE [3, 149] results are presented as a reference once again along with those obtained from applying a single-channel VTS feature compensation algorithm [137, 138] on the primary channel. For the single-channel VTS compensation, the two types of clean speech partial estimation described in Subsection 4.2.3 were considered as well. The corresponding experiments are labeled as 1-VTS$_a$ (where $\mathbf{y}_1$ is employed instead of $\mathbf{y}$) and 1-VTS$_b$ [159]. For a fair comparison, in these experiments the required hyperparameters $\boldsymbol{\mu}_{n_1}$ and $\boldsymbol{\Sigma}_{n_1}$ as well as the clean speech GMM were obtained as explained at the end of Subsection 4.2.2 and down below, respectively. Finally, as in the previous subsection, an ASR system employing noisy speech features from the primary channel after mean subtraction is used as baseline.

We must recall that, once again, the primary sensor located at the bottom of the dual-microphone smartphone is identified with the primary channel in the AURORA2-2C-CT/FT corpora. On the other hand, MVDR beamforming is again applied over all the six microphones in the tablet to generate the virtual primary channel for the CHiME-3 database.

The GMM defined in (4.3) to describe the clean speech statistics at the primary channel is comprised of $\mathcal{K} = 256$ multivariate Gaussian components with diagonal covariance matrices. A GMM is computed for the AURORA2-2C-CT/FT databases by performing the EM algorithm on the same dataset as the one used for clean acoustic model training in those databases. Besides this, by also employing the EM algorithm, two additional GMMs are generated for the CHiME-3 corpus: one from the 399 real clean training utterances recorded in the booth and another from the 7138 clean utterances used to define the simulated training dataset. This way, specific GMMs are

available to be used with either real or simulated data to partially overcome the mismatch between both types of data. Furthermore, it must be noticed that the real data GMM is trained from the virtual primary channel computed by means of beamforming from those 399 real clean training utterances.

The parameters of the PDF $p(\mathbf{a}_{21})$, $\boldsymbol{\mu}_{a_{21}}$ and $\boldsymbol{\Sigma}_{a_{21}}$, are *a priori* computed for the AURORA2-2C-CT and AURORA2-2C-FT databases (separately), and the CHiME-3 corpus. Furthermore, in the case of CHiME-3, such parameters are obtained for the simulated and real cases independently. This is again convenient due to the mismatch between the simulated and real data. In this work we assume that $p(\mathbf{a}_{21})$ follows a stationary distribution. In this way, we consider that $\mathbf{a}_{21}$ at every time frame t is a realization of the variable. The mean vector $\boldsymbol{\mu}_{a_{21}}$ and the covariance matrix $\boldsymbol{\Sigma}_{a_{21}}$ are estimated as the sample mean and sample covariance, respectively, from $\mathbf{a}_{21}$ samples. Moreover, we assume independence across frequency bins for $\mathbf{a}_{21}$ and, hence, a diagonal covariance matrix $\boldsymbol{\Sigma}_{a_{21}}$ is used. For all the AURORA2-2C-CT/FT and CHiME-3 corpora we obtain $\mathbf{a}_{21}$ samples from their corresponding development datasets as $\mathbf{a}_{21} = \mathbf{x}_2 - \mathbf{x}_1$ (see Eq. (4.2)).

Finally, as indicated in Table 6.4, we consider that the first and last $M = 20$ frames of every utterance contain only noise energy. This is used to compute the required noise statistics (i.e. $\boldsymbol{\mu}_{n_i}$, $\boldsymbol{\Sigma}_{n_i}$ $(i = 1, 2)$ and $\boldsymbol{\Sigma}_{n_{12}}$) as described in detail in Subsection 4.2.2.

6.2.3.1 AURORA2-2C-CT/FT results

Tables 6.17 and 6.18 show the word accuracy results achieved on the AURORA2-2C-CT database (close-talk) when clean and multi-style acoustic models are used, respectively. Results are averaged across all types of noise in each test set as well as broken down by SNR. Similarly, Tables 6.19 and 6.20 list the results obtained when clean and multi-style acoustic models are employed, respectively, with the AURORA2-2C-FT corpus (far-talk). Again, as expected, due to the minor mismatch between training and test data, the use of multi-style instead of clean acoustic models leads to better ASR accuracy results. Likewise, the results show that the approach for clean speech partial estimate computation that only uses the information from the primary channel, VTS_b, provides better accuracy than VTS_a. Additionally, in all cases, the dual-channel VTS compensation approach (either the stacked version or the alternative 2-VTS^C) outperforms on average the single-channel one. This is expected since the former can exploit the spatial properties of speech and noise signals by means of the relative acoustic path (RAP) vector $\mathbf{a}_{21}$ and the spatial covariance matrix of noise $\boldsymbol{\Sigma}_n$. These parameters are directly involved in the definition of the noisy speech

Method	Test set	SNR (dB) -5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	21.14	38.19	64.60	87.71	95.99	98.13	99.13	67.63	-
	B	15.15	25.50	47.61	77.84	93.44	97.38	99.13	59.49	-
	Avg.	18.15	31.85	56.11	82.78	94.72	97.76	99.13	63.56	-
AFE	*A*	44.31	70.63	87.71	94.93	97.33	98.53	**99.24**	82.24	14.61
	B	27.31	60.29	82.61	92.67	96.58	98.13	**99.24**	76.27	16.78
	Avg.	35.81	65.46	85.16	93.80	96.96	98.33	**99.24**	79.25	15.69
DSW-(U+Eq)$_{MMSE}$	*A*	39.46	67.07	87.28	95.28	97.42	98.44	99.02	80.83	13.20
	B	24.00	49.31	80.22	92.72	97.01	98.19	99.02	73.58	14.09
	Avg.	31.73	58.19	83.75	94.00	97.22	98.32	99.02	77.20	13.64
1-VTS$_a$	*A*	51.95	77.17	91.31	96.04	97.85	98.61	99.09	85.49	17.86
	B	34.17	66.55	86.79	94.42	97.53	98.39	99.09	79.64	20.15
	Avg.	43.06	71.86	89.05	95.23	97.69	98.50	99.09	82.57	19.01
2-VTS$_a^S$	*A*	58.66	81.97	93.25	96.79	98.07	98.64	99.04	87.90	20.27
	B	42.81	74.30	**90.22**	**95.84**	97.99	98.44	99.04	83.27	23.78
	Avg.	50.74	78.14	91.74	96.32	98.03	98.54	99.04	85.59	22.03
1-VTS$_b$	*A*	53.09	77.89	92.24	96.36	97.86	98.59	99.09	86.01	18.38
	B	35.41	67.61	87.13	94.51	97.55	98.38	99.09	80.10	20.61
	Avg.	44.25	72.75	89.69	95.44	97.71	98.49	99.09	83.06	19.50
2-VTS$_b^S$	*A*	59.10	82.52	93.48	97.14	98.14	98.72	99.04	88.18	20.55
	B	43.61	74.14	90.08	95.77	**98.04**	98.49	99.04	83.36	23.87
	Avg.	51.36	78.33	**91.78**	**96.46**	98.09	98.61	99.04	85.77	22.21
2-VTS$_b^C$	*A*	**60.46**	**83.09**	**93.49**	**97.19**	**98.25**	**98.80**	99.09	**88.55**	**20.92**
	B	**45.99**	**75.09**	89.77	95.72	98.03	**98.51**	99.09	**83.85**	**24.36**
	Avg.	**53.23**	**79.09**	91.63	**96.46**	**98.14**	**98.66**	99.09	**86.20**	**22.64**

Table 6.17: Word accuracy results (in terms of percentage and for different SNR values) obtained for our VTS feature compensation proposals and comparison techniques evaluated on the AURORA2-2C-CT (close-talk) database when using clean acoustic models.

PDFs $p(\mathbf{y})$ and $p(\mathbf{y}_2|\mathbf{y}_1)$, determining more accurately the importance of each clean speech partial estimate in the final estimation of (4.4) from either (4.23) or (4.24) in the case of the alternative approach 2-VTS^C.

As can be seen, 2-VTS$_b^C$ is, on average, our best VTS approach under both close- and far-talk conditions as well as by employing either clean or multi-style acoustic models. Moreover, in all cases and on average, 2-VTS$_b^C$ obtains the best performance at the SNRs of -5 dB and 0 dB, making it a suitable approach for challenging low-SNR environments (as those where mobile devices may be used). While 2-VTS^S strongly outperforms 1-VTS, 2-VTS$_b^C$ is clearly superior to 2-VTS$_b^S$ as well. We should recall that a main feature of 2-VTS^S is that the secondary channel is treated in a parallel manner to the primary one, using equivalent distortion models. Nevertheless, we determined that the relation between the secondary noisy observation and the clean speech is more uncertain than that of the primary channel since clean speech is easily masked by noise at the secondary channel. Thus, to decrease the influence of the clean speech variable, 2-VTS^C conditions that distortion model at the secondary channel to the certain noisy observation from the primary channel since both channels are heavily correlated. As a consequence, 2-VTS^C is a more robust estimator than 2-VTS^S, which

Method	Test set	SNR (dB) -5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	47.64	76.99	92.36	96.94	97.98	98.49	98.77	85.07	-
	B	26.22	56.39	85.33	94.52	97.14	98.12	98.77	76.29	-
	Avg.	36.93	66.69	88.85	95.73	97.56	98.31	98.77	80.68	-
AFE	*A*	57.36	82.98	94.13	97.08	98.38	98.82	**99.07**	88.13	3.06
	B	39.06	73.74	90.35	95.99	97.83	98.49	**99.07**	82.58	6.29
	Avg.	48.21	78.36	92.24	96.54	98.11	98.66	**99.07**	85.35	4.67
DSW-(U+Eq)$_{MMSE}$	*A*	65.02	**88.15**	**95.84**	**97.88**	**98.43**	98.60	98.65	**90.65**	**5.58**
	B	42.87	76.50	92.24	**96.62**	98.15	98.58	98.65	84.16	7.87
	Avg.	53.95	82.33	**94.04**	**97.25**	**98.29**	98.59	98.65	87.41	6.73
1-VTS$_a$	*A*	54.83	79.82	93.07	96.42	97.95	98.37	98.84	86.74	1.67
	B	35.74	69.62	89.09	95.51	97.58	98.28	98.84	80.97	4.68
	Avg.	45.29	74.72	91.08	95.97	97.77	98.33	98.84	83.86	3.18
2-VTS$_a^S$	*A*	62.19	85.31	94.81	97.14	98.32	98.54	98.73	89.39	4.32
	B	45.69	77.56	**92.36**	96.57	**98.22**	98.43	98.73	84.81	8.52
	Avg.	53.94	81.44	93.59	96.86	98.27	98.49	98.73	87.10	6.42
1-VTS$_b$	*A*	57.17	80.76	93.00	96.78	98.20	98.58	98.79	87.42	2.35
	B	38.79	71.37	89.43	95.22	97.62	98.39	98.79	81.80	5.51
	Avg.	47.98	76.07	91.22	96.00	97.91	98.49	98.79	84.61	3.93
2-VTS$_b^S$	*A*	63.65	85.44	94.71	97.19	98.40	98.66	98.82	89.68	4.61
	B	47.51	77.93	92.18	96.51	98.13	98.56	98.82	85.14	8.85
	Avg.	55.58	81.69	93.45	96.85	98.27	98.61	98.82	87.41	6.73
2-VTS$_b^C$	*A*	**65.31**	86.07	94.98	97.32	98.42	**98.84**	98.88	90.16	5.09
	B	**49.57**	**78.62**	92.19	96.41	98.09	**98.61**	98.88	**85.58**	**9.29**
	Avg.	**57.44**	**82.35**	93.59	96.87	98.26	**98.73**	98.88	**87.87**	**7.19**

Table 6.18: Word accuracy results (in terms of percentage and for different SNR values) obtained for our VTS feature compensation proposals and comparison techniques evaluated on the AURORA2-2C-CT (close-talk) database when using multi-style acoustic models.

is confirmed by the presented word recognition results.

While there are small differences in terms of WAcc between dual-channel VTS feature compensation and DSW-(U+Eq)$_{MMSE}$ when employing multi-style acoustic models, this is not the case when using clean acoustic models. In the latter scenario, as can be seen from the tables, the performance of the dual-channel VTS feature compensation regarding our best power spectrum enhancement contribution is far greater. This indicates that while both approaches have the ability to reduce the mismatch generated by the acoustic noise, VTS feature compensation is much more able to precisely estimate the actual clean speech features. Anyway, it must be taken into account that both types of approaches work in different domains (i.e. linear power spectral and log-Mel domains), so that they are not incompatible and, therefore, they can be jointly applied in synergy as presented later in this subsection.

6.2.3.2 CHiME-3 results

The WER results achieved on the CHiME-3 database by single-channel VTS feature compensation and the methods selected for comparison are shown in Tables 6.21 and

Method	Test set	SNR (dB) -5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	26.01	43.52	66.77	88.41	96.23	98.18	99.10	69.85	-
	B	16.24	26.54	51.14	81.06	94.43	97.81	99.10	61.20	-
	Avg.	21.13	35.03	58.96	84.74	95.33	98.00	99.10	65.53	-
AFE	*A*	48.36	72.71	87.95	95.39	97.61	98.29	**99.24**	83.39	13.54
	B	30.38	63.40	85.37	93.79	96.99	98.38	**99.24**	78.05	16.85
	Avg.	39.37	68.06	86.66	94.59	97.30	98.34	**99.24**	80.72	15.19
DSW-(U+Eq)$_{MMSE}$	*A*	39.20	62.12	81.42	93.17	96.79	98.16	98.99	78.48	8.63
	B	22.17	43.83	73.73	90.55	95.84	97.86	98.99	70.66	9.46
	Avg.	30.69	52.98	77.58	91.86	96.32	98.01	98.99	74.57	9.04
1-VTS$_a$	*A*	53.81	77.83	91.42	96.62	98.12	98.55	99.09	86.06	16.21
	B	36.47	68.53	87.76	95.09	97.41	98.37	99.09	80.61	19.41
	Avg.	45.14	73.18	89.59	95.86	97.77	98.46	99.09	83.34	17.81
2-VTS$_a^S$	*A*	58.31	80.46	92.74	96.86	98.23	98.64	99.04	87.54	17.69
	B	40.29	72.17	88.53	95.78	97.60	98.46	99.04	82.14	20.94
	Avg.	49.30	76.32	90.64	96.32	97.92	98.55	99.04	84.84	19.31
1-VTS$_b$	*A*	54.97	78.77	91.52	96.71	98.26	98.67	99.09	86.48	16.63
	B	38.12	69.95	88.00	95.34	97.57	98.27	99.09	81.21	20.01
	Avg.	46.55	74.36	89.76	96.03	97.92	98.47	99.09	83.85	18.32
2-VTS$_b^S$	*A*	58.91	81.34	92.86	97.08	98.31	98.70	99.05	87.87	18.02
	B	41.95	72.92	88.84	95.76	97.67	98.34	99.05	82.58	21.38
	Avg.	50.43	77.13	90.85	96.42	97.99	98.52	99.05	85.23	19.70
2-VTS$_b^C$	*A*	**61.25**	**82.66**	**93.46**	**97.32**	**98.49**	**98.81**	99.11	**88.67**	**18.82**
	B	**43.71**	**74.52**	**89.49**	**96.09**	**97.74**	**98.53**	99.11	**83.35**	**22.15**
	Avg.	**52.48**	**78.59**	**91.48**	**96.71**	**98.12**	**98.67**	99.11	**86.01**	**20.48**

Table 6.19: Word accuracy results (in terms of percentage and for different SNR values) obtained for our VTS feature compensation proposals and comparison techniques evaluated on the AURORA2-2C-FT (far-talk) database when using clean acoustic models.

6.22 when considering GMMs and DNNs for acoustic modeling, respectively. These results, obtained from the real data evaluation set when employing multi-style acoustic models, are broken down by type of noise. Once again, as expected, similar trends are achieved by employing GMMs and DNNs for acoustic modeling. Then, it should be reminded that all AFE, DSW-(U+Eq)$_{MMSE}$ and 1-VTS$_b$ are applied using the virtual primary channel generated by means of beamforming, i.e. after MVDR (6 ch.). With this in mind, the first thing that strikes us is that 1-VTS$_b$ considerably deteriorates the results derived from directly using the virtual primary channel without additional processing. In other words, when testing 1-VTS$_b$ under the current conditions, the WER increases 7.74% and 2.83% with respect to MVDR (6 ch.) for GMM- and DNN-based acoustic modeling, respectively.

The poor performance of the VTS feature compensation under multi-style acoustic modeling has already been reported in the literature [53]. In this paper, single-channel VTS feature compensation yields a drop in performance under this circumstance. Unfortunately, while it is evident that VTS feature compensation increases the mismatch in this scenario, a more precise explanation to this fact is not given. In [53], it is also reported that VTS feature compensation, which is intended to precisely estimate the

		SNR (dB)								
Method	**Test set**	-5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	49.66	76.86	92.68	97.08	98.06	98.57	98.76	85.49	-
	B	27.18	58.76	86.93	95.32	97.53	98.35	98.76	77.35	-
	Avg.	38.42	67.81	89.81	96.20	97.80	98.46	98.76	81.42	-
AFE	*A*	59.61	83.33	94.03	97.52	**98.41**	98.69	**99.06**	88.60	3.11
	B	41.66	75.59	91.05	96.18	**98.02**	98.60	**99.06**	83.52	6.17
	Avg.	50.64	79.46	92.54	96.85	**98.22**	**98.65**	**99.06**	86.06	4.64
DSW-(U+Eq)$_{MMSE}$	*A*	65.11	**86.19**	**95.00**	97.59	98.27	**98.70**	98.57	**90.14**	**4.65**
	B	41.10	73.44	91.16	96.27	97.76	98.55	98.57	83.05	5.70
	Avg.	53.11	79.82	93.08	96.93	98.02	98.63	98.57	86.60	5.18
1-VTS$_a$	*A*	56.16	80.18	92.95	97.07	98.21	98.54	98.84	87.19	1.70
	B	39.11	72.01	89.51	95.72	97.74	98.50	98.84	82.10	4.75
	Avg.	47.64	76.10	91.23	96.40	97.98	98.52	98.84	84.65	3.23
2-VTS$_a^S$	*A*	61.44	83.21	93.95	97.37	98.32	98.50	98.78	88.80	3.31
	B	43.66	75.49	90.73	96.35	97.87	98.43	98.78	83.76	6.41
	Avg.	52.55	79.35	92.34	96.86	98.10	98.47	98.78	86.28	4.86
1-VTS$_b$	*A*	58.94	81.92	92.96	97.11	98.22	98.57	98.88	87.95	2.46
	B	42.98	74.38	90.42	95.89	97.83	98.48	98.88	83.33	5.98
	Avg.	50.96	78.15	91.69	96.50	98.03	98.53	98.88	85.64	4.22
2-VTS$_b^S$	*A*	62.85	84.10	93.81	97.40	98.28	98.55	98.82	89.17	3.68
	B	46.48	76.78	91.22	96.22	97.94	98.53	98.82	84.53	7.18
	Avg.	54.67	80.44	92.52	96.81	98.11	98.54	98.82	86.85	5.43
2-VTS$_b^C$	*A*	**65.57**	85.33	94.45	**97.67**	98.40	98.67	98.87	90.02	4.53
	B	**49.47**	**78.23**	**91.96**	**96.48**	97.95	**98.63**	98.87	**85.45**	**8.10**
	Avg.	**57.52**	**81.78**	**93.21**	**97.08**	98.18	**98.65**	98.87	**87.74**	**6.32**

Table 6.20: Word accuracy results (in terms of percentage and for different SNR values) obtained for our VTS feature compensation proposals and comparison techniques evaluated on the AURORA2-2C-FT (far-talk) database when using multi-style acoustic models.

	Type of noise					
Method	BUS	CAF	PED	STR	*Average*	*Rel. improv.*
Baseline	49.64	32.72	27.30	21.03	32.67	-
AFE	35.91	16.96	17.84	15.75	21.62	11.05
MVDR (6 ch.)	32.90	16.83	17.40	14.72	20.46	12.21
DSW-(U+Eq)$_{MMSE}$	**29.07**	**12.63**	**15.64**	**13.39**	**17.68**	**14.99**
1-VTS$_b$	38.12	28.60	26.66	19.42	28.20	4.47

Table 6.21: Word error rate results (in terms of percentage and per type of noise) for VTS feature compensation and comparison techniques evaluated with CHiME-3 when multi-style acoustic models are employed. Results are from the real data evaluation set when considering GMMs for acoustic modeling.

	Type of noise					
Method	BUS	CAF	PED	STR	*Average*	*Rel. improv.*
Baseline	51.13	35.06	28.31	21.48	34.00	-
AFE	31.00	14.77	16.69	14.10	19.14	14.86
MVDR (6 ch.)	29.50	14.79	16.37	13.88	18.64	15.36
DSW-(U+Eq)$_{MMSE}$	**26.96**	**11.88**	**15.06**	**13.02**	**16.73**	**17.27**
1-VTS$_b$	29.44	19.93	20.98	15.54	21.47	12.53

Table 6.22: Word error rate results (in terms of percentage and per type of noise) for VTS feature compensation and comparison techniques evaluated with CHiME-3 when multi-style acoustic models are employed. Results are from the real data evaluation set when considering DNNs for acoustic modeling.

actual clean speech features as aforementioned, improves the baseline when considering clean acoustic models. This is our experience with CHiME-3 as well. In this regard, it is worth to note that the following average WER results are achieved on the real data evaluation set when using clean GMM-based acoustic models: 80.17% as baseline, 46.03% for MVDR (6 ch.) and 40.37% for 1-VTS$_b$. However, 40.37% WER is still too high in comparison with a simple multi-condition training (32.67% WER). In addition, this leads us to the conclusion that multi-condition training is an essential element to be integrated in an ASR system whenever possible in order to provide a good starting point in terms of robustness against noise. Thus, while VTS feature compensation provides improvements on AURORA2-2C-CT/FT when employing multi-style acoustic models, this is not the case with CHiME-3 (medium-vocabulary task and real data). Finally, we must notice that no additional VTS feature compensation results (either dual- or single-channel) have been included in Tables 6.21 and 6.22 since the discussed poor performance is the norm.

In conclusion, our power spectrum enhancement contribution DSW-(U+Eq)$_{MMSE}$ behaving as a post-filter of MVDR beamforming is still the best approach among the evaluated to provide robustness in CHiME-3.

6.2.3.3 Power spectrum enhancement as pre-processing

Here, our power spectrum enhancement contributions and VTS feature compensation are jointly applied in synergy, working both types of approaches in different domains (i.e. power spectral and log-Mel domains). The key idea is as follows: since the higher the SNR of the speech data, the higher the recognition accuracy provided by VTS feature compensation, power spectrum enhancement can be used to increase the starting SNR of the noisy speech. In other words, power spectrum enhancement can be employed as pre-processing before VTS feature compensation to further improve

Method	Test set	SNR (dB) -5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	21.14	38.19	64.60	87.71	95.99	98.13	**99.13**	67.63	-
	B	15.15	25.50	47.61	77.84	93.44	97.38	**99.13**	59.49	-
	Avg.	18.15	31.85	56.11	82.78	94.72	97.76	**99.13**	63.56	-
1-VTS$_b$	*A*	53.09	77.89	92.24	96.36	97.86	98.59	99.09	86.01	18.38
	B	35.41	67.61	87.13	94.51	97.55	98.38	99.09	80.10	20.61
	Avg.	44.25	72.75	89.69	95.44	97.71	98.49	99.09	83.06	19.50
DCSS+(1-VTS$_b$)	*A*	63.24	84.85	94.18	97.21	98.24	98.62	98.78	89.39	21.76
	B	49.26	77.23	90.55	95.87	98.01	98.48	98.78	84.90	25.41
	Avg.	56.25	81.04	92.37	96.54	98.13	98.55	98.78	87.15	23.59
(P-MVDR)+(1-VTS$_b$)	*A*	63.18	84.89	94.18	97.25	98.30	98.68	98.94	89.41	21.78
	B	49.10	77.20	90.58	95.85	98.02	98.53	98.94	84.88	25.39
	Avg.	56.14	81.05	92.38	96.55	98.16	98.61	98.94	87.15	23.59
(DSW-(U+Eq)$_{MMSE}$)+(1-VTS$_b$)	*A*	66.01	85.03	94.38	97.04	98.16	98.74	99.01	89.89	22.26
	B	53.52	78.91	91.33	95.89	97.94	98.45	99.01	86.01	26.52
	Avg.	59.77	81.97	92.86	96.47	98.05	98.60	99.01	87.95	24.39
2-VTS$_b^C$	*A*	60.46	83.09	93.49	97.19	98.25	98.80	99.09	88.55	20.92
	B	45.99	75.09	89.77	95.72	98.03	98.51	99.09	83.85	24.36
	Avg.	53.23	79.09	91.63	96.46	98.14	98.66	99.09	86.20	22.64
DCSS+(2-VTS$_b^C$)	*A*	65.51	86.07	94.68	97.48	98.35	98.81	98.85	90.15	22.52
	B	51.10	78.22	91.48	96.32	98.11	98.54	98.85	85.63	26.14
	Avg.	58.31	82.15	93.08	96.90	98.23	98.68	98.85	87.89	24.33
(P-MVDR)+(2-VTS$_b^C$)	*A*	65.39	86.12	**94.72**	**97.56**	**98.41**	**98.85**	98.96	90.18	22.55
	B	51.29	78.25	91.41	**96.37**	**98.18**	**98.61**	98.96	85.69	26.20
	Avg.	58.34	82.19	93.07	**96.97**	**98.30**	**98.73**	98.96	87.94	24.38
(DSW-(U+Eq)$_{MMSE}$)+(2-VTS$_b^C$)	*A*	**68.93**	**86.91**	94.69	97.26	98.27	98.83	99.00	**90.82**	**23.19**
	B	**56.65**	**80.20**	**91.92**	96.03	98.01	98.43	99.00	**86.87**	**27.38**
	Avg.	**62.79**	**83.56**	**93.31**	96.65	98.14	98.63	99.00	**88.85**	**25.29**

Table 6.23: Word accuracy results (in terms of percentage and for different SNR values) obtained for our power spectrum enhancement proposals as pre-processing of VTS evaluated on the AURORA2-2C-CT (close-talk) database when using clean acoustic models.

the recognizer performance by removing the remaining residual noise after the first enhancement.

Tables 6.23 (under clean acoustic modeling) and 6.24 (under multi-style acoustic modeling) report the word accuracy results obtained on the AURORA2-2C-CT database (close-talk) when DCSS, P-MVDR and DSW-(U+Eq)$_{MMSE}$ are used as pre-processing techniques for 1-VTS$_b$ and 2-VTS$_b^C$. As usual, results are averaged across all types of noise in each test set as well as broken down by SNR. 1-VTS$_b$ and 2-VTS$_b^C$ were chosen for this combined approach as they showed above the best single- and dual-channel VTS feature compensation performance, respectively. Analogously, Tables 6.25 and 6.26 list the corresponding joint results when employing clean and multi-style acoustic models, respectively, on the AURORA2-2C-FT corpus (far-talk). It should be remarked that, on the one hand, the enhanced primary spectrum from DCSS, P-MVDR or DSW-(U+Eq)$_{MMSE}$ is used as input for 1-VTS$_b$. In addition, the input for 2-VTS$_b^C$ consists of this enhanced primary spectrum along with the original noisy spectrum from the secondary channel. This time, results for CHiME-3 are not included since, as it has been previously discussed, VTS feature compensation yields a drop in performance on that task.

Method	Test set	SNR (dB) -5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	47.64	76.99	92.36	96.94	97.98	98.49	98.77	85.07	-
	B	26.22	56.39	85.33	94.52	97.14	98.12	98.77	76.29	-
	Avg.	36.93	66.69	88.85	95.73	97.56	98.31	98.77	80.68	-
1-VTS$_b$	*A*	57.17	80.76	93.00	96.78	98.20	98.58	98.79	87.42	2.35
	B	38.79	71.37	89.43	95.22	97.62	98.39	98.79	81.80	5.51
	Avg.	47.98	76.07	91.22	96.00	97.91	98.49	98.79	84.61	3.93
DCSS+(1-VTS$_b$)	*A*	68.11	87.57	95.36	97.50	98.35	98.64	98.73	90.92	5.85
	B	55.10	80.57	92.73	96.77	98.13	98.44	98.73	86.96	10.67
	Avg.	61.61	84.07	94.05	97.14	98.24	98.54	98.73	88.94	8.26
(P-MVDR)+(1-VTS$_b$)	*A*	68.19	87.47	95.37	97.46	98.34	98.64	98.82	90.91	5.84
	B	54.96	80.83	92.83	96.70	98.21	98.50	98.82	87.01	10.72
	Avg.	61.58	84.15	94.10	97.08	98.28	98.57	98.82	88.96	8.28
(DSW-(U+Eq)$_{MMSE}$)+(1-VTS$_b$)	*A*	70.94	88.16	95.75	97.48	98.28	98.60	98.72	91.54	6.47
	B	58.22	82.48	93.23	96.69	97.98	98.39	98.72	87.83	11.54
	Avg.	64.58	85.32	94.49	97.09	98.13	98.50	98.72	89.69	9.01
2-VTS$_b^C$	*A*	65.31	86.07	94.98	97.32	98.42	**98.84**	**98.88**	90.16	5.09
	B	49.57	78.62	92.19	96.41	98.09	**98.61**	**98.88**	85.58	9.29
	Avg.	57.44	82.35	93.59	96.87	98.26	**98.73**	**98.88**	87.87	7.19
DCSS+(2-VTS$_b^C$)	*A*	71.32	89.06	**96.04**	97.70	98.43	98.67	98.71	91.87	6.80
	B	58.15	82.15	93.42	**97.04**	**98.32**	98.47	98.71	87.93	11.64
	Avg.	64.74	85.61	94.73	97.37	**98.38**	98.57	98.71	89.90	9.22
(P-MVDR)+(2-VTS$_b^C$)	*A*	71.22	88.89	95.79	**97.75**	**98.45**	98.68	98.83	91.80	6.73
	B	57.86	81.97	93.17	97.01	98.27	98.51	98.83	87.80	11.51
	Avg.	64.54	85.43	94.48	**97.38**	98.36	98.60	98.83	89.80	9.12
(DSW-(U+Eq)$_{MMSE}$)+(2-VTS$_b^C$)	*A*	**72.62**	**89.17**	95.94	97.68	98.27	98.60	98.67	**92.05**	**6.98**
	B	**61.22**	**83.42**	**93.83**	96.81	98.12	98.42	98.67	**88.64**	**12.35**
	Avg.	**66.92**	**86.30**	**94.89**	97.25	98.20	98.51	98.67	**90.35**	**9.67**

Table 6.24: Word accuracy results (in terms of percentage and for different SNR values) obtained for our power spectrum enhancement proposals as pre-processing of VTS evaluated on the AURORA2-2C-CT (close-talk) database when using multi-style acoustic models.

While 2-VTS$_b^C$ clearly outperforms 1-VTS$_b$ when they are applied in isolation, differences become smaller when they are combined with DCSS, P-MVDR and DSW-(U+Eq)$_{MMSE}$, as can be observed from the corresponding tables. In this sense, it must be noted that when DCSS, P-MVDR or DSW-(U+Eq)$_{MMSE}$ is combined with 1-VTS$_b$, the same spatial information as in the case of 2-VTS$_b^C$ is being used, since all DCSS, P-MVDR and DSW-(U+Eq)$_{MMSE}$ already exploit the relative acoustic path information (through the term $\mathcal{A}_{21}(f,t)$) and noise spatial correlations. Therefore, it is reasonable to expect small improvements when these are combined with 2-VTS$_b^C$. Nevertheless, as can be seen, 2-VTS$_b^C$ is still better able to improve on average DCSS, P-MVDR and DSW-(U+Eq)$_{MMSE}$ than 1-VTS$_b$, with either clean or multi-style acoustic models under both close- and far-talk conditions. Additionally, while 2-VTS$_b^C$ exhibits a very good performance over the whole SNR range considered (when applied either isolatedly or jointly with DCSS/P-MVDR/DSW-(U+Eq)$_{MMSE}$), it particularly stands out at low SNRs. This is a remarkable result, since, as we know, mobile devices are often used in highly noisy environments such as crowded streets or other public venues.

On average, DSW-(U+Eq)$_{MMSE}$ and P-MVDR are the best pre-processing techniques for 2-VTS$_b^C$ in close- and far-talk conditions, respectively. Indeed, these are the

Method	Test set	SNR (dB) -5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	26.01	43.52	66.77	88.41	96.23	98.18	99.10	69.85	-
	B	16.24	26.54	51.14	81.06	94.43	97.81	99.10	61.20	-
	Avg.	21.13	35.03	58.96	84.74	95.33	98.00	99.10	65.53	-
1-VTS$_b$	*A*	54.97	78.77	91.52	96.71	98.26	98.67	99.09	86.48	16.63
	B	38.12	69.95	88.00	95.34	97.57	98.27	99.09	81.21	20.01
	Avg.	46.55	74.36	89.76	96.03	97.92	98.47	99.09	83.85	18.32
DCSS+(1-VTS$_b$)	*A*	62.36	83.38	92.89	96.86	98.20	98.53	98.79	88.70	18.85
	B	44.11	73.33	89.56	95.60	97.45	98.36	98.79	83.07	21.87
	Avg.	53.24	78.36	91.23	96.23	97.83	98.45	98.79	85.89	20.36
(P-MVDR)+(1-VTS$_b$)	*A*	63.08	83.89	93.57	97.14	98.38	98.65	98.99	89.12	19.27
	B	44.99	75.22	90.27	**96.26**	97.65	98.44	98.99	83.81	22.61
	Avg.	54.04	79.56	91.92	96.70	98.02	98.55	98.99	86.47	20.94
(DSW-(U+Eq)$_{MMSE}$)+(1-VTS$_b$)	*A*	62.64	82.20	92.51	96.77	97.98	98.50	98.94	88.43	18.58
	B	46.77	73.41	89.22	95.81	97.31	98.26	98.94	83.46	22.26
	Avg.	54.71	77.81	90.87	96.29	97.65	98.38	98.94	85.95	20.42
2-VTS$_b^C$	*A*	61.25	82.66	93.46	**97.32**	**98.49**	**98.81**	**99.11**	88.67	18.82
	B	43.71	74.52	89.49	96.09	**97.74**	**98.53**	**99.11**	83.35	22.15
	Avg.	52.48	78.59	91.48	96.71	**98.12**	**98.67**	**99.11**	86.01	20.48
DCSS+(2-VTS$_b^C$)	*A*	62.98	83.67	93.27	96.98	98.04	98.50	98.70	88.91	19.06
	B	44.00	73.31	89.62	95.83	97.42	98.19	98.70	83.06	21.86
	Avg.	53.49	78.49	91.45	96.41	97.73	98.35	98.70	85.99	20.46
(P-MVDR)+(2-VTS$_b^C$)	*A*	**64.66**	**84.74**	**93.92**	97.31	98.34	98.63	98.90	**89.60**	**19.75**
	B	46.97	**76.33**	**90.69**	96.24	97.73	98.44	98.90	**84.40**	**23.20**
	Avg.	55.82	**80.54**	**92.31**	**96.78**	98.04	98.54	98.90	**87.00**	**21.47**
(DSW-(U+Eq)$_{MMSE}$)+(2-VTS$_b^C$)	*A*	64.20	83.40	92.71	96.92	97.97	98.47	98.91	88.95	19.10
	B	**47.98**	73.70	89.46	95.70	97.37	98.25	98.91	83.74	22.54
	Avg.	**56.09**	78.55	91.09	96.31	97.67	98.36	98.91	86.35	20.82

Table 6.25: Word accuracy results (in terms of percentage and for different SNR values) obtained for our power spectrum enhancement proposals as pre-processing of VTS evaluated on the AURORA2-2C-FT (far-talk) database when using clean acoustic models.

best word accuracy results obtained so far on the AURORA2-2C-CT/FT databases thanks to the synergy generated by the combination of two different dual-channel noise-robust ASR approaches.

6.2.4 Deep learning-based techniques

The dual-channel deep learning-based techniques described in Chapter 5 are evaluated on the AURORA2-2C-CT corpus hereunder. While these deep learning-based methods could also be applied to a far-talk scenario, word accuracy results are solely presented for the close-talk case as these methods specifically exploit the power level differences (PLDs) between the two available channels. Hence, the strengths and potential of these deep learning-based techniques can really be appreciated under a close-talk scenario. As it was mentioned in Chapter 5, the quality of the missing-data masks estimated by means of a DNN is tested in terms of word recognition accuracy when employed by a spectral reconstruction technique (namely truncated-Gaussian based imputation, TGI [65]). Similarly, the quality of the noise estimates obtained from using a DNN is evaluated when they are applied to single-channel VTS feature compensation, i.e.

Method	Test set	SNR (dB) -5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	49.66	76.86	92.68	97.08	98.06	98.57	98.76	85.49	-
	B	27.18	58.76	86.93	95.32	97.53	98.35	98.76	77.35	-
	Avg.	38.42	67.81	89.81	96.20	97.80	98.46	98.76	81.42	-
1-VTS$_b$	*A*	58.94	81.92	92.96	97.11	98.22	98.57	**98.88**	87.95	2.46
	B	42.98	74.38	90.42	95.89	97.83	98.48	**98.88**	83.33	5.98
	Avg.	50.96	78.15	91.69	96.50	98.03	98.53	**98.88**	85.64	4.22
DCSS+(1-VTS$_b$)	*A*	68.45	87.18	94.71	97.47	98.24	98.54	98.66	90.77	5.28
	B	52.52	79.83	92.12	96.58	97.71	98.43	98.66	86.20	8.85
	Avg.	60.49	83.51	93.42	97.03	97.98	98.49	98.66	88.49	7.07
(P-MVDR)+(1-VTS$_b$)	*A*	66.76	85.78	94.48	97.59	98.21	98.58	98.74	90.23	4.74
	B	51.09	79.27	91.86	96.50	**97.96**	98.41	98.74	85.85	8.50
	Avg.	58.93	82.53	93.17	97.05	98.09	98.50	98.74	88.04	6.62
(DSW-(U+Eq)$_{MMSE}$)+(1-VTS$_b$)	*A*	67.32	86.32	94.61	97.40	98.06	98.50	98.77	90.37	4.88
	B	52.87	79.16	91.92	96.24	**97.96**	98.34	98.77	86.08	8.73
	Avg.	60.10	82.74	93.27	96.82	98.01	98.42	98.77	88.23	6.81
2-VTS$_b^C$	*A*	65.57	85.33	94.45	**97.67**	**98.40**	**98.67**	98.87	90.02	4.53
	B	49.47	78.23	91.96	96.48	97.95	**98.63**	98.87	85.45	8.10
	Avg.	57.52	81.78	93.21	97.08	**98.18**	**98.65**	98.87	87.74	6.32
DCSS+(2-VTS$_b^C$)	*A*	**69.88**	**87.70**	**95.08**	97.65	98.12	98.47	98.54	**91.15**	**5.66**
	B	53.80	79.98	92.51	96.66	97.89	98.37	98.54	86.54	9.19
	Avg.	**61.84**	83.84	**93.80**	97.16	98.01	98.42	98.54	88.85	7.43
(P-MVDR)+(2-VTS$_b^C$)	*A*	69.77	87.26	94.89	97.62	98.25	98.56	98.65	91.06	5.57
	B	53.49	**80.95**	**92.71**	**96.84**	97.95	98.54	98.65	**86.75**	**9.40**
	Avg.	61.63	**84.11**	**93.80**	**97.23**	98.10	98.55	98.65	**88.91**	**7.49**
(DSW-(U+Eq)$_{MMSE}$)+(2-VTS$_b^C$)	*A*	68.59	87.27	94.62	97.47	98.01	98.47	98.73	90.74	5.25
	B	**54.35**	79.24	92.10	96.24	97.78	98.35	98.73	86.34	8.99
	Avg.	61.47	83.26	93.36	96.86	97.90	98.41	98.73	88.54	7.12

Table 6.26: Word accuracy results (in terms of percentage and for different SNR values) obtained for our power spectrum enhancement proposals as pre-processing of VTS evaluated on the AURORA2-2C-FT (far-talk) database when using multi-style acoustic models.

1-VTS$_b$. To do this, as before, the primary channel in the AURORA2-2C-CT database is identified as the primary microphone of the smartphone, while the sensor at its rear is set as the secondary channel.

For both types of approaches and as a reference, we show once again the results obtained by the ETSI AFE [3]. In addition, the baseline, corresponding to the results obtained when the noisy speech features from the primary channel are employed, is again presented.

6.2.4.1 Missing-data masks for spectral reconstruction

The binary masks estimated by our DNN-based proposal are compared with those calculated by thresholding an estimation of the *a priori* SNR of the primary channel (T-SNR) and then used by the TGI algorithm [65]. The *a priori* SNR for each T-F bin, $\xi_1(f,t)$, is approximated by using the following maximum likelihood (ML) estimator [43]:

$$\hat{\xi}_1(f,t) = \max\left(\frac{|Y_1(f,t)|^2}{|\hat{N}_1(f,t)|^2} - 1, 0\right), \tag{6.4}$$

Method	Test set	SNR (dB) -5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	21.14	38.19	64.60	87.71	95.99	98.13	99.13	67.63	-
	B	15.15	25.50	47.61	77.84	93.44	97.38	99.13	59.49	-
	Avg.	18.15	31.85	56.11	82.78	94.72	97.76	99.13	63.56	-
AFE	*A*	44.31	70.63	87.71	94.93	97.33	98.53	**99.24**	82.24	14.61
	B	27.31	**60.29**	82.61	92.67	96.58	98.13	**99.24**	76.27	16.78
	Avg.	35.81	65.46	85.16	93.80	96.96	98.33	**99.24**	79.25	15.69
TGI+Oracle	*A*	*84.79*	*95.43*	*98.03*	*98.73*	*98.95*	*99.07*	99.13	*95.83*	*28.20*
	B	*74.15*	*91.11*	*96.77*	*98.38*	*98.80*	*99.06*	99.13	*93.05*	*33.56*
	Avg.	*79.47*	*93.27*	*97.40*	*98.56*	*98.88*	*99.07*	99.13	*94.44*	*30.88*
TGI+(T-SNR)	*A*	44.22	65.92	83.96	92.62	96.24	97.75	98.88	80.12	12.49
	B	25.15	54.37	77.11	89.40	94.74	97.14	98.88	72.99	13.50
	Avg.	34.69	60.15	80.54	91.01	95.49	97.45	98.88	76.56	13.00
TGI+DNN	*A*	**54.80**	**79.42**	**92.67**	**97.08**	**98.45**	**98.93**	99.13	**86.89**	**19.26**
	B	**32.29**	60.24	**84.12**	**94.38**	**97.54**	**98.38**	99.13	**77.83**	**18.34**
	Avg.	**43.55**	**69.83**	**88.40**	**95.73**	**98.00**	**98.66**	99.13	**82.36**	**18.80**

Table 6.27: Word accuracy results (in terms of percentage and for different SNR values) obtained for TGI+DNN and comparison techniques evaluated on the AURORA2-2C-CT (close-talk) database when using clean acoustic models.

Method	Test set	SNR (dB) -5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	47.64	76.99	92.36	96.94	97.98	98.49	98.77	85.07	-
	B	26.22	56.39	85.33	94.52	97.14	98.12	98.77	76.29	-
	Avg.	36.93	66.69	88.85	95.73	97.56	98.31	98.77	80.68	-
AFE	*A*	57.36	**82.98**	**94.13**	97.08	98.38	98.82	**99.07**	88.13	3.06
	B	**39.06**	**73.74**	**90.35**	**95.99**	**97.83**	**98.49**	**99.07**	**82.58**	**6.29**
	Avg.	**48.21**	**78.36**	**92.24**	**96.54**	**98.11**	**98.66**	**99.07**	**85.35**	**4.67**
TGI+Oracle	*A*	*84.76*	*95.36*	*98.20*	*98.86*	*99.01*	*99.03*	99.06	*95.87*	*10.80*
	B	*72.66*	*90.59*	*96.66*	*98.40*	*98.86*	*99.06*	99.06	*92.71*	*16.42*
	Avg.	*78.71*	*92.98*	*97.43*	*98.63*	*98.94*	*99.05*	99.06	*94.29*	*13.61*
TGI+(T-SNR)	*A*	49.44	75.57	90.06	95.32	97.15	97.76	98.32	84.22	-0.85
	B	28.75	62.09	84.81	93.56	96.24	97.40	98.32	77.14	0.85
	Avg.	39.10	68.83	87.44	94.44	96.70	97.58	98.32	80.68	0.00
TGI+DNN	*A*	**57.89**	82.18	94.12	**97.50**	**98.46**	**98.85**	98.61	**88.17**	**3.10**
	B	35.75	65.88	87.15	95.02	97.41	98.02	98.61	79.87	3.58
	Avg.	46.82	74.03	90.64	96.26	97.94	98.44	98.61	84.02	3.34

Table 6.28: Word accuracy results (in terms of percentage and for different SNR values) obtained for TGI+DNN and comparison techniques evaluated on the AURORA2-2C-CT (close-talk) database when using multi-style acoustic models.

where $|Y_1(f,t)|^2$ is the filterbank output power spectrum of the noisy speech coming from the primary channel at frequency bin f and time frame t, being $|\hat{N}_1(f,t)|^2$ the corresponding noise power spectrum estimate. As in [65], as well as it has been usual throughout this Thesis, noise estimates are obtained by linear interpolation between the averages of the first and last $M = 20$ frames in the log-Mel domain. Finally, each T-F bin of the mask, $\hat{m}(f,t)$, is calculated as

$$\hat{m}(f,t) = \begin{cases} 1 & \text{if } 10\log_{10}\hat{\xi}_1(f,t) \geq \gamma_m; \\ 0 & \text{otherwise,} \end{cases} \tag{6.5}$$

where $\gamma_m = 0$ dB is the SNR threshold. This value was experimentally chosen by means of the corresponding development dataset.

TGI with oracle masks (TGI+Oracle) is also evaluated as a reference. It must be reminded that oracle masks are obtained by a direct comparison between the clean and noisy utterances using a threshold of $\eta_O = 7$ dB signal-to-noise ratio (SNR). TGI is performed using a 256-component clean speech GMM with diagonal covariance matrices. GMM training is performed by the EM algorithm on the same dataset used for clean acoustic model training (so this GMM is the same as the employed for VTS feature compensation on the AURORA2-2C-CT database). Moreover, it should be recalled that for all the comparison techniques, only the signals from the primary channel were used.

The DNN was trained using 19200 sample pairs of input-target vectors. Training input data consisted of a mixture of samples contaminated with the noises of test set A (i.e. bus, babble, car and pedestrian street) at several SNRs (-5 dB, 0 dB, 5 dB, 10 dB, 15 dB and 20 dB). Noises of test set B are reserved to evaluate the generalization ability of the DNN when exposed to unseen noises during the training phase (namely café, street, bus station and train station). 100 epochs per each RBM were used during the unsupervised pre-training phase, while 1000 epochs were used for the backpropagation algorithm. A learning rate of 0.1 was established and the training dataset was divided into minibatches (small subsets of training data) of 10 samples by following the recommendations in [81]. Preliminary experiments revealed that increasing L (the number of look-forward and look-backward frames) from zero to a few units provides a better performance. Finally, $L = 2$ was chosen (i.e. temporal window size of $2L + 1 = 5$ frames). Thus, the input layer has 230 units or nodes according to Eq. (5.16), both hidden layers have 460 nodes and the output layer has $\mathcal{M} = 23$ nodes. The DNN implementation was carried out using Python along with the library Theano [4].

In Tables 6.27 and 6.28, the word accuracy results achieved on the AURORA2-2C-CT corpus (close-talk) are detailed for the different test sets and techniques evaluated

when employing clean and multi-style acoustic models, respectively. Results, which are averaged across all types of noise in each test set, are broken down by SNR. First of all, we must take into account that TGI+Oracle is an upper reference, that is, the best that TGI may perform (from using ideal missing-data masks). Under clean acoustic modeling, the dual-channel DNN-based system outperforms, on average for all the SNR values but for the clean case, AFE and TGI+(T-SNR). However, under multi-style acoustic modeling, while TGI+DNN is clearly superior to TGI+(T-SNR), the best results are obtained by AFE (excluding TGI+Oracle, of course). In addition, and according to the results for the test set *B*, we can observe that the DNN exhibits some generalization abilities.

As aforementioned, the DNN could exploit temporal correlations by increasing the frame context through the number of look-forward and look-backward frames, L. The relative improvements in terms of word accuracy (average) over $L = 0$ were 1.43%, 3.17% and 3.47% for L values of 1, 2 and 3, respectively, when considering clean acoustic models. As we experimentally checked, the performance tends to saturate for $L = 2$ and greater values. Because of this fact, one can guess that the DNN is mainly exploiting the PLD between the primary and secondary channels. Since most of the information required to provide a PLD-based estimate at frame t is close to that frame, the proposed DNN approach does not benefit of further increasing the length of the analysis window.

Another issue is that the performance of those methods that try to precisely estimate the clean speech features can be severely limited when employing multi-style acoustic models, as it seemed to be the case of VTS feature compensation and now TGI. At this respect, while TGI+(T-SNR) significantly improves the baseline under clean acoustic modeling, that method provides the same average result than a simple multi-condition training when considering multi-style acoustic models. Furthermore, as can be seen, TGI+Oracle performs, in absolute terms and on average, even worse with multi-style than with clean acoustic models. We previously established that multi-condition training is an essential element to be integrated in an ASR system whenever possible in order to provide a good starting point in terms of robustness against noise. On this basis, we can additionally conclude that spectral reconstruction relying on clean speech GMM models and similar approaches are dead ends for noise-robust ASR if a competitive performance under real (i.e. complex) conditions is desired.

6.2.4.2 Noise estimates for feature compensation

Taking into account the speech recognition task as well as the different noise conditions considered in the AURORA2-2C-CT corpus, for the sake of efficiency and to avoid data

		SNR (dB)								
Method	**Test set**	-5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	21.14	38.19	64.60	87.71	95.99	98.13	99.13	67.63	-
	B	15.15	25.50	47.61	77.84	93.44	97.38	99.13	59.49	-
	Avg.	18.15	31.85	56.11	82.78	94.72	97.76	99.13	63.56	-
AFE	*A*	44.31	70.63	87.71	94.93	97.33	98.53	**99.24**	82.24	14.61
	B	27.31	60.29	82.61	92.67	96.58	98.13	**99.24**	76.27	16.78
	Avg.	35.81	65.46	85.16	93.80	96.96	98.33	**99.24**	79.25	15.69
TGI+DNN	*A*	54.80	79.42	92.67	97.08	98.45	98.93	99.13	86.89	19.26
	B	32.29	60.24	84.12	94.38	97.54	98.38	99.13	77.83	18.34
	Avg.	43.55	69.83	88.40	95.73	98.00	98.66	99.13	82.36	18.80
DNN_1	*A*	41.69	67.21	85.56	93.09	96.28	96.92	97.59	80.13	12.50
	B	20.72	45.81	73.57	88.56	94.36	96.50	97.59	69.92	10.43
	Avg.	31.21	56.51	79.57	90.83	95.32	96.71	97.59	75.03	11.47
DNN_2	*A*	**63.02**	**85.74**	**94.53**	**97.72**	**98.59**	**98.94**	99.02	**89.76**	**22.13**
	B	**40.87**	**71.09**	**90.20**	**96.21**	**98.17**	**98.77**	99.02	**82.55**	**23.06**
	Avg.	**51.95**	**78.42**	**92.37**	**96.97**	**98.38**	**98.86**	99.02	**86.16**	**22.60**
DNN_1^{NAT}	*A*	41.93	71.02	90.03	96.62	98.32	98.82	98.83	82.79	15.16
	B	22.70	54.18	82.11	93.61	97.37	98.49	98.83	74.74	15.25
	Avg.	32.32	62.60	86.07	95.12	97.85	98.66	98.83	78.77	15.21
DNN_2^{NAT}	*A*	52.85	78.76	92.06	96.70	98.23	98.84	98.60	86.24	18.61
	B	33.04	63.17	86.74	94.51	97.34	98.40	98.60	78.87	19.38
	Avg.	42.95	70.97	89.40	95.61	97.79	98.62	98.60	82.56	19.00

Table 6.29: Word accuracy results (in terms of percentage and for different SNR values) obtained for our DNN-based noise estimation approaches in combination with VTS feature compensation, and comparison techniques, evaluated on the AURORA2-2C-CT (close-talk) database when using clean acoustic models.

		SNR (dB)								
Method	**Test set**	-5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	47.64	76.99	92.36	96.94	97.98	98.49	98.77	85.07	-
	B	26.22	56.39	85.33	94.52	97.14	98.12	98.77	76.29	-
	Avg.	36.93	66.69	88.85	95.73	97.56	98.31	98.77	80.68	-
AFE	*A*	57.36	82.98	94.13	97.08	98.38	98.82	**99.07**	88.13	3.06
	B	39.06	73.74	90.35	95.99	97.83	98.49	**99.07**	82.58	6.29
	Avg.	48.21	78.36	92.24	96.54	98.11	98.66	**99.07**	85.35	4.67
TGI+DNN	*A*	57.89	82.18	94.12	97.50	98.46	**98.85**	98.61	88.17	3.10
	B	35.75	65.88	87.15	95.02	97.41	98.02	98.61	79.87	3.58
	Avg.	46.82	74.03	90.64	96.26	97.94	98.44	98.61	84.02	3.34
DNN_1	*A*	48.14	72.91	89.29	95.19	97.59	98.16	98.49	83.55	-1.52
	B	24.10	51.93	78.73	91.96	96.04	97.68	98.49	73.41	-2.88
	Avg.	36.12	62.42	84.01	93.58	96.82	97.92	98.49	78.48	-2.20
DNN_2	*A*	**67.05**	**87.95**	**95.39**	**97.93**	**98.60**	98.79	98.99	**90.95**	**5.88**
	B	**44.37**	**75.02**	**91.57**	**96.84**	**98.34**	**98.65**	98.99	**84.13**	**7.84**
	Avg.	**55.71**	**81.49**	**93.48**	**97.39**	**98.47**	**98.72**	98.99	**87.54**	**6.86**
DNN_1^{NAT}	*A*	46.73	75.01	91.91	96.73	98.24	98.63	98.89	84.54	-0.53
	B	26.40	60.24	85.33	94.32	97.15	98.29	98.89	76.96	0.67
	Avg.	36.57	67.63	88.62	95.53	97.70	98.46	98.89	80.75	0.07
DNN_2^{NAT}	*A*	58.21	82.77	93.80	97.44	98.30	98.72	98.74	88.21	3.14
	B	38.17	69.69	88.99	95.36	97.56	98.31	98.74	81.35	5.06
	Avg.	48.19	76.23	91.40	96.40	97.93	98.52	98.74	84.78	4.10

Table 6.30: Word accuracy results (in terms of percentage and for different SNR values) obtained for our DNN-based noise estimation approaches in combination with VTS feature compensation, and comparison techniques, evaluated on the AURORA2-2C-CT (close-talk) database when using multi-style acoustic models.

redundancy, our DNN intended to noise estimation was trained using 25600 pairs of input-target vectors (∼1 second of audio per noisy condition). As for missing-data mask estimation, training input data consisted of a mixture of samples contaminated with the noises of test set A at the SNRs of -5 dB, 0 dB, 5 dB, 10 dB, 15 dB and 20 dB. This way, the noise types of test set B are useful to test the generalization capability of the DNN to unseen noise conditions during training. On the one hand, for the unsupervised pre-training stage the number of epochs in each RBM was 40 and the learning rate was set to 0.0005. On the other hand, for the fine-tuning step the number of epochs was 100 and a learning rate of 0.1 was employed. The momentum rate used was 0.9. Once again, by following the tips from the Hinton's report in [81], the minibatch size was 10 sample pairs. To improve the generalization capability of the network, early-stopping was adopted as a regularization strategy to avoid overfitting during training. Moreover, since the missing-data mask estimation task addressed above is similar to that tackled in this point, and assuming that noise has weak temporal correlations, L was again set to 2. Since $\mathcal{M}$ is set to 23 bins, the input layer has $\dim(\boldsymbol{\mathcal{Y}}(t)) = 230$ (in accordance with Eq. (5.16)) and $\dim(\boldsymbol{\mathcal{Y}}_{NAT}(t)) = 323$ (see Eq. (5.22)) neurons for the DNN without and with NAT (Subsection 5.3.1), respectively. For both DNN configurations the output layer has $\mathcal{M} = 23$ neurons and five hidden layers were set up, according to preliminary recognition experiments, with 512 neurons each. For NAT, $M = 20$ was considered once more. Finally, as for missing-data mask estimation, the implementation of the DNN was done using Python along with the library Theano [4].

Tables 6.29 and 6.30 present a comparison in terms of WAcc between our DNN approach without NAT in combination with 1-VTS$_b$ (DNN$_2$) and other reference techniques when employing clean and multi-style acoustic models, respectively. These techniques are a single-channel DNN-based noise estimator in combination with 1-VTS$_b$ (DNN$_1$) as well as the previous spectral reconstruction approach using DNN-based missing-data masks (TGI+DNN), which shares many similarities with the current approach. It should be noticed that the only difference between DNN$_1$ and DNN$_2$ is that Eq. (5.14) is redefined as $\mathbf{y}(t) = \mathbf{y}_1(t)$ for the former one. Additionally, DNN$_1^{NAT}$ and DNN$_2^{NAT}$ refer to DNN$_1$ and DNN$_2$ when integrating the NAT approach of Subsection 5.3.1, respectively. For both types of acoustic models, the best results are achieved by DNN$_2$, which makes it a better choice than TGI+DNN to provide robustness for ASR in dual-microphone smartphones. Also by a large margin (11.13% and 9.06% on average under clean and multi-style acoustic modeling, respectively) DNN$_2$ is clearly superior to DNN$_1$ as it exploits the information from the secondary channel, which is a good noise reference since, as we know, speech is much attenuated in it. Once again, better WAcc results are generally obtained by employing multi-style instead of

		SNR (dB)								
Method	**Test set**	-5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	21.14	38.19	64.60	87.71	95.99	98.13	**99.13**	67.63	-
	B	15.15	25.50	47.61	77.84	93.44	97.38	**99.13**	59.49	-
	Avg.	18.15	31.85	56.11	82.78	94.72	97.76	**99.13**	63.56	-
Rang	*A*	46.50	69.51	85.78	93.10	95.49	96.48	96.48	81.14	13.51
	B	29.80	59.19	80.82	91.34	95.35	96.31	96.48	75.47	15.98
	Avg.	38.15	64.35	83.30	92.22	95.42	96.40	96.48	78.31	14.75
IMCRA	*A*	44.03	69.43	87.11	94.78	97.53	98.42	99.01	81.88	14.25
	B	26.10	57.33	80.08	91.30	96.44	97.99	99.01	74.87	15.38
	Avg.	35.07	63.38	83.60	93.04	96.99	98.21	99.01	78.38	14.82
MS	*A*	44.45	70.26	87.36	94.68	97.47	98.40	98.90	82.10	14.47
	B	26.57	57.32	80.25	91.27	96.25	98.14	98.90	74.97	15.48
	Avg.	35.51	63.79	83.81	92.98	96.86	98.27	98.90	78.54	14.98
MMSE+NE	*A*	47.56	71.86	88.30	95.02	97.71	98.42	99.08	83.15	15.52
	B	30.17	60.68	82.83	92.28	96.57	98.24	99.08	76.80	17.31
	Avg.	38.87	66.27	85.57	93.65	97.14	98.33	99.08	79.97	16.41
Int	*A*	53.09	77.89	92.24	96.36	97.86	98.59	99.09	86.01	18.38
	B	35.41	67.61	87.13	94.51	97.55	98.38	99.09	80.10	20.61
	Avg.	44.25	72.75	89.69	95.44	97.71	98.49	99.09	83.06	19.50
PLDNE	*A*	46.49	72.90	88.16	94.69	97.17	98.16	98.95	82.93	15.30
	B	34.64	65.74	85.94	93.75	96.97	98.00	98.95	79.17	19.68
	Avg.	40.57	69.32	87.05	94.22	97.07	98.08	98.95	81.05	17.49
DNN_2	*A*	**63.02**	**85.74**	**94.53**	**97.72**	**98.59**	**98.94**	99.02	**89.76**	**22.13**
	B	**40.87**	**71.09**	**90.20**	**96.21**	**98.17**	**98.77**	99.02	**82.55**	**23.06**
	Avg.	**51.95**	**78.42**	**92.37**	**96.97**	**98.38**	**98.86**	99.02	**86.16**	**22.60**

Table 6.31: Word accuracy results (in terms of percentage and for different SNR values) obtained for different noise estimation methods in combination with VTS feature compensation evaluated on the AURORA2-2C-CT (close-talk) database when using clean acoustic models.

clean acoustic models, since the mismatch between training and test data is lower. In addition, test set *B* baseline results are substantially worse than those of test set *A*. Nevertheless, DNN_2 exhibits some generalization capabilities to noise conditions not seen during training.

By taking a look at the results achieved by DNN_1^{NAT}, we can observe that DNN_1 has experienced an average relative improvement of 3.74% and 2.27% in terms of WAcc when employing clean and multi-style acoustic models, respectively. On the other hand, NAT degrades the performance of DNN_2. This could be explained because the secondary channel serves as a better noise reference itself than the information considered in our NAT-based approach, which introduces a greater uncertainty.

To conclude this experimental evaluation, DNN_2, which has exhibited the best performance so far, is compared with different single-channel noise estimation algorithms when applied on the primary channel in combination with 1-VTS_b: Rangachari's algorithm (Rang) [156], improved minima controlled recursive averaging (IMCRA) [28], minimum statistics (MS) [130], MMSE-based noise estimation (MMSE-NE) [72] and linear interpolation in the log-Mel domain (Int) as repeatedly used throughout this Thesis with, once again, $M = 20$ (see Section 2.3 for details). Furthermore, power

		SNR (dB)								
Method	**Test set**	-5	0	5	10	15	20	Clean	*Avg. (-5 to 20)*	*Rel. improv.*
Baseline	*A*	47.64	76.99	92.36	96.94	97.98	98.49	98.77	85.07	-
	B	26.22	56.39	85.33	94.52	97.14	98.12	98.77	76.29	-
	Avg.	36.93	66.69	88.85	95.73	97.56	98.31	98.77	80.68	-
Rang	*A*	56.17	79.03	91.38	96.41	97.59	98.34	98.40	86.49	1.42
	B	35.35	67.48	88.05	94.82	97.28	98.10	98.40	80.18	3.89
	Avg.	45.76	73.26	89.72	95.62	97.44	98.22	98.40	83.34	2.66
IMCRA	*A*	51.33	77.26	91.23	96.32	97.94	98.53	98.88	85.44	0.37
	B	32.23	65.72	86.36	94.32	97.44	98.41	98.88	79.08	2.79
	Avg.	41.78	71.49	88.80	95.32	97.69	98.47	98.88	82.26	1.58
MS	*A*	52.07	77.81	91.35	96.52	98.07	98.62	98.89	85.74	0.67
	B	**47.22**	65.61	86.57	94.46	97.42	98.43	98.89	81.62	5.33
	Avg.	49.65	71.71	88.96	95.49	97.75	98.53	98.89	83.68	3.00
MMSE-NE	*A*	53.63	77.82	90.83	96.25	97.87	98.62	**99.01**	85.84	0.77
	B	36.34	67.88	87.60	94.51	97.28	98.36	**99.01**	80.33	4.04
	Avg.	44.99	72.85	89.22	95.38	97.58	98.49	**99.01**	83.09	2.41
Int	*A*	57.17	80.76	93.00	96.78	98.20	98.58	98.79	87.42	2.35
	B	38.79	71.37	89.43	95.22	97.62	98.39	98.79	81.80	5.51
	Avg.	47.98	76.07	91.22	96.00	97.91	98.49	98.79	84.61	3.93
PLDNE	*A*	55.76	82.30	94.09	96.94	98.25	98.68	98.71	87.67	2.60
	B	40.46	73.10	91.27	96.35	97.85	98.37	98.71	82.90	6.61
	Avg.	48.11	77.70	92.68	96.65	98.05	98.53	98.71	85.29	4.61
DNN_2	*A*	**67.05**	**87.95**	**95.39**	**97.93**	**98.60**	**98.79**	98.99	**90.95**	**5.88**
	B	44.37	**75.02**	**91.57**	**96.84**	**98.34**	**98.65**	98.99	**84.13**	**7.84**
	Avg.	**55.71**	**81.49**	**93.48**	**97.39**	**98.47**	**98.72**	98.99	**87.54**	**6.86**

Table 6.32: Word accuracy results (in terms of percentage and for different SNR values) obtained for different noise estimation methods in combination with VTS feature compensation evaluated on the AURORA2-2C-CT (close-talk) database when using multi-style acoustic models.

level difference noise estimation (PLDNE) [97], which is a dual-channel noise estimation algorithm based on recursive averaging, is also tested in combination with 1-VTS_b. PLDNE is especially interesting since it is intended for dual-microphone smartphones employed in close-talk conditions by assuming both a homogeneous diffuse noise field and that clean speech at the secondary channel is much attenuated with respect to the primary one. The corresponding WAcc results obtained when clean and multi-style acoustic models are used can be seen in Tables 6.31 and 6.32, respectively. As can be observed, on average and for all the SNRs considered but (marginally) the clean case, DNN_2 shows the best performance among the noise estimation algorithms evaluated. In particular, thanks to the powerful regression capabilities of DNNs, DNN_2 is able to achieve a much greater performance than PLDNE with no other assumptions than just exploiting the PLD between the two channels of the device.

6.3 Summary

In this chapter we have evaluated and compared the recognition performance of the contributions presented along this Thesis for noise-robust ASR on IMDs with several sensors. In the first half of the chapter we have introduced the experimental framework, that is, the speech data resources used along with the feature extraction process and the back-end configuration of the ASR system. More precisely, the AURORA2-2C-CT/FT and the CHiME-3 corpora were considered for evaluation as these databases are intended for research on multi-channel noise-robust ASR. While the CHiME-3 is a novel framework that is part of the well-known CHiME challenge series, the AURORA2-2C-CT/FT corpora have been developed in our research group and these were highlighted as another contribution of this Thesis. On the one hand, CHiME-3 concerns the use of a tablet with six microphones in everyday, noisy environments. On the other hand, the AURORA2-2C-CT/FT databases are generated as extensions to the well-known Aurora-2 corpus and emulate the acquisition of noisy speech by means of a dual-microphone smartphone employed in close- and far-talk conditions. These mobile devices (i.e. the tablet and the smartphone) have a rear microphone to better capture the acoustic environment which was equated with the so-called secondary microphone.

Then, the comparative experimental results in terms of word accuracy and/or word error rate were shown. While the primary channel in the AURORA2-2C-CT/FT databases was identified with the primary microphone located at the bottom of the smartphone, a virtual primary channel was obtained for CHiME-3 by means of MVDR beamforming from all the six microphones in the tablet. In other words, since CHiME-3 embeds more than one front sensor, the strategy illustrated in Figure 3.1 was followed in such a manner that our contributions act as beamformer post-filters. Of course, for all the corpora the secondary channel was identified with the aforementioned rear (secondary) microphone. The presentation of the recognition accuracy results was structured bearing in mind the type of noise-robust approach, as in the previous chapters: power spectrum enhancement, VTS feature compensation and deep learning results.

Regarding the power spectrum enhancement experiments, for all the corpora, the best results were obtained under multi-style acoustic modeling when using our unbiased spectral weighting with noise equalization, DSW-(U+Eq)$_{MMSE}$. Our RSG estimation method based on MMSE estimation provides similar or better results with respect to a state-of-the-art technique based on eigenvalue decomposition. As an advantage, our method is a much more efficient way in terms of computational complexity. In addition, it was also shown that the performance of classical beamforming is quite poor when applied to a microphone array comprised of only two sensors very close each other and one of them placed in an acoustic shadow regarding the target (clean speech) signal. In

the case of CHiME-3, classical beamforming obtained significant improvements over the baseline because of the greater number of sensors more separated each other. On the other hand, the secondary sensor for D&S yielded a drop in performance and MVDR modestly improved with respect to use only the five sensors facing forward (similarly to the case of AURORA2-2C-CT/FT). In short, the experimental results revealed the convenience of treating the secondary signal in a differentiated manner to provide useful information about the acoustic environment.

On the other hand, while VTS feature compensation provided major improvements on AURORA2-2C-CT/FT, this was not the case with CHiME-3 (medium-vocabulary task and real data) under multi-style acoustic modeling. A related bad behavior of VTS feature compensation was already reported in the literature [53]. Under such conditions, DSW-(U+Eq)$_{MMSE}$ behaving as a post-filter of MVDR beamforming was finally the best technique among the tested in CHiME-3. Beyond this, for the AURORA2-2C-CT/FT corpora, we showed the substantially better performance of the dual- versus the single-channel VTS feature compensation approach. As mentioned, this was expected since the former is able to exploit additional (spatial) information: the RAP vector $\mathbf{a}_{21}$ and the spatial covariance matrix of noise $\mathbf{\Sigma}_n$. Moreover, a joint scheme was again exploited by employing power spectrum enhancement as pre-processing in synergy with VTS feature compensation to further improve the recognizer performance. In fact, this strategy provided the best results of the chapter for the AURORA2-2C-CT/FT databases.

Thereafter, the results achieved by the dual-channel deep learning-based methods from Chapter 5 were shown. These methods were only evaluated on the AURORA2-2C-CT database (close-talk) as they specifically exploit the power level differences (PLDs) between the two available channels. Missing-data masks and noise log-Mel spectra were estimated for spectral reconstruction and VTS feature compensation, respectively, the accuracy of which was reflected through the obtained recognition results. This could be achieved in a very efficient manner, as well as with no assumptions, by jointly exploiting the dual-channel noisy information and the powerful modeling capabilities of DNNs. Since the secondary channel is a good noise reference, the DNN exhibited, for both tasks, some generalization ability to noise conditions unseen during the training phase. Moreover, because of this same reason, it was shown that the use of noise-aware training (NAT) for DNN-based noise estimation leads to a drop in performance as the information considered by our NAT-based approach introduces uncertainty.

Finally, let us mention two general conclusions derived from the overall results. First, we confirmed that better recognition accuracy results are generally obtained by employing multi-style instead of clean acoustic models in our dual-channel context,

since the mismatch between training and test data is lower. Therefore, we conclude that multi-condition training is also an essential component to be incorporated in a dual-channel ASR system whenever possible in order to provide a good starting point in terms of robustness against noise. In second place, it must be highlighted that our contributions broadly showed an outstanding performance at low SNRs (especially at -5 dB and 0 dB), which makes them promising techniques to be used in highly noisy environments such as those where mobile devices might be used.

CHAPTER 7

Conclusions

In this work we have carried out a study on noise-robust automatic speech recognition (ASR) on multi-channel intelligent mobile devices (IMDs), which has recently become a popular topic. The conclusions drawn from all the work developed in this Thesis are presented in Section 7.1. Finally, Sections 7.2 and 7.3 are devoted to summarize the contributions and future work, respectively.

7.1 Conclusions

A number of conclusions can be drawn from all the work developed in this Thesis. Some of the most relevant are listed down below:

- Intelligent mobile devices (IMDs) such as smartphones or tablets have permeated our society and this is reflected by a sustained growth of sales of IMDs year after year. Due to this fact, ASR has experienced a new upswing, as this technology has begun to be extensively integrated in IMDs to comfortably accomplish many different tasks by means of speech. Since mobile devices can be employed anywhere at any time, tackling with acoustic noise is more important than ever before so that we can ensure a good user experience when running speech recognition-based applications on these devices.

- Due to the decrease in the price of hardware over the recent years, IMDs have begun to embed small microphone arrays (i.e. microphone arrays comprised of a few sensors close each other) in order to mainly perform speech enhancement through noise cancellation. It has been proven both in the literature and by us

that the multi-channel information coming from this type of IMDs can also be exploited for noise-robust ASR purposes, thereby outperforming the single-channel approach. More precisely, we have developed a series of contributions intended to operate in a dual-microphone set-up. This dual-microphone set-up, which can be found in many of the latest IMDs, consists of a primary microphone to capture the voice of the speaker plus a secondary sensor aimed at obtaining information about the acoustic environment. Moreover, the secondary sensor is likely placed in an acoustic shadow regarding the speaker's mouth. As we proved in this work as well as previously reported in the literature [179, 180], classical beamforming exhibits poor performance in this dual-microphone scenario. Therefore, designing *ad-hoc* solutions is mandatory to achieve high recognition accuracy on this kind of IMDs.

- We checked that the statistical distribution of the speech energy is altered in the presence of ambient noise, thereby producing mismatch between the training and testing conditions if the speech recognizer, trained with clean speech data, is employed in noisy environments. This mismatch leads to poor recognition results. A number of single- and multi-channel methods intended to mitigate the effects of noise has been proposed in the literature in order to improve the recognition performance. It has been proven that outstanding ASR results on multi-microphone IMDs can be achieved by combining robust single- and multi-channel algorithms. In this regard, the preferred multi-channel enhancement scheme consists of microphone array processing followed by some kind of post-filtering to overcome the weaknesses of beamforming.

- A couple of noisy speech databases (i.e. the AURORA2-2C-CT/FT corpora) were generated as part of this Thesis to carry out evaluations under a dual-microphone mobile device scenario. To the best of our knowledge, until the appearance of the CHiME-3 corpus throughout year 2015, a database for noise-robust ASR experimental purposes on multi-microphone IMDs was not available.

- Our unbiased dual-channel spectral weighting with noise equalization based on our relative speech gain (RSG) estimation method, DSW-(U+Eq)$_{MMSE}$, showed to leverage multi-condition training especially in comparison with other related enhancement methods. Thus, DSW-(U+Eq)$_{MMSE}$ with multi-style models exhibited quite a competitive performance and consistent results on all the experimental corpora considered in this Thesis. Furthermore, it was proven that our MMSE-based RSG estimation method provides similar or better results regarding a state-of-the-art technique based on eigenvalue decomposition with a

fraction of the computational complexity. Besides this, the experimental results achieved by P-MVDR in comparison with those obtained by MVDR beamforming demonstrated that discarding the phase information is beneficial to overcome the limitations of classical MVDR beamforming when applied on a mobile device with only two microphones very close each other.

- Thanks to exploiting the spatial properties of speech and noise signals through the relative acoustic path (RAP) vector $\mathbf{a}_{21}$ and the spatial covariance matrix of noise $\mathbf{\Sigma}_n$, our dual-channel vector Taylor series (VTS) feature enhancement proposal has shown to be clearly superior to the single-channel VTS approach. More specifically, as a consequence of the stacked formulation, the two-channel joint information is indirectly exploited by means of $\mathbf{a}_{21}$ and $\mathbf{\Sigma}_n$. Then, it was proved to be more robust modeling the conditional dependence of the noisy secondary channel given the primary one in order to explicitly take advantage of the correlations between the two channels. Additionally, for clean speech partial estimate computation, it was experimentally proven that taking into account only the primary instead of the dual-channel information is better, as the secondary signal is usually noisier than the primary one in our dual-microphone set-up.

- The hybrid DNN/signal processing architectures exploit the best features of both the deep learning and signal processing paradigms. They are expected to continue to be successfully investigated in the near future. In this regard, we explored two dual-channel deep learning-based approaches to address the design of two complex stages, from an analytical point of view, of a noise-robust ASR system. DNNs can be trained to efficiently obtain missing-data mask and noise estimates from dual-channel information with good generalization ability by means of exploiting the power level difference (PLD) between the two available channels. In accordance with our experiments, these estimates clearly outperform different analytical techniques when used for missing-data and feature compensation purposes, respectively. Since speech is much attenuated at the secondary sensor of a dual-microphone smartphone employed in close-talk conditions, that sensor seems a very good noise reference. Because of this, the integration of the devised noise-aware training (NAT) strategy introduces greater uncertainty, leading to a drop in performance. While noise estimation algorithms are able to deal with stationary noise very well, the same does not apply to more complex types of noise. In fact, the use of the secondary sensor itself can be understood as a more robust kind of NAT strategy.

- It has also been presented the so-called combinatorial strategy. Thus, in the case of several front sensors in the IMD, a higher quality primary signal can be generated from these front sensors by means of microphone array processing. In addition, a joint scheme has been developed to take advantage of different noise-robust methods performing at different stages (i.e. domains) of the front-end. It has been experimentally shown the effectiveness of such an approach. Indeed, the best CHiME-3 results were obtained by concatenation of beamforming plus a kind of dual-channel post-filtering, namely MVDR+(DSW-(U+Eq)$_{MMSE}$). Apart from this, on the AURORA2-2C-CT/FT corpora, the best accuracy results were achieved by dual-channel feature enhancement followed by dual-channel VTS feature compensation, i.e. (DSW-(U+Eq)$_{MMSE}$)+(2-VTS$_b^C$). These combinations allow us to get outstanding performance at low signal-to-noise ratios (SNRs), which is a promising result as mobile devices are often used in highly noisy environments.

- Multi-condition training is an essential element to be incorporated in a dual-channel ASR system whenever possible in order to provide a good starting point in terms of robustness against noise. Unfortunately, we experienced that the performance of those methods that try to precisely estimate the clean speech features is severely limited when employing multi-style acoustic models, especially in the real-life scenario. Hence, we consider that spectral reconstruction relying on clean speech GMM models and similar approaches can turn out into dead ends for noise-robust ASR if competitive performance under actual (i.e. complex) conditions is desired.

7.2 Contributions

The different technical contributions resulting from our work are summarized in the following:

- Two basic dual-channel power spectrum enhancement techniques devised from the spectral subtraction (SS) and MVDR principles, named DCSS and P-MVDR, respectively [119].

- A dual-channel spectral weighting (DSW) in the power spectral domain including two new ways of estimating the *a priori* SNR in a dual-channel context for Wiener filter (WF) computation as well as a noise equalization procedure [121].

- A complex MMSE-based relative speech gain (RSG) estimation method which can be used to linearly model the channel between two sensors. While this method might be used for several purposes such as the definition of the steering vector for beamforming, it has been applied in combination with the three dual-channel power spectrum enhancement contributions presented in this Thesis.
- A dual-channel VTS feature compensation method based on a stacked formulation [122] along with an alternative and more robust way to compute the posteriors required by that VTS method.
- Two dual-channel DNN-based methods to similarly estimate missing-data masks [120] and noise [123] in the log-Mel domain. These DNNs exploit the power level difference (PLD) between the two available channels to efficiently obtain, in close-talk conditions, accurate missing-data mask and noise estimates with good generalization ability.
- Two different corpora generated as extensions to the well-known Aurora-2 database [148] to experiment with a dual-microphone smartphone used in both close- and far-talk conditions: the AURORA2-2C-CT (Aurora-2 - 2 Channels - Close-Talk) [119] and the AURORA2-2C-FT (Aurora-2 - 2 Channels - Far-Talk) [121] databases.

7.3 Future work

The different contributions presented in this Thesis were devised to take advantage of a dual-microphone set-up consisting of a primary microphone intended to capture the voice of the speaker plus a secondary sensor aimed at getting clearer information on the acoustic environment. Hence, as future work it would be interesting to investigate how the different proposals reported in this dissertation can be extended in order to operate on different portable electronic devices with other small microphone array configurations. In the first instance, the integration of more than one front sensor into the corresponding enhancement framework to replace beamforming would be explored.

On the other hand, it is reminded that the multi-channel information is being only exploited by dual-channel VTS feature compensation during the calculation of the posteriors $P(k|\mathbf{y})$ or $P(k|\mathbf{y}_1, \mathbf{y}_2)$ (depending on the selected approach), which behave as weights in order to combine the clean speech partial estimates. Therefore, we would like to take advantage of the multi-channel noisy observation also for the clean speech partial estimation procedure, since, at the moment, the approach in Eq. (4.37), only based on the primary channel, performs better than the one in (4.35).

With respect to the DNN-based proposals of Chapter 5, an exhaustive search regarding the architecture and training configuration of the DNNs could further improve their performance. For example, to better exploit temporal correlations of the speech signal, the application of recurrent neural networks (RNNs) could be investigated. Also, the use of additional or different kind of features (e.g. pitch-based features to further improve the generalization ability of the DNNs) could be an interesting research topic. Finally, our objective is to extend these methods in order to deal with a hands-free/far-talk scenario. This scenario is more challenging, as the PLD assumptions taken into account are not completely valid since both speech and noise sources might be in far-field conditions.

APPENDIX A

Derivation of the Noise Equalization Weight Vector

THE goal of the noise equalization procedure described in Subsection 3.3.3 is to force that the noise field homogeneity assumption becomes true, namely $\mathcal{S}_{n_1}(f,t) \approx \mathcal{S}_{n_2}(f,t)$. Thus, it was desired to obtain a signal $|\bar{Y}_2(f,t)|^2 = |\bar{X}_2(f,t)|^2 + |\bar{N}_2(f,t)|^2$, where $|\bar{X}_2(f,t)|^2 \approx |X_2(f,t)|^2$ and $|\bar{N}_2(f,t)|^2 \approx |N_1(f,t)|^2$, to be used instead of $|Y_2(f,t)|^2$ for the estimation of the power spectral density (PSD) $\mathcal{S}_{y_2}(f,t)$ required in Eqs. (3.31) and (3.37) for Wiener filter (WF) computation. As established in Subsection 3.3.3, $|\bar{Y}_2(f,t)|^2$ is obtained from the linear combination of the dual-channel noisy observation as in (3.44). Inspired by MVDR beamforming, the required weighting vector to perform this method, $\mathbf{g}_{f,t}$, was calculated as

$$\begin{aligned} &\hat{\mathbf{g}}_{f,t} = \text{argmin}_{\mathbf{g}_{f,t}} \, \mathrm{E}\left[\varepsilon_{f,t}^2\right]; \\ &\text{subject to } \; \mathbf{g}_{f,t}^{\top}\boldsymbol{\alpha}(f,t) = 1, \end{aligned} \tag{A.1}$$

where

$$\varepsilon_{f,t} = \left(|N_1(f,t)|^2 + std\left(|N_1(f,t)|^2\right)\right) - \mathbf{g}_{f,t}^{\top}\bar{\boldsymbol{\nu}}(f,t), \tag{A.2}$$

as well as $\boldsymbol{\alpha}(f,t) = \left(1, \mathcal{A}_{21}^{-1}(f,t)\right)^{\top}$ and $\bar{\boldsymbol{\nu}}(f,t) = (|N_2(f,t)|^2, |N_1(f,t)|^2)^{\top}$.

The minimization problem above is solved by introducing a Lagrange multiplier, λ, in order to incorporate the distortionless constraint, which defines the following

Lagrangian function:

$$\begin{aligned}\mathcal{L}\left(\mathbf{g}_{f,t},\lambda\right) &= \mathrm{E}\left[\left(\left(|N_1(f,t)|^2+std\left(|N_1(f,t)|^2\right)\right)-\mathbf{g}_{f,t}^{\top}\bar{\boldsymbol{\nu}}(f,t)\right)^2\right] \\ &\quad +\lambda\left(\mathbf{g}_{f,t}^{\top}\boldsymbol{\alpha}(f,t)-1\right).\end{aligned} \tag{A.3}$$

The MMSE solution is then obtained by operating $\nabla\mathcal{L}(\mathbf{g}_{f,t},\lambda)=0$ as follows:

$$\begin{cases}\dfrac{\partial\mathcal{L}\left(\mathbf{g}_{f,t},\lambda\right)}{\partial\mathbf{g}_{f,t}} &= -2\mathrm{E}\left[\left(\left(|N_1(f,t)|^2+std\left(|N_1(f,t)|^2\right)\right)-\mathbf{g}_{f,t}^{\top}\bar{\boldsymbol{\nu}}(f,t)\right)\bar{\boldsymbol{\nu}}(f,t)\right]+\lambda\boldsymbol{\alpha}(f,t)=\mathbf{0}_2; \\ \dfrac{\partial\mathcal{L}\left(\mathbf{g}_{f,t},\lambda\right)}{\partial\lambda} &= \mathbf{g}_{f,t}^{\top}\boldsymbol{\alpha}(f,t)-1=0,\end{cases} \tag{A.4}$$

where $\mathbf{0}_2$ is a 2-dimensional zero vector. First of all, we expand the partial derivative $\partial\mathcal{L}\left(\mathbf{g}_{f,t},\lambda\right)/\partial\mathbf{g}_{f,t}=\mathbf{0}_2$ in order to find the weighting vector $\mathbf{g}_{f,t}$:

$$\begin{aligned}\frac{\partial\mathcal{L}\left(\mathbf{g}_{f,t},\lambda\right)}{\partial\mathbf{g}_{f,t}} &= -2\underbrace{\mathrm{E}\left[|N_1(f,t)|^2\bar{\boldsymbol{\nu}}(f,t)\right]}_{\boldsymbol{\phi}_N^{(1)}(f,t)}-2std\left(|N_1(f,t)|^2\right)\underbrace{\mathrm{E}\left[\bar{\boldsymbol{\nu}}(f,t)\right]}_{\boldsymbol{\mu}_N(f,t)} \\ &\quad +2\underbrace{\mathrm{E}\left[\bar{\boldsymbol{\nu}}(f,t)\bar{\boldsymbol{\nu}}^{\top}(f,t)\right]}_{\bar{\boldsymbol{\Phi}}_N(f,t)}\mathbf{g}_{f,t}+\lambda\boldsymbol{\alpha}(f,t)=\mathbf{0}_2.\end{aligned} \tag{A.5}$$

Then, (A.5) can be written in further compact form as,

$$\begin{aligned}\frac{\partial\mathcal{L}\left(\mathbf{g}_{f,t},\lambda\right)}{\partial\mathbf{g}_{f,t}} &= \underbrace{-\boldsymbol{\phi}_N^{(1)}(f,t)-std\left(|N_1(f,t)|^2\right)\boldsymbol{\mu}_N(f,t)}_{-\boldsymbol{\gamma}_N(f,t)}+\bar{\boldsymbol{\Phi}}_N(f,t)\mathbf{g}_{f,t}+\tfrac{1}{2}\lambda\boldsymbol{\alpha}(f,t) \\ &= -\boldsymbol{\gamma}_N(f,t)+\bar{\boldsymbol{\Phi}}_N(f,t)\mathbf{g}_{f,t}+\tfrac{1}{2}\lambda\boldsymbol{\alpha}(f,t)=\mathbf{0}_2.\end{aligned} \tag{A.6}$$

From (A.6), the weighting vector is expressed as,

$$\mathbf{g}_{f,t}=\bar{\boldsymbol{\Phi}}_N^{-1}(f,t)\left(\boldsymbol{\gamma}_N(f,t)-\frac{1}{2}\lambda\boldsymbol{\alpha}(f,t)\right). \tag{A.7}$$

The result in (A.7) is substituted into the partial derivative $\partial\mathcal{L}\left(\mathbf{g}_{f,t},\lambda\right)/\partial\lambda=0$ to find an expression for the Lagrange multiplier λ. First,

$$\frac{\partial\mathcal{L}\left(\mathbf{g}_{f,t},\lambda\right)}{\partial\lambda}=\boldsymbol{\alpha}^{\top}(f,t)\bar{\boldsymbol{\Phi}}_N^{-1}(f,t)\left(\boldsymbol{\gamma}_N(f,t)-\frac{1}{2}\lambda\boldsymbol{\alpha}(f,t)\right)-1=0, \tag{A.8}$$

from which we determine that λ results

$$\lambda=2\frac{\boldsymbol{\alpha}^{\top}(f,t)\bar{\boldsymbol{\Phi}}_N^{-1}(f,t)\boldsymbol{\gamma}_N(f,t)-1}{\boldsymbol{\alpha}^{\top}(f,t)\bar{\boldsymbol{\Phi}}_N^{-1}(f,t)\boldsymbol{\alpha}(f,t)}. \tag{A.9}$$

To conclude, (A.9) is integrated into (A.7) to obtain the final estimate of the noise equalization weighting vector already presented in Subsection 3.3.3:

$$\hat{\mathbf{g}}_{f,t} = \bar{\mathbf{\Phi}}_N^{-1}(f,t)\left[\boldsymbol{\gamma}_N(f,t) - \frac{\boldsymbol{\alpha}^\top(f,t)\bar{\mathbf{\Phi}}_N^{-1}(f,t)\boldsymbol{\gamma}_N(f,t) - 1}{\boldsymbol{\alpha}^\top(f,t)\bar{\mathbf{\Phi}}_N^{-1}(f,t)\boldsymbol{\alpha}(f,t)}\boldsymbol{\alpha}(f,t)\right]. \tag{A.10}$$

APPENDIX B

MMSE Derivation of the Relative Speech Gain Imaginary Part

In Section 3.4, an MMSE-based estimator for the relative speech gain (RSG) in the short-time Fourier transform domain (STFT), $\mathbf{a}_{21}$, was presented. It was stated that the RSG vector could be decomposed into real and imaginary parts as

$$\mathbf{a}_{21} = \mathbf{a}_{21}^r + j\mathbf{a}_{21}^i. \tag{B.1}$$

Additionally, it was assumed statistical independence between the real and imaginary parts $\mathbf{a}_{21}^r$ and $\mathbf{a}_{21}^i$ in such a manner that the final estimation of $\mathbf{a}_{21}$ could be decomposed into the sum of the estimates for the real and imaginary parts independently as $\hat{\mathbf{a}}_{21} = \hat{\mathbf{a}}_{21}^r + j\hat{\mathbf{a}}_{21}^i$. For the sake of clarity, and due to the parallelism between the estimation of the real and imaginary parts of the RSG, only the equations for the obtainment of $\hat{\mathbf{a}}_{21}^r$ where presented in Section 3.4. In the following, the mathematical procedure to get $\hat{\mathbf{a}}_{21}^i$ is described.

As for the case of $\mathbf{a}_{21}^r$ and $\mathbf{y}_2^r$, we also assumed that $\mathbf{a}_{21}^i$ and $\mathbf{y}_2^i$ were Gaussian-distributed, i.e. $\mathbf{a}_{21}^i \sim \mathcal{N}\left(\boldsymbol{\mu}_{A_{21}^i}, \boldsymbol{\Sigma}_{A_{21}^i}\right)$ and $\mathbf{y}_2^i \sim \mathcal{N}\left(\bar{\mathbf{y}}_2^i, \boldsymbol{\Sigma}_{Y_2^i}\right)$. Furthermore, we will assume that $\mathbf{a}_{21}^i$ and $\mathbf{y}_2^i$ are jointly Gaussian and, hence, the conditional probability density function (PDF) $p(\mathbf{a}_{21}^i|\mathbf{y}_2^i)$ is also Gaussian [20]. As a consequence,

$$\begin{aligned}
\hat{\mathbf{a}}_{21}^i &= \mathrm{E}\left[\mathbf{a}_{21}^i|\mathbf{y}_2^i\right] \\
&= \int \mathbf{a}_{21}^i p(\mathbf{a}_{21}^i|\mathbf{y}_2^i) d\mathbf{a}_{21}^i \\
&= \boldsymbol{\mu}_{A_{21}^i} + \boldsymbol{\Sigma}_{A_{21}^i Y_2^i} \boldsymbol{\Sigma}_{Y_2^i}^{-1} \left(\mathbf{y}_2^i - \bar{\mathbf{y}}_2^i\right).
\end{aligned} \tag{B.2}$$

Since it is considered that the parameters of the *a priori* distribution of $\mathbf{a}_{21}^i$, $\boldsymbol{\mu}_{A_{21}^i}$ and $\boldsymbol{\Sigma}_{A_{21}^i}$, are known, we concentrate on the estimation of $\boldsymbol{\Sigma}_{A_{21}^i Y_2^i}$, $\boldsymbol{\Sigma}_{Y_2^i}$ and $\bar{\mathbf{y}}_2^i$ from now onwards.

First, we calculate the mean vector $\bar{\mathbf{y}}_2^i$ and the covariance matrix of the PDF $p(\mathbf{y}_2^i) = \mathcal{N}\left(\bar{\mathbf{y}}_2^i, \boldsymbol{\Sigma}_{Y_2^i}\right)$. From (3.59), $\mathbf{y}_2^i$ is

$$\begin{aligned} \mathbf{y}_2^i &= \mathbf{h}_i\left(\mathbf{a}_{21}^r, \mathbf{a}_{21}^i, \mathbf{n}_1^r, \mathbf{n}_1^i, \mathbf{n}_2^i; \mathbf{y}_1^r, \mathbf{y}_1^i\right) \\ &= \mathbf{a}_{21}^r \odot \left(\mathbf{y}_1^i - \mathbf{n}_1^i\right) + \mathbf{a}_{21}^i \odot \left(\mathbf{y}_1^r - \mathbf{n}_1^r\right) + \mathbf{n}_2^i. \end{aligned} \tag{B.3}$$

Since it was assumed that the variables $\mathbf{n}_k^r$ and $\mathbf{n}_k^i$ $(k = 1, 2)$ follow multivariate Gaussian distributions [44], as well as any linear combination of Gaussian variables follows another Gaussian distribution [150], we similarly linearize the distortion model in (B.3) as a first step before describing $\mathbf{y}_2^i$ by means of a multivariate Gaussian distribution. This is carried out by means of the following first-order vector Taylor series (VTS) expansion of (B.3) around $(\boldsymbol{\mu}_{A_{21}^r}, \boldsymbol{\mu}_{A_{21}^i}, \boldsymbol{\mu}_{N_1^r}, \boldsymbol{\mu}_{N_1^i}, \boldsymbol{\mu}_{N_2^i})$:

$$\begin{aligned} \mathbf{y}_2^i &= \mathbf{h}_i\left(\mathbf{a}_{21}^r, \mathbf{a}_{21}^i, \mathbf{n}_1^r, \mathbf{n}_1^i, \mathbf{n}_2^i; \mathbf{y}_1^r, \mathbf{y}_1^i\right) \\ &\approx \mathbf{h}_i\left(\boldsymbol{\mu}_{A_{21}^r}, \boldsymbol{\mu}_{A_{21}^i}, \boldsymbol{\mu}_{N_1^r}, \boldsymbol{\mu}_{N_1^i}, \boldsymbol{\mu}_{N_2^i}; \mathbf{y}_1^r, \mathbf{y}_1^i\right) + \mathbf{J}_{A_{21}^r}^i\left(\mathbf{a}_{21}^r - \boldsymbol{\mu}_{A_{21}^r}\right) \\ &\quad + \mathbf{J}_{A_{21}^i}^i\left(\mathbf{a}_{21}^i - \boldsymbol{\mu}_{A_{21}^i}\right) + \mathbf{J}_{N_1^r}^i\left(\mathbf{n}_1^r - \boldsymbol{\mu}_{N_1^r}\right) + \mathbf{J}_{N_1^i}^i\left(\mathbf{n}_1^i - \boldsymbol{\mu}_{N_1^i}\right) \\ &\quad + \mathbf{J}_{N_2^i}^i\left(\mathbf{n}_2^i - \boldsymbol{\mu}_{N_2^i}\right), \end{aligned} \tag{B.4}$$

where the $\mathcal{M}\times\mathcal{M}$ Jacobian matrices $\mathbf{J}_{A_{21}^r}^i$, $\mathbf{J}_{A_{21}^i}^i$, $\mathbf{J}_{N_1^r}^i$, $\mathbf{J}_{N_1^i}^i$ and $\mathbf{J}_{N_2^i}^i$ have, respectively, the following definitions,

$$\begin{aligned} \mathbf{J}_{A_{21}^r}^i &= \left.\frac{\partial \mathbf{h}_i}{\partial \mathbf{a}_{21}^r}\right|_{\boldsymbol{\mu}_{N_1^i}} = -\mathbf{J}_{A_{21}^i}^r = \operatorname{diag}\left(\mathbf{y}_1^i - \boldsymbol{\mu}_{N_1^i}\right); \\ \mathbf{J}_{A_{21}^i}^i &= \left.\frac{\partial \mathbf{h}_i}{\partial \mathbf{a}_{21}^i}\right|_{\boldsymbol{\mu}_{N_1^r}} = \mathbf{J}_{A_{21}^r}^r = \operatorname{diag}\left(\mathbf{y}_1^r - \boldsymbol{\mu}_{N_1^r}\right); \\ \mathbf{J}_{N_1^r}^i &= \left.\frac{\partial \mathbf{h}_i}{\partial \mathbf{n}_1^r}\right|_{\boldsymbol{\mu}_{A_{21}^i}} = -\mathbf{J}_{N_1^i}^r = -\operatorname{diag}\left(\boldsymbol{\mu}_{A_{21}^i}\right); \\ \mathbf{J}_{N_1^i}^i &= \left.\frac{\partial \mathbf{h}_i}{\partial \mathbf{n}_1^i}\right|_{\boldsymbol{\mu}_{A_{21}^r}} = \mathbf{J}_{N_1^r}^r = -\operatorname{diag}\left(\boldsymbol{\mu}_{A_{21}^r}\right); \\ \mathbf{J}_{N_2^i}^i &= \frac{\partial \mathbf{h}_i}{\partial \mathbf{n}_2^i} = \mathbf{J}_{N_2^r}^r = \mathbf{I}_{\mathcal{M}}. \end{aligned} \tag{B.5}$$

It is interesting to observe the symmetry between the real and imaginary cases. Then, by considering the linearized distortion model of (B.4), the mean vector of the PDF $p(\mathbf{y}_2^i)$ can be approximated as,

$$\begin{aligned} \bar{\mathbf{y}}_2^i &= \mathrm{E}\left[\mathbf{y}_2^i\right] \\ &\approx \mathbf{h}_i\left(\boldsymbol{\mu}_{A_{21}^r}, \boldsymbol{\mu}_{A_{21}^i}, \boldsymbol{\mu}_{N_1^r}, \boldsymbol{\mu}_{N_1^i}, \boldsymbol{\mu}_{N_2^i}; \mathbf{y}_1^r, \mathbf{y}_1^i\right). \end{aligned} \tag{B.6}$$

This mean can also be considered as a predicted value for $\mathbf{y}_2^i$ obtained from $\boldsymbol{\mu}_{A_{21}^r}$, $\boldsymbol{\mu}_{A_{21}^i}$, $\boldsymbol{\mu}_{N_1^r}$, $\boldsymbol{\mu}_{N_1^i}$, $\boldsymbol{\mu}_{N_2^i}$ and $\mathbf{y}_1$.

The covariance matrix of $p(\mathbf{y}_2^i)$ can be approximated by following a similar procedure from (B.4) and assuming once again statistical independence between $\mathbf{a}_{21}$ and $(\mathbf{n}_1, \mathbf{n}_2)$, as well as between the real and imaginary parts of all the variables involved:

$$\begin{aligned} \boldsymbol{\Sigma}_{Y_2^i} &= \mathrm{E}\left[(\mathbf{y}_2^i - \bar{\mathbf{y}}_2^i)(\mathbf{y}_2^i - \bar{\mathbf{y}}_2^i)^\top\right] \\ &\approx \mathbf{J}_{A_{21}^r}^i \boldsymbol{\Sigma}_{A_{21}^r} \mathbf{J}_{A_{21}^r}^{i\,\top} + \mathbf{J}_{A_{21}^i}^i \boldsymbol{\Sigma}_{A_{21}^i} \mathbf{J}_{A_{21}^i}^{i\,\top} + \mathbf{J}_{N_1^r}^i \boldsymbol{\Sigma}_{N_1^r} \mathbf{J}_{N_1^r}^{i\,\top} \\ &\quad + \mathbf{J}_{N_1^i}^i \boldsymbol{\Sigma}_{N_1^i N_2^i} \mathbf{J}_{N_2^i}^{i\,\top} + \mathbf{J}_{N_1^i}^i \boldsymbol{\Sigma}_{N_1^i} \mathbf{J}_{N_1^i}^{i\,\top} + \mathbf{J}_{N_2^i}^i \boldsymbol{\Sigma}_{N_2^i N_1^i} \mathbf{J}_{N_1^i}^{i\,\top} \\ &\quad + \mathbf{J}_{N_2^i}^i \boldsymbol{\Sigma}_{N_2^i} \mathbf{J}_{N_2^i}^{i\,\top}, \end{aligned} \tag{B.7}$$

with $\boldsymbol{\Sigma}_{N_1^i N_2^i} = \boldsymbol{\Sigma}_{N_2^i N_1^i}^\top = \mathrm{E}\left[(\mathbf{n}_1^i - \boldsymbol{\mu}_{N_1^i})(\mathbf{n}_2^i - \boldsymbol{\mu}_{N_2^i})^\top\right]$.

Finally, taking into account the statistical independence between $\mathbf{a}_{21}$ and $(\mathbf{n}_1, \mathbf{n}_2)$, as well as between $\mathbf{a}_{21}^r$ and $\mathbf{a}_{21}^i$, the covariance matrix $\boldsymbol{\Sigma}_{A_{21}^i Y_2^i}$ can be approximated as,

$$\begin{aligned} \boldsymbol{\Sigma}_{A_{21}^i Y_2^i} &= \mathrm{E}\left[(\mathbf{a}_{21}^i - \boldsymbol{\mu}_{A_{21}^i})(\mathbf{y}_2^i - \bar{\mathbf{y}}_2^i)^\top\right] \\ &\approx \boldsymbol{\Sigma}_{A_{21}^i} \mathbf{J}_{A_{21}^i}^{i\,\top}. \end{aligned} \tag{B.8}$$

APPENDIX C

Resumen

En el presente apéndice se recoge un resumen en castellano de la Memoria de Tesis con el objeto de cumplir con la normativa de elaboración proveniente de la Escuela de Posgrado de la Universidad de Granada. Este resumen se estructura en las siguientes secciones. En primer lugar, las secciones *Introducción*, *Objetivos* y *Estructura de la memoria* se corresponden con el Capítulo 1. A continuación, las secciones *Fundamentos de procesamiento robusto de voz monocanal y multicanal*, *Realce del espectro de potencia multicanal*, *Compensación de características basada en VTS bi-canal*, *Técnicas bi-canal basadas en aprendizaje automático*, *Evaluación experimental* y *Conclusiones y contribuciones* corresponden a los Capítulos 2, 3, 4, 5, 6 y 7, respectivamente.

C.1 Introducción

Los dispositivos móviles inteligentes (DMIs), tales como *smartphones* o tabletas, han revolucionado la manera en que vivimos. Estos dispositivos nos permiten llevar a cabo una gran variedad de tareas que hacen nuestra vida más fácil, como, por ejemplo, comunicarnos con otras personas en cualquier lugar y en cualquier momento o buscar información de forma instantánea. Los DMIs han permeado nuestra sociedad de tal modo que un gran porcentaje de la población alrededor del globo dispone de, al menos, un DMI. Por supuesto, esto tiene su reflejo en un crecimiento sostenido de las ventas de DMIs año tras año. Por ejemplo, la Figura 1.1 muestra, en millones de unidades, el número de *smartphones* vendidos en todo el planeta para el rango comprendido entre los años 2010 y 2015. Como se observa, el incremento de las ventas acontecido a lo largo

de los últimos años ha sido espectacular: de algo menos de 300 millones de *smartphones* vendidos en el año 2010 se ha pasado a cerca de 1500 millones para el año 2015.

Debido a lo anterior junto a la extraordinaria potencia computacional de los más recientes DMIs, el reconocimiento automático del habla (RAH) ha experimentado un nuevo auge. El RAH es hoy día una tecnología madura que ha comenzado a ser integrada de forma extensiva en los DMIs con el fin de ejecutar diferentes tareas, tales como búsqueda por voz, dictado, control por voz y muchas otras. A pesar de esto, los sistemas de RAH se encuentran aún lejos de la precisión de reconocimiento del habla que demuestra el ser humano. A este respecto, considérese una tarea de pequeño vocabulario consistente en el reconocimiento de secuencias de dígitos. En este caso, mientras que el ser humano presenta una tasa de error por debajo del 0.009% [117], algunos de los mejores sistemas de RAH no logran bajar del 0.55% de tasa de error [207]. Por supuesto, la diferencia de rendimiento entre humanos y máquinas crece conforme lo hace la complejidad del vocabulario. Por ejemplo, en un contexto de conversación telefónica, los seres humanos exhiben una tasa de error en torno al 4%, mientras que los sistemas de RAH llegan fácilmente en esta situación a una tasa en torno al 12% [25]. Diferentes factores intervienen en esta diferencia de rendimiento entre humanos y máquinas, estando básicamente relacionados con la introducción de discrepancias entre las condiciones de entrenamiento y evaluación del sistema de RAH (esta cuestión es discutida con más detalle a lo largo del Capítulo 2). Uno de los factores más importantes que contribuyen a la degradación del rendimiento de un sistema de RAH es el ruido acústico. Así, mientras que el ser humano demuestra un alto grado de robustez frente al ruido cuando se trata de reconocer el habla, esta clase de distorsión puede llegar a hacer inutilizables los sistemas de RAH aun cuando estos integren soluciones específicas para enfrentar el ruido acústico [117].

Los dispositivos móviles pueden ser empleados en cualquier lugar y en cualquier momento, por lo que hacer frente a una amplia variedad de entornos ruidosos se convierte en obligatorio con el fin de asegurar una buena experiencia de usuario cuando se usan aplicaciones basadas en RAH en estos dispositivos. En resumen, precisamente por la proliferación de DMIs que integran tecnología de RAH, combatir el ruido acústico es más importante que nunca.

A lo largo de los últimos años, con la intención de realzar la voz ruidosa, los DMIs han comenzado a integrar pequeños *arrays* de sensores, es decir, *arrays* de micrófonos compuestos de pocos sensores próximos entre sí. Por ejemplo, la Figura 1.2 ilustra un *smartphone* que integra dos micrófonos. Aparte del sensor localizado en la parte baja del dispositivo con el fin de estar próximo a la boca del hablante cuando emplea el dispositivo en posición de conversación, un sensor secundario se sitúa en la parte

posterior del *smartphone*. Cuando el usuario habla por teléfono, este último sensor se encuentra orientado hacia su entorno, pudiendo así capturar valiosa información que puede ser usada para llevar a cabo cancelación de ruido de manera sencilla y eficiente.

De acuerdo con las razones discutidas anteriormente, el objetivo principal de esta Tesis es, en consecuencia, el diseño de técnicas que hagan robustos al ruido a los sistemas de RAH que corren sobre DMIs. Más en concreto, aprovecharemos la información multicanal procedente de pequeños *arrays* de sensores embebidos en los más recientes DMIs con el fin de superar el rendimiento de métodos robustos al ruido monocanal. Si bien se podrían emplear técnicas clásicas de procesamiento de *arrays* de micrófonos para este propósito, se ha demostrado en la literatura que su rendimiento se encuentra sustancialmente limitado bajo ciertas configuraciones de pequeños *arrays* de micrófonos [179, 180]. Por tanto, la exploración de nuevas aproximaciones en este contexto parece preferible con el fin de aprovechar de un mejor modo las peculiaridades de los DMIs con varios sensores. Finalmente, nótese que nos centraremos en el tratamiento del ruido emitido por otras fuentes sonoras, normalmente modelado como aditivo, mientras que el tratamiento del ruido de carácter convolutivo debido a reverberación y otros efectos de canal se encuentra fuera del objetivo de la presente Tesis.

C.2 Objetivos

Tal y como hemos introducido, los sistemas de RAH aún sufren de problemas de precisión cuando son desplegados en ambientes ruidosos. Actualmente, este problema es más importante que nunca debido al uso generalizado de aplicaciones basadas en RAH que corren sobre dispositivos móviles, los cuales pueden ser usados en todo momento y lugar. En efecto, hacer frente al ruido acústico se ha convertido en algo imprescindible con el fin de garantizar una buena experiencia de usuario. Puesto que el RAH robusto al ruido es aún hoy día un tema abierto a pesar de todo el progreso llevado a cabo durante las últimas décadas, el objetivo clave de esta Tesis es lograr avances en el mencionado tema a la par que nos enfocamos en un escenario de dispositivo móvil. Dado que muchos DMIs embeben pequeños *arrays* de sensores, querremos aprovechar la información multicanal procedente de ellos con el fin de superar aproximaciones clásicas monocanal de RAH robustas al ruido. Además, también sabemos que el rendimiento de las técnicas clásicas de *beamforming* con pequeños *arrays* de sensores es notablemente limitado [179, 180] y, por tanto, resulta crucial el desarrollo de soluciones específicas que funcionen satisfactoriamente en este escenario. Más específicamente, podemos destacar los siguientes objetivos:

1. Llevar a cabo una revisión de la literatura sobre RAH robusto al ruido monocanal así como sobre aquellos métodos de procesamiento de voz robustos al ruido multicanal especialmente pensados para entornos móviles.

2. Los DMIs incorporan con frecuencia un micrófono (normalmente en su parte posterior) destinado a la captura de información del entorno acústico más que a la obtención de la voz del hablante. Este es el conocido en este trabajo como micrófono secundario. En consecuencia, otro objetivo es el desarrollo de una nueva serie de algoritmos de doble canal que aprovechen la información proporcionada por un micrófono secundario con el fin de mejorar la precisión de RAH en DMIs que son empleados en entornos ruidosos cotidianos.

3. Generar nuevos recursos de voz bajo un marco de trabajo de dispositivo móvil de doble canal con propósitos experimentales.

4. Evaluar nuestros desarrollos y compararlos con otras técnicas del estado del arte con el fin de extraer conclusiones que permitan continuar progresando.

C.3 Estructura de la memoria

Esta Tesis se compone de un total de siete capítulos más tres apéndices, correspondiendo el último de estos apéndices al presente resumen en castellano. Tras la introducción recogida en el Capítulo 1, los fundamentos teóricos de esta Tesis son presentados en el Capítulo 2. A continuación, los Capítulos 3, 4 y 5 están enfocados a describir nuestras contribuciones en RAH multicanal robusto al ruido sobre DMIs. La diferenciación de nuestras contribuciones en tres capítulos independientes se ha hecho a cuenta del tipo de aproximación y del dominio de operación. Finalmente, en los Capítulos 6 y 7 se presentan la evaluación experimental y las conclusiones, respectivamente. Más en concreto:

- En el Capítulo 1 se exponen las tres primeras secciones de este apéndice junto con una breve introducción al problema del reconocimiento automático del habla.

- En el Capítulo 2, se lleva a cabo una revisión de la literatura con el fin de presentar los fundamentos teóricos que justifican nuestros desarrollos posteriores. A su vez, este capítulo se compone de un total de cinco secciones. El modelo de distorsión de la voz que sirve de base para el desarrollo de una variedad de aproximaciones de RAH robusto al ruido y los efectos que produce el ruido acústico sobre la distribución de la energía de la voz se presentan en la primera

sección. Seguidamente son descritos los fundamentos tanto del reconocimiento del habla monocanal robusto al ruido como de la estimación del ruido. Para concluir, aparte de un resumen, se revisan los fundamentos del procesado de la voz multicanal robusto al ruido sobre DMIs. En dicha revisión nos enfocamos en el estudio del *beamforming*, puesto que el procesado de *array* de micrófonos es típicamente empleado junto con técnicas de procesamiento monocanal para proveer de robustez frente al ruido a los sistemas de RAH que corren sobre DMIs con varios sensores. Adicionalmente, se desarrolla el concepto de diferencia de niveles de potencia (PLD, *Power Level Difference*) de doble canal, puesto que éste es un principio conductor de nuestras contribuciones.

- Tres propuestas de realce del espectro de potencia de doble canal son formuladas en el Capítulo 3: DCSS (*Dual-Channel Spectral Subtraction*, es decir, sustracción espectral de doble canal), P-MVDR (*Power-Minimum Variance Distortionless Response*, o respuesta sin distorsión de mínima varianza en potencia) y DSW (*Dual-channel Spectral Weighting*, es decir, pesado espectral de doble canal). DCSS y P-MVDR son métodos básicos de realce sustentados en los principios de sustracción espectral y MVDR, respectivamente. De otro lado, DSW se fundamenta en un filtrado de Wiener e integra un procedimiento de ecualización de ruido también basado en el principio de respuesta sin distorsión de mínima varianza. Todas estas técnicas asumen que el dispositivo móvil dispone únicamente de un sensor frontal (primario), así como de un micrófono en su parte posterior con el fin de capturar mejor la información procedente del entorno acústico. Por ello, se establece una estrategia combinatoria que integra *beamforming* que puede ser tenida en cuenta en el caso de un DMI con más de un sensor frontal, condensando así la información multicanal en sólo dos canales. Finalmente, dado que todas estas propuestas requieren conocer la ganancia de voz relativa entre los dos canales disponibles, también se desarrolla un método de estimación de dicho factor fundamentado en un criterio de minimización de error cuadrático medio aplicado en el dominio de la transformada de Fourier de tiempo reducido.

- Por su parte, en el Capítulo 4 se describe una técnica de compensación (es decir, realce) de características basada en un desarrollo en serie de Taylor vectorial de doble canal. Una vez recordado el modelo de distorsión de voz de doble canal, se expone la formulación del método, la cual se deriva de un esquema de apilamiento. Además de dicho esquema, también se estudia una aproximación alternativa más robusta para el cálculo de las probabilidades *a posteriori* requeridas por el método. En el marco de dicha alternativa, el modelo de distorsión co-

rrespondiente al canal secundario se condiciona a la observación ruidosa y cierta procedente del canal primario.

- El uso de aprendizaje profundo para RAH robusto al ruido sobre DMIs de doble canal se explora en el Capítulo 5. Este capítulo comienza con una breve revisión de aprendizaje profundo aplicado. Si bien esta perspectiva se centra en el procesamiento de la voz, se prefirió su inclusión en este punto de la Tesis dado que también se realiza un breve repaso de arquitecturas de aprendizaje profundo que no presentan una relación directa con los fundamentos del procesamiento de la voz robusto al ruido. A continuación son descritas técnicas de estimación de ruido y máscaras de datos perdidos fundamentadas en el uso de redes neuronales profundas. Estas técnicas se encargan de explotar la información de doble canal en sinergia con las potentes capacidades de modelado de las redes neuronales profundas con el fin de proveer de estimaciones precisas también de un modo eficiente.

- Tras la presentación de las contribuciones anteriores, en el Capítulo 6 se expone la evaluación experimental. Aparte del correspondiente resumen, este capítulo contiene dos secciones más: una sobre el marco experimental y otra referente a los resultados experimentales. En la primera de ellas se describen los recursos de voz multicanal empleados con propósitos experimentales, es decir, las bases de datos AURORA2-2C-CT/FT y CHiME-3. Junto con ellas, también se explican el procedimiento de extracción de características y la configuración del motor de reconocimiento. En particular, debemos destacar los corpus de voz AURORA2-2C-CT/FT como una contribución más de esta Tesis. Las bases de datos AURORA2-2C-CT/FT emulan la adquisición de voz ruidosa en entornos ruidosos cotidianos por medio de un *smartphone* de doble micrófono. Seguidamente, se muestran los resultados experimentales obtenidos tanto por nuestras contribuciones como por los métodos comparativos considerados en términos de precisión de reconocimiento y/o tasa de palabras erróneas. Estos resultados son discutidos y organizados apropiadamente de acuerdo con la estructura de capítulos de esta disertación.

- Finalmente, las conclusiones de esta Tesis se recogen en el Capítulo 7 junto con un resumen de nuestras contribuciones y el trabajo futuro que podría ser llevado a cabo.

C.4 Fundamentos de procesamiento robusto de voz monocanal y multicanal

Tal y como sabemos, el rendimiento de todo sistema de RAH se puede ver sustancialmente degradado cuando existen discrepancias entre las condiciones de entrenamiento y evaluación. Existe una gran cantidad de fuentes de discrepancia, y una de las más relevantes debido a su omnipresencia es el ruido acústico. Dado que proveer de robustez a los sistemas de RAH que corren sobre DMIs es el objetivo principal de esta Tesis, en este capítulo se introducen los fundamentos del procesamiento robusto al ruido en RAH tanto desde una óptica monocanal como multicanal. Ello sirve de presentación de las bases teóricas a partir de las cuales se definen, a lo largo de los próximos capítulos, las diferentes contribuciones de robustez frente al ruido.

En primera instancia se formula el modelo general de distorsión de la voz considerado como el marco de trabajo básico tanto para revisar las aproximaciones de robustez frente al ruido del estado del arte como para desarrollar nuestras contribuciones a lo largo de los siguientes capítulos. Junto con esto, analizamos cómo se modifica la distribución estadística de la energía de la voz en presencia de ruido ambiente.

A continuación se exponen los fundamentos del RAH monocanal robusto al ruido. Esto se lleva a cabo a través de la revisión de algunas de las aproximaciones robustas al ruido más significativas de acuerdo con nuestros propósitos, categorizadas estas en cuatro clases distintas: aproximaciones del espacio de características, aproximaciones basadas en los modelos acústicos del reconocedor, modelado explícito de la distorsión mediante series de Taylor vectoriales (VTS, por sus siglas en inglés) y aproximaciones de datos perdidos. Además, destacamos diferentes ventajas e inconvenientes propias de estas cuatro clases. Así, la selección de la aproximación más adecuada depende en cada caso del escenario de uso del sistema de RAH. Por ejemplo, al contrario que las aproximaciones del espacio de características, los métodos basados en los modelos acústicos del reconocedor se caracterizan por una complejidad computacional relativamente alta. Como ventaja, estos últimos métodos son con frecuencia más robustos frente a distorsiones de la voz que las aproximaciones del espacio de características. También, se observa que la estrategia de VTS, la cual puede emplearse indistintamente para realce de características o adaptación de los modelos del reconocedor, es más precisa con este último propósito que las técnicas clásicas de adaptación de modelos dado que VTS hace uso de un modelo físico que explica cómo es la interacción no lineal entre la señal de voz y las distorsiones ambientales. Respecto a las aproximaciones de datos perdidos, se argumenta que una parte crítica de estas es la estimación de máscaras que identifiquen las regiones espectro-temporales no fiables de la señal de voz ruidosa.

Algunos métodos robustos al ruido, p.ej. filtro de Wiener o realce de características de VTS, requieren de un módulo que estime el ruido acústico de fondo que contamina la señal de voz. En efecto, el rendimiento de esta clase de métodos queda fuertemente supeditado a la precisión de tales estimas de ruido. En consecuencia, a pesar de que a lo largo de los últimos años el foco se ha puesto sobre otras soluciones robustas al ruido que no requieren de una estimación de ruido explícita, brevemente se revisan algunas de las técnicas clásicas más prominentes de estimación de ruido dado que esta cuestión ha sido tradicionalmente muy importante.

La segunda parte de este capítulo se concentra en el procesamiento de voz robusto al ruido multicanal aplicado a DMIs. Afirmamos que el procesamiento de la voz multicanal robusto al ruido ha ganado popularidad a lo largo de los últimos años gracias a su potencial con respecto a las soluciones monocanal así como a la disminución en el precio del hardware. En primer lugar, se proporciona una visión general del RAH multicanal robusto al ruido sobre DMIs. Nos enfocamos en los recientes retos CHiME sobre RAH robusto al ruido sobre una tableta multi-micrófono y observamos que se puede lograr un excelente rendimiento de reconocimiento a partir de combinar algoritmos robustos de tipo monocanal y multicanal. A este respecto, el esquema multicanal preferido consiste en procesamiento de *array* de micrófonos seguido de algún tipo de post-filtrado para solventar las deficiencias del *beamforming*. Por lo tanto, dado que el *beamforming* es un pilar fundamental del RAH multicanal robusto al ruido, algunos de sus fundamentos son presentados junto con los principales campos de ruido. Seguidamente, se comentan los *beamformers* fijos más conocidos, es decir, *delay-and-sum* y MVDR, así como el procesamiento adaptativo de *arrays*. Luego se revisan diferentes post-filtros clásicos de Wiener junto con aproximaciones más recientes específicamente destinadas a RAH multicanal robusto al ruido. Para concluir, se presenta el principio de diferencia de niveles de potencia de doble canal. Como se comprueba, este principio explica las particularidades espaciales de las señales de voz y ruido en un contexto de configuración de micrófono dual. En esencia, se determina que la energía de voz limpia es mayor en el micrófono primario que en el secundario, mientras que las densidades de potencia espectral de ruido observadas por ambos sensores son similares. Finalmente, se señala que este principio será tenido en cuenta a la hora del diseño de nuestras contribuciones destinadas a tal escenario de doble micrófono mientras que el *beamforming* exhibe importantes limitaciones en este contexto.

C.5 Realce del espectro de potencia multicanal

El núcleo de este capítulo es la presentación de tres métodos diferentes para el realce del espectro de potencia, los cuales están destinados a aprovechar la señal de voz ruidosa de doble canal procedente de un dispositivo móvil con el objetivo de mejorar la tasa de reconocimiento de palabras: DCSS, P-MVDR y DSW. Si bien es normal esperar que un dispositivo móvil no posea más de un sensor secundario (es decir, un micrófono cuyo propósito principal es obtener información acerca del entorno acústico dado que no se encuentra orientado hacia el locutor), es probable que tal dispositivo sí integre varios sensores frontales. Bajo este escenario definimos un canal primario virtual a través de la aplicación de *beamforming*, bien a partir únicamente de los sensores frontales o empleando todos los micrófonos del dispositivo (es decir, teniendo en cuenta asimismo el micrófono secundario). Así, este canal primario virtual es usado en conjunción con el canal secundario por los métodos de realce del espectro de potencia de doble canal, de tal manera que estos se comportan como post-filtros. En ello consiste la conocida en esta Tesis como estrategia combinatoria.

Las dos aproximaciones fundamentales DCSS y P-MVDR son presentadas en primer lugar. De un lado, DCSS se ocupa de extender la sustracción espectral a un marco de trabajo de doble canal para superar así el rendimiento de la sustracción espectral monocanal. Adicionalmente, la técnica P-MVDR está basada en *beamforming* MVDR con la salvedad de que la información de fase de la señal se descarta con el fin de solventar algunas de las limitaciones del clásico *beamforming* MVDR cuando es aplicado sobre un dispositivo móvil con pocos micrófonos muy cercanos entre sí. Tanto DCSS como P-MVDR aprovechan las propiedades espaciales de la voz y el ruido mediante un factor de ganancia relativa de voz y términos de correlación espacial de ruido, respectivamente. Seguidamente, a través de un estudio comparativo se determina que P-MVDR es más robusto que DCSS, así como que cuanto menor es la energía de voz limpia en el canal secundario (como por ejemplo en el caso de hacer uso de un *smartphone* de doble micrófono en posición de conversación) más se asemeja el funcionamiento de ambos métodos.

La tercera contribución de realce del espectro de potencia, DSW, consiste en un pesado espectral de doble canal sustentado en filtrado de Wiener. DSW parte de una formulación sencilla por la cual se asume que el micrófono secundario únicamente captura ruido, así como también se presupone la existencia de un campo de ruido homogéneo. Puesto que ambos supuestos no son siempre precisos, este pesado basado en filtrado de Wiener se modifica mediante *1)* la introducción de un término de corrección del sesgo (para rectificar los pesos espectrales resultantes cuando una componente no despreciable de voz está presente en el canal secundario) y *2)* una ecualización del ruido

(inspirada en *beamforming* MVDR) que se aplica sobre el canal secundario antes del cómputo de los pesos espectrales.

Finalmente, puesto que DCSS, P-MVDR y DSW precisan conocer la ganancia de voz relativa entre los micrófonos secundario y primario, con el cometido de obtener el mencionado parámetro, se desarrolla un estimador eficiente (tal y como se demuestra en el capítulo dedicado a la evaluación experimental) de mínimo error cuadrático medio (MMSE, por sus siglas en inglés) al final del capítulo.

C.6 Compensación de características basada en VTS bi-canal

En este capítulo se desarrolla una extensión a un marco de trabajo de doble canal de la compensación (realce) de características de voz basada en VTS. Al igual que para el caso de las técnicas de realce del espectro de potencia de doble canal presentadas en el Capítulo 3, se ha propuesto considerar la estrategia combinatoria basada en *beamforming* cuando este método de compensación de características de VTS bi-canal necesite ser aplicado sobre un dispositivo móvil con más de un micrófono frontal (primario), o más de dos micrófonos de cualquier clase.

El hilo conductor de este método VTS de doble canal es el esquema de formulación apilada. A partir de éste, se desarrolla un estimador MMSE de las características de voz limpia en el dominio log-Mel. A su vez, dicho estimador se sustenta en una expansión VTS de un modelo de distorsión de voz de doble canal. En concreto, a partir de aprovechar la información de doble canal, este método estima las características de voz limpia log-Mel en el canal primario, puesto que se espera que la señal de este canal no se encuentre más afectada por el ruido ambiente que la señal procedente del canal secundario.

Como se observa en el capítulo, este estimador MMSE se formula como una combinación lineal de un conjunto de estimaciones parciales de voz limpia, las cuales son correspondientemente pesadas por otro conjunto de probabilidades *a posteriori*. Se estudian dos aproximaciones diferentes para el cálculo de cada conjunto de parámetros del estimador. En el caso de las probabilidades *a posteriori*, en primer lugar se procede a su derivación a través del esquema de formulación apilada. Como consecuencia de este proceder, la información conjunta de doble canal es aprovechada indirectamente por medio de la matriz de covarianza espacial de ruido y de un término que modela el camino acústico relativo de voz limpia entre los dos canales del dispositivo. A continuación se desarrolla una estrategia más robusta consistente en el modelado de la

dependencia condicional del canal secundario ruidoso dado el primario, aprovechando así explícitamente las correlaciones entre los dos canales (y no de un modo indirecto como en el caso del esquema de formulación apilada). Además, para el cálculo de las estimas parciales de voz limpia, aparte de una aproximación MMSE basada en el esquema de formulación apilada, se explora una estrategia más sencilla y directa. Esta última consiste en únicamente tener en consideración el canal primario en lugar de la información de doble canal dado que la señal secundaria es típicamente más ruidosa que la primaria en nuestro escenario de trabajo.

C.7 Técnicas bi-canal basadas en aprendizaje automático

En este capítulo se explora el uso de aprendizaje profundo aplicado a RAH robusto al ruido sobre dispositivos móviles con varios sensores. En primera instancia, se trata de proporcionar una definición de aprendizaje profundo a la vez que se mencionan sus ventajas cuando se aplica a la resolución de problemas que han sido tradicionalmente abordados desde el paradigma analítico del procesado de señal clásico. Entre estas ventajas, se destacan las enormes capacidades de modelado de las arquitecturas de aprendizaje profundo sin necesidad de llevar a cabo aproximaciones o asunciones referentes al problema subyacente que se trata. Seguidamente se elabora una breve revisión de la literatura con respecto al aprendizaje profundo aplicado. Más en concreto, nos enfocamos en aquellas aproximaciones basadas en aprendizaje automático pensadas tanto para RAH como para realce de la señal de voz, haciendo hincapié en las arquitecturas híbridas de procesado de señal-redes neuronales profundas (DNNs, por sus siglas en inglés). Tales arquitecturas tratan de aprovechar las mejores características de los paradigmas de procesado de señal y aprendizaje profundo para lograr excelentes rendimientos de los sistemas. Esta filosofía, la cual se espera que sea explorada amplia y satisfactoriamente en el corto plazo, es la seguida por nuestras contribuciones basadas en aprendizaje profundo. Dado que tales contribuciones hacen uso de DNNs, sus fundamentos teóricos son explicados. Además, puesto que el algoritmo de retropropagación puede quedar "atrapado" en un mínimo local durante el entrenamiento supervisado de la red debido a la compleja superficie de error derivada del alto número de capas ocultas, se requiere de la apropiada inicialización de los parámetros de la DNN para evitar este indeseable hecho. Así, se argumenta que tal inicialización puede llevarse a cabo por medio de un pre-entrenamiento generativo no supervisado de la DNN a partir de considerar cada par de capas como máquinas de Boltzmann restringidas (RBMs, por sus siglas en inglés). Por tanto, las RBMs son también brevemente des-

critas. Para completar esta revisión teórica, se introducen dos tipos de arquitecturas de aprendizaje profundo cuya popularidad se ha incrementado rápidamente a lo largo de los últimos años entre la comunidad de investigadores en tecnologías del habla: las redes neuronales recurrentes (RNNs, por sus siglas en inglés) y las redes neuronales convolucionales (CNNs, por sus siglas en inglés).

Para concluir el capítulo, se presentan dos contribuciones basadas en aprendizaje profundo de doble canal que abordan el desarrollo de dos tareas complejas (desde un punto de vista analítico) dentro de un sistema de RAH robusto al ruido. Estas tareas son estimación de máscaras de datos perdidos y estimación de ruido, las cuales son enfrentadas a raíz de aprovechar las excelentes capacidades de modelado de las DNNs. Específicamente, estas DNNs explotan la diferencia de niveles de potencia entre los dos canales disponibles (obsérvese la sinergia) para obtener de un modo eficiente las correspondientes estimas con una buena capacidad de generalización a condiciones no vistas durante el entrenamiento de las DNNs. Si bien las estimaciones de máscaras de datos perdidos y las de ruido pueden emplearse de diferentes maneras en un contexto de RAH robusto al ruido, durante la fase experimental de esta Tesis dichas estimaciones son usadas para reconstrucción espectral y compensación de características de voz, respectivamente. Adicionalmente, también se desarrolla una estrategia de entrenamiento con consciencia del ruido con el objetivo de explorar si este método de estimación de ruido de doble canal basado en DNN puede mejorarse a partir de incrementar su conocimiento acerca de la distorsión que contamina aditivamente la voz en cada caso.

C.8 Evaluación experimental

El objetivo de este capítulo es el de evaluar y comparar el rendimiento de las contribuciones presentadas en esta disertación cuando son empleadas para RAH robusto al ruido sobre DMIs con varios sensores. En la primera parte del capítulo se introduce el marco experimental, esto es, los recursos de datos de voz usados junto con el proceso de extracción de características y la configuración del motor de reconocimiento del sistema de RAH. Concretamente, las bases de datos AURORA2-2C-CT/FT y CHiME-3 son consideradas para evaluación dado que dichas bases de datos se encuentran destinadas a investigación en RAH multicanal robusto al ruido. Mientras que CHiME-3 es un marco de trabajo novedoso perteneciente a la famosa serie de retos CHiME, los corpus de voz AURORA2-2C-CT/FT han sido desarrollados en nuestro grupo de investigación y pueden destacarse como otra de las contribuciones de esta Tesis. De un lado, CHiME-3 abarca el uso de una tableta con seis micrófonos en entornos ruidosos cotidianos. De otra parte, las bases de datos AURORA2-2C-CT/FT son generadas como

extensiones del corpus bien conocido Aurora-2 y emulan la adquisición de voz ruidosa por medio de un *smartphone* de doble micrófono empleado en condiciones de habla cercana (posición de conversación) y lejana (el dispositivo se sostiene en una mano a cierta distancia de la cara del locutor). Estos dispositivos móviles (es decir, tanto la tableta como el *smartphone*) tienen un micrófono en su parte posterior para capturar mejor la información procedente del entorno acústico, siendo estos considerados los correspondientes micrófonos secundarios durante la fase experimental.

A continuación se recogen los resultados experimentales comparativos en términos de precisión de reconocimiento y/o tasa de palabras erróneas. Mientras que el canal primario en las bases de datos AURORA2-2C-CT/FT se identifica con el micrófono primario localizado en la parte baja del *smartphone*, en el caso de CHiME-3 se obtiene un canal primario virtual por medio de aplicar *beamforming* sobre los seis micrófonos de la tableta. En otras palabras, puesto que CHiME-3 incorpora más de un sensor frontal, se sigue la estrategia combinatoria ilustrada en la Figura 3.1, de tal manera que nuestras contribuciones actúan como post-filtros del *beamformer*. Por supuesto, en todas las bases de datos consideradas el canal secundario se identifica con el micrófono secundario mencionado anteriormente. La presentación de los resultados de reconocimiento se estructura teniendo presente el tipo de aproximación robusta al ruido, como en los capítulos previos: resultados de realce del espectro de potencia, resultados de compensación de características de VTS y resultados de aprendizaje profundo.

Con respecto a los experimentos de realce del espectro de potencia, para todas las bases de datos, los mejores resultados se obtienen bajo modelado acústico multi-estilo cuando se emplea nuestro pesado espectral de sesgo corregido con ecualización de ruido considerando a su vez nuestro método de estimación MMSE del término de ganancia de voz relativa, DSW-(U+Eq)$_{MMSE}$. Además, se prueba que nuestro método de estimación MMSE de ganancia de voz relativa proporciona similares o mejores resultados de un modo mucho más eficiente en términos de complejidad computacional que una técnica del estado del arte basada en descomposición en valores singulares. También se prueba que el rendimiento de las técnicas clásicas de *beamforming* es notablemente bajo cuando se aplican sobre un *array* de micrófonos compuesto únicamente de dos sensores muy cercanos entre sí y estando uno de ellos localizado en una sombra acústica con respecto a la fuente de señal objetivo. En el caso de CHiME-3, si bien las técnicas de *beamforming* clásicas logran mejoras significativas de rendimiento dado el mayor número de sensores más separados entre sí, considerar el sensor secundario para *delay-and-sum* conlleva un empeoramiento del rendimiento mientras que MVDR mejora modestamente en relación a sólo usar los cinco sensores orientados hacia el locutor (similar a lo ocurrido en AURORA2-2C-CT/FT). En resumen, estos resultados experimentales revelan

la conveniencia de tratar la señal secundaria de una manera diferenciada con el objetivo de proporcionar información útil acerca del entorno acústico.

De otro lado, mientras que la compensación de características de VTS produce importantes mejoras en AURORA2-2C-CT/FT, este no es el caso en CHiME-3 (tarea de complejidad media y datos reales) bajo modelado acústico multi-estilo. De hecho, un pobre rendimiento de la compensación de características de VTS en similares condiciones ya fue documentado en la literatura [53]. Bajo tales condiciones, DSW-(U+Eq)$_{MMSE}$ comportándose como post-filtro del *beamforming* MVDR es finalmente la mejor técnica de entre las evaluadas en este capítulo para proveer de robustez frente al ruido en CHiME-3. Más allá de esto, para las bases de datos AURORA2-2C-CT/FT, mostramos la superioridad de la aproximación de compensación de características de VTS de doble canal frente a la monocanal. Tal y como se menciona, este hecho era esperado, dado que la estrategia bi-canal es capaz de aprovechar información adicional de carácter espacial: el vector que modela el camino acústico relativo de voz $\mathbf{a}_{21}$ y la matriz de covarianza espacial de ruido $\mathbf{\Sigma}_n$. Además, se hace uso de nuevo de la estrategia combinatoria a partir de emplear el realce del espectro de potencia como pre-procesamiento en sinergia con la compensación de características de VTS para así mejorar aún más el rendimiento del reconocedor del habla. De hecho, esta estrategia proporciona los mejores resultados de todo el capítulo para las bases de datos AURORA2-2C-CT/FT.

Después, se muestran los resultados obtenidos por los métodos basados en aprendizaje profundo de doble canal definidos en el Capítulo 5. Estas técnicas son evaluadas únicamente sobre el corpus AURORA2-2C-CT (*smartphone* en posición conversacional) puesto que están destinadas a explotar específicamente la diferencia de niveles de potencia entre los dos canales disponibles. Así, de acuerdo con los resultados, se determina que de forma precisa se estiman máscaras de datos perdidos y ruido en el dominio log-Mel para reconstrucción espectral y compensación de características de voz de VTS, respectivamente. Ello puede ser logrado de un modo eficiente así como sin realizar ninguna clase de asunción a partir de explotar conjuntamente la información ruidosa de doble canal y las enormes capacidades de modelado de las DNNs. Puesto que el canal secundario es una buena referencia del ruido acústico existente, la DNN exhibe, para ambas tareas de estimación, una significativa capacidad de generalización a condiciones de ruido no vistas durante la fase de entrenamiento. Adicionalmente, debido a esta misma razón, se demuestra que el uso de una estrategia de entrenamiento con consciencia del ruido en el contexto del estimador de ruido basado en DNN lleva a una degradación del rendimiento del método dado que la información considerada por dicha estrategia introduce incertidumbre.

Finalmente, nos ocupamos de mencionar dos conclusiones generales derivadas del conjunto de resultados. En primer lugar, confirmamos que, en general, se obtienen mejores resultados de reconocimiento cuando se emplean modelos acústicos multi-estilo en lugar de modelos entrenados con voz limpia, ya que la discrepancia entre los datos de evaluación y entrenamiento es menor en el primero de los casos. En consecuencia, concluimos que el entrenamiento multi-condición es un componente esencial que incorporar en un sistema de RAH multicanal siempre que sea posible con el fin de proporcionar un buen punto de partida en términos de robustez frente a distorsiones. En segundo lugar, debe destacarse que nuestras contribuciones muestran por lo general un rendimiento muy notable a bajas SNRs (especialmente a -5 dB y 0 dB), hecho que las hace muy apropiadas para ser usadas en entornos altamente ruidosos, como aquellos en los que los dispositivos móviles son susceptibles de usarse, p. ej. calles abarrotadas u otros lugares públicos.

C.9 Conclusiones y contribuciones

Diferentes conclusiones pueden ser extraídas a partir de todo el trabajo desarrollado en esta Tesis y algunas de las más importantes se listan a continuación:

- Los dispositivos móviles inteligentes (DMIs), tales como *smartphones* o tabletas, han permeado nuestra sociedad y ello, como no podía ser de otra forma, se ve reflejado en un crecimiento sostenido de las ventas de DMIs año tras año. Debido a este hecho, el RAH ha experimentado un nuevo auge, dado que esta tecnología ha comenzado a integrarse de un modo extensivo en los DMIs para llevar a cabo cómodamente diversas tareas por medio de la voz. Puesto que los dispositivos móviles pueden emplearse en cualquier momento y lugar, el tratamiento del ruido acústico es más importante que nunca con el objetivo de asegurar una buena experiencia de usuario cuando se usan aplicaciones basadas en RAH en estos dispositivos.

- Debido a la disminución del precio del hardware, los DMIs han comenzado a integrar pequeños *arrays* de sensores a lo largo de los últimos años con el fin principal de llevar a cabo realce de la voz mediante cancelación de ruido. Tanto en la literatura como en esta Tesis se demuestra que la información multicanal procedente de esta clase de DMIs puede ser también aprovechada con propósitos de RAH robusto al ruido, mejorando el rendimiento de la aproximación monocanal. Más en concreto, hemos desarrollado una serie de contribuciones destinadas a operar sobre una configuración de doble micrófono. Dicha configuración, la cual puede

encontrarse en muchos de los últimos DMIs, consiste en un micrófono primario para capturar la voz del locutor más uno secundario pensado para obtener información sobre el entorno acústico. Además, este sensor secundario se emplaza típicamente en una sombra acústica con respecto al locutor. Como también se prueba en este trabajo aparte de haber sido previamente documentado en la literatura [179, 180], las técnicas clásicas de *beamforming* exhiben un pobre rendimiento en este escenario de micrófono dual. Por tanto, el diseño de soluciones *ad-hoc* resulta crucial con el fin de lograr una alta precisión de reconocimiento en esta clase de DMIs.

- Hemos comprobado que la distribución estadística de la energía de voz se ve modificada en presencia de ruido ambiente, apareciendo así discrepancias entre las condiciones de entrenamiento y evaluación si el reconocedor del habla, entrenado con datos de voz limpia, es empleado en entornos ruidosos. Estas discrepancias conducen a pobres resultados de reconocimiento. Entonces, revisamos diversos métodos monocanal y multicanal destinados a mitigar los efectos del ruido con el fin de mejorar el rendimiento del reconocedor. Se ha probado que combinar algoritmos robustos monocanal y multicanal es la manera de obtener resultados sobresalientes de RAH en DMIs multi-micrófono. A este respecto, el esquema de realce multicanal preferido consiste en procesado de *array* de micrófonos seguido de alguna clase de post-filtrado para solventar las debilidades del *beamforming*.

- Dos bases de datos de voz ruidosa (es decir, AURORA2-2C-CT/FT) fueron generadas como parte de esta Tesis para experimentar bajo un escenario de dispositivo móvil de doble micrófono. Por lo que sabemos, hasta la aparición del corpus CHiME-3 a lo largo del año 2015, no había disponible una base de datos con propósitos de experimentación en RAH robusto al ruido sobre DMIs multi-micrófono.

- Nuestro pesado espectral de doble canal no sesgado con ecualización de ruido, el cual también considera nuestro método de estimación de la ganancia relativa de voz, DSW-(U+Eq)$_{MMSE}$, demostró potenciar un entrenamiento multi-condición, especialmente en comparación con otros métodos de realce similares. Así, DSW-(U+Eq)$_{MMSE}$ con modelos multi-estilo evidenció un rendimiento altamente competitivo y resultados consistentes sobre todas las bases de datos experimentales empleadas en esta Tesis. Además, se probó que nuestro método de estimación MMSE de la ganancia relativa de voz proporciona resultados similares o mejores, de un modo mucho más eficiente en términos de complejidad computacional, con respecto a una técnica del estado del arte basada en descomposición de valores

singulares. Aparte de esto, los resultados experimentales logrados por P-MVDR en comparación con aquellos obtenidos por un *beamforming* MVDR demostraron que descartar la información de fase es beneficioso para superar las limitaciones del *beamforming* MVDR clásico cuando se aplica sobre un dispositivo móvil con sólo dos micrófonos muy cercanos entre sí.

- Gracias al aprovechamiento de las propiedades espaciales de las señales de voz y ruido a través del vector de camino acústico relativo $\mathbf{a}_{21}$ y de la matriz de covarianza espacial de ruido $\mathbf{\Sigma}_n$, nuestra propuesta de realce de características VTS bi-canal ha demostrado claramente ser superior a la correspondiente aproximación VTS monocanal. Más en concreto, como consecuencia del esquema de formulación apilada, la información conjunta de dos canales es explotada indirectamente por medio de $\mathbf{a}_{21}$ y $\mathbf{\Sigma}_n$. Entonces se mostró que resulta más robusto modelar la dependencia condicional del canal ruidoso secundario dado el primario con el fin de aprovechar explícitamente las correlaciones entre los dos canales. Adicionalmente, para el cálculo de las estimas parciales de voz limpia se probó experimentalmente que únicamente tener en consideración el canal primario en lugar de la información bi-canal es mejor, dado que la señal secundaria es con frecuencia más ruidosa que la primaria en nuestra configuración de micrófono dual.

- Las arquitecturas de procesado de señal-redes neuronales profundas explotan las mejores características de los paradigmas de procesado de señal y aprendizaje profundo. De hecho, se espera que estas continúen siendo investigadas en el futuro inmediato de un modo satisfactorio. A este respecto, hemos explorado dos aproximaciones basadas en aprendizaje profundo de doble canal para afrontar el diseño de dos etapas complejas (desde el punto de vista analítico) de un sistema de RAH robusto al ruido. Así, fueron entrenadas redes neuronales profundas para obtener eficientemente estimaciones de máscaras de datos perdidos y de ruido con una buena capacidad de generalización a través de explotar la diferencia de niveles de potencia entre los dos canales disponibles. De acuerdo con nuestros experimentos, estas estimaciones superan con claridad diferentes técnicas analíticas cuando son usadas, respectivamente, en conjunción con métodos de datos perdidos y de compensación de características. Además, puesto que la señal de voz se encuentra altamente atenuada en el canal secundario de un *smartphone* de doble micrófono empleado en posición de habla cercana, dicho canal es una muy buena referencia del ruido acústico presente. Por esta razón, la integración de la estrategia diseñada para el entrenamiento con consciencia del ruido introduce una

mayor incertidumbre, llevando así a una degradación del rendimiento. Si bien los algoritmos de estimación de ruido son capaces de tratar con el ruido estacionario de forma satisfactoria, esto mismo no ocurre en el caso de tipos de ruido más complejos. De hecho, la utilización de la información procedente del sensor secundario puede ser entendida como una clase de estrategia de entrenamiento con consciencia del ruido más robusta.

- Se ha presentado la que se ha venido en denominar estrategia combinatoria. Por ella, en el caso de que el DMI integre varios sensores frontales, se genera una señal primaria de calidad superior (para usarse junto con la señal procedente del sensor secundario) a partir de estos sensores frontales por medio de procesamiento de *array* de micrófonos. Esta noción combinatoria se ha extrapolado en el sentido de la aplicación de diferentes métodos robustos al ruido a lo largo de distintas etapas (es decir, dominios) del extractor de características. Experimentalmente ha sido demostrada la efectividad de tal aproximación. Los mejores resultados sobre el corpus CHiME-3 fueron obtenidos a través de la concatenación de *beamforming* más un tipo de post-filtrado, es decir, MVDR+(DSW-(U+Eq)$_{MMSE}$). Aparte, sobre las bases de datos AURORA2-2C-CT/FT, los mejores resultados fueron logrados por medio de realce de características de doble canal seguido de compensación de características VTS bi-canal, esto es, (DSW-(U+Eq)$_{MMSE}$)+(2-VTS$_b^C$). Estas combinaciones permiten alcanzar un excelente rendimiento a bajas SNRs, siendo este un resultado reseñable puesto que los dispositivos móviles son usados con frecuencia en entornos altamente ruidosos, p. ej., calles multitudinarias u otros espacios públicos.

- El entrenamiento multi-condición es un elemento esencial que incorporar en un sistema de RAH multicanal siempre que sea posible con el fin de proporcionar un buen punto de partida en términos de robustez frente al ruido. Desafortunadamente, experimentamos que el rendimiento de aquellos métodos que intentan estimar con precisión las características de voz limpia se ve seriamente limitado cuando se hace uso de modelos acústicos multi-estilo, especialmente en un escenario real de uso. Así, se prevé que aproximaciones de reconstrucción espectral basadas en modelos de mezcla de gaussianas de voz limpia y similares caigan en "vía muerta".

Diversas contribuciones al RAH robusto al ruido en dispositivos móviles con varios sensores han sido reunidas en esta Tesis como resultado de varias publicaciones científicas. Además, debemos destacar que la calidad de dos de estas publicaciones ha sido reconocida mediante la concesión de dos premios internacionales:

- Premio al mejor artículo de estudiante en EUSIPCO (European Signal Processing Conference) 2014 por el trabajo "*Feature Enhancement for Robust Speech Recognition on Smartphones with Dual-Microphone*" [119].
- Premio al mejor artículo en IberSPEECH 2016 por el trabajo "*Deep Neural Network-Based Noise Estimation for Robust ASR in Dual-Microphone Smartphones*" [123].

Las diferentes contribuciones técnicas resultantes de nuestro trabajo son sintetizadas a continuación:

- Dos técnicas fundamentales de realce del espectro de potencia de doble canal diseñadas a partir de los principios de sustracción espectral y MVDR, denominadas DCSS y P-MVDR, respectivamente [119].
- Un pesado espectral de doble canal en el dominio de potencia espectral, DSW, el cual incluye dos nuevos modos de estimar la SNR *a priori* en un contexto de doble canal para el cálculo de un filtro de Wiener así como un procedimiento de ecualización de ruido [121].
- Un método de estimación MMSE de la ganancia relativa de voz que puede ser usado con el objetivo de modelar linealmente el canal entre dos sensores. Si bien esta técnica podría usarse con diferentes propósitos tales como la definición del vector de dirección en *beamforming*, en nuestro caso se ha considerado en combinación con las tres contribuciones de realce del espectro de potencia de doble canal presentadas en esta Tesis.
- Un método de compensación de características VTS bi-canal basado en un esquema de formulación apilada [122] junto con un modo alternativo más robusto de calcular las probabilidades *a posteriori* requeridas por este método VTS.
- Dos métodos de doble canal basados en DNNs que, de un modo similar, estiman máscaras de datos perdidos [120] y ruido [123] en el dominio log-Mel. Dichas DNNs explotan la diferencia de niveles de potencia entre los dos canales disponibles para obtener, de forma eficiente y con buena capacidad de generalización, estimaciones precisas de máscaras de datos perdidos y ruido.
- Dos bases de datos de voz generadas como extensiones del corpus bien conocido Aurora-2 [148] para experimentar con un *smartphone* de micrófono dual usado tanto en condiciones de habla cercana como lejana: AURORA2-2C-CT (*Aurora-2 - 2 Channels - Close-Talk*) [119] y AURORA2-2C-FT (*Aurora-2 - 2 Channels - Far-Talk*) [121], respectivamente.

Bibliography

[1] *Artificial neural network (Wikipedia), https://en.wikipedia.org/wiki/ artificial_neural_network.* XIII, 113

[2] *ETSI ES 201 108 - Distributed speech recognition; Front-end feature extraction algorithm; Compression algorithms.* 130, 136

[3] *ETSI ES 202 050 - Distributed speech recognition; Advanced front-end feature extraction algorithm; Compression algorithms.* 31, 141, 144, 153, 163

[4] *Theano library, http://deeplearning.net/software/theano/.* 165, 168

[5] Acero, A.: *Acoustical and Environmental Robustness in Automatic Speech Recognition.* Cambridge University Press, 1993. 17

[6] Acero, A. *et al.*: *HMM adaptation using vector Taylor series for noisy speech recognition.* In *Proc. of 6th International Conference of Spoken Language Processing, October 16–20, Beijing, China*, pages 869–872, 2000. 18, 34, 38

[7] Anastasakos, T. *et al.*: *A compact model for speaker-adaptive training.* In *Proc. of 4th International Conference of Spoken Language Processing, October, Philadelphia, USA*, pages 1137–1140, 1996. 35

[8] Anguera, X., C. Wooters, and J. Hernando: *Acoustic beamforming for speaker diarization of meetings.* IEEE Transactions on Audio, Speech, and Language Processing, 15:2011–2023, 2007. 58

[9] Argawal, A. and Y. M. Cheng: *Two-stage Mel-warped Wiener filter for robust speech recognition.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, Keystone, USA*, pages 67–70, 1999. 31

[10] Atal, B. S.: *Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification.* J. Aco. Soc. Am., 55:1304–1312, 1974. 25, 26, 97

[11] Baby, D. *et al.*: *Exemplar-based speech enhancement for deep neural network based automatic speech recognition.* In *Proc. of 40th International Conference on Acoustics, Speech, and Signal Processing, April 19–24, Brisbane, Australia*, 2015. 24

[12] Bagchi, D. *et al.*: *Combining spectral feature mapping and multi-channel model-based source separation for noise-robust automatic speech recognition.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 13–17, Scottsdale, USA*, 2015. 51, 52, 58, 64

[13] Barker, J., M. Cooke, and D. P. W. Ellis: *Decoding speech in the presence of other sources.* Speech Communication, 45:5–25, 2005. 40

[14] Barker, J. *et al.*: *Soft decisions in missing data techniques for robust automatic speech recognition.* In *Proc. of 6th International Conference of Spoken Language Processing, October 16–20, Beijing, China*, 2000. XI, 42

[15] Barker, J. *et al.*: *The third 'CHiME' speech separation and recognition challenge: Dataset, task and baselines.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 13–17, Scottsdale, USA*, 2015. 51, 130, 135, 136, 137, 138, 141, 142

[16] Baum, L.: *An inequality and associated maximization technique in statistical estimation of probabilistic function of a Markov process.* Inequalities, 3:1–8, 1972. 8

[17] Bengio, Y.: *Learning deep architectures for AI.* Foundations and Trends in Machine Learning, 2:1–127, 2009. 113

[18] Berouti, M., R. Schwartz, and J. Makhoul: *Enhancement of speech corrupted by acoustic noise.* In *Proc. of 4th International Conference on Acoustics, Speech, and Signal Processing, April 2–4, Washington D.C., USA*, pages 208–211, 1979. 30, 83, 87

[19] Bingyin, X. and B. Changchun: *Wiener filtering based speech enhancement with weighted denoising auto-encoder and noise classification.* Speech Communication, 60:13–29, 2014. 79, 110

[20] Bishop, C. M.: *Pattern Recognition and Machine Learning.* Springer: Information Science and Statistics, 2006. 90, 185

[21] Blandin, C., E. Vincent, and A. Ozerov: *Multi-source TDOA estimation in reverberant audio using angular spectra and clustering.* Signal Processing, 92:1950–1960, 2012. 58, 141

[22] Boll, S. F.: *Suppression of acoustic noise in speech using spectral subtraction.* IEEE Transactions on Acoustics, Speech, and Signal Processing, 27:113–120, 1979. 29

[23] Bouquin-Jeannes, R. Le and G. Faucon: *Study of a voice activity detector and its influence on a noise reduction system.* Speech Communication, 16:245–254, 1995. 67

[24] Bu, S. *et al.*: *Second order vector Taylor series based robust speech recognition.* In *Proc. of 39th International Conference on Acoustics, Speech, and Signal Processing, May 4–9, Florence, Italy*, pages 1788–1792, 2014. 38

[25] Chen, S. F. *et al.*: *Advances in speech transcription at IBM under the DARPA EARS program.* IEEE Transactions on Audio, Speech, and Language Processing, 14:1596–1608, 2006. 2, 190

[26] Choi, J. H. and J. H. Chang: *Dual-microphone voice activity detection technique based on two-step power level difference ratio.* IEEE Transactions on Audio, Speech, and Language Processing, 22:1069–1081, 2014. 67

[27] Ciresan, D. C. *et al.*: *Deep, big, simple neural nets for handwritten digit recognition.* Neural Computation, 22:3207–3220, 2010. 115

[28] Cohen, I.: *Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging.* IEEE Transactions on Speech and Audio Processing, 11:466–475, 2003. 46, 169

[29] Cohen, I. and B. Berdugo: *Noise estimation by minima controlled recursive averaging for robust speech enhancement.* IEEE Signal Processing Letters, 9:12–15, 2002. 45

[30] Cook, R. K. *et al.*: *Measurement of correlation coefficients in reverberant sound fields.* Journal of the Acoustical Society of America, 27:1072–1077, 1955. 56

[31] Cooke, M. *et al.*: *Robust automatic speech recognition with missing data and unreliable acoustic data.* Speech Communication, 34:267–285, 2001. 23, 40, 119

[32] Davis, S. and P. Mermelstein: *Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences.* IEEE Transactions on Acoustics, Speech, and Signal Processing, 28:357–366, 1980. 4, 18

[33] Deng, L.: *Front-end, back-end, and hybrid techniques for noise-robust speech recognition.* Robust Speech Recognition of Uncertain or Missing Data: Theory and Application, pages 67–99, 2011. 22

[34] Deng, L. *et al.*: *Large vocabulary speech recognition under adverse acoustic environment.* In *Proc. of 6th International Conference of Spoken Language Processing, October 16–20, Beijing, China*, 2000. 23, 34

[35] Deng, L., J. Droppo, and A. Acero: *Log-domain speech feature enhancement using sequential MAP noise estimation and a phase-sensitive model of the acoustic environment.* In *Proc. of 7th Int. Conference of Spoken Language Processing, September 16–20, Denver, USA*, 2002. 18

[36] Deng, L., J. Droppo, and A. Acero: *Recursive estimation of nonstationary noise using iterative stochastic approximation for robust speech recognition.* IEEE Transactions on Speech and Audio Processing, 11:568–580, 2003. 23

[37] Deng, L., G. Hinton, and B. Kingsbury: *New types of deep neural network learning for speech recognition and related applications: an overview.* In *Proc. of 38th International Conference on Acoustics, Speech, and Signal Processing, May 26–31, Vancouver, Canada*, 2013. 109

[38] Deshpande, A.: *A beginner's guide to understanding convolutional neural networks.* https://adeshpande3.github.io/adeshpande3.github.io/A-Beginner's-Guide-To-Understanding-Convolutional-Neural-Networks/, 2016. XIII, 118

[39] Dharanipragada, S. and M. Padmanabhan: *A nonlinear unsupervised adaptation technique for speech recognition.* In *Proc. of 25th International Conference on Acoustics, Speech, and Signal Processing, June 5–9, Istanbul, Turkey*, pages 556–559, 2000. 28

[40] Droppo, J. and A. Acero: *Environmental robustness.* Handbook of Speech Processing (Springer), 2008. 27

[41] Du, J. *et al.*: *The USTC-iFlytek system for CHiME-4 challenge.* In *Proc. of 17th Annual Conference of the International Speech Communication Association, September 8–12, San Francisco, USA*, 2016. 52

[42] Eneman, K.: *Demo Beamforming.* https://perswww.kuleuven.be/∼u0023287/demo_beam/demo_beam.html, 1998. XI, 62

[43] Ephraim, Y. and D. Malah: *Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator.* IEEE Transactions on Acoustics, Speech and Signal Processing, 32:1109–1121, 1984. 42, 163

[44] Erkelens, J. S., R. C. Hendriks, and R. Heusdens: *On the estimation of complex speech DFT coefficients without assuming independent real and imaginary parts.* IEEE Signal Processing Letters, 15:213–216, 2008. 91, 186

[45] Erkelens, J. S. and R. Heusdens: *Fast noise tracking based on recursive smoothing of MMSE noise power estimates.* In *Proc. of 33rd International Conference on Acoustics, Speech, and Signal Processing, March 30–April 4, Las Vegas, USA*, 2008. 80

[46] Fan, N., J. Rosca, and R. Balan: *Speech noise estimation using enhanced minima controlled recursive averaging.* In *Proc. of 32nd International Conference on Acoustics, Speech, and Signal Processing, April 16–20, Honolulu, USA*, pages 581–584, 2007. 46

[47] Faubel, F.: *Speech feature enhancement for speech recognition by sequential Monte Carlo methods.* University of Karlsruhe, 2006. 108

[48] Faubel, F., J. McDonough, and D. Klakow: *A phase-averaged model for the relationship between noisy speech, clean speech and noise in the log-Mel domain.* In *EUROSPEECH 2008 – 9th Annual Conference of the International Speech Communication Association, September 22–26, Brisbane, Australia, Proceedings*, 2008. 18

[49] Faubel, F., J. McDonough, and D. Klakow: *On expectation maximization based channel and noise estimation beyond the vector Taylor series expansion.* In *Proc. of 35th International Conference on Acoustics, Speech, and Signal Processing, March 14–19, Dallas, USA*, 2010. 99, 103

[50] Flanagan, J. L., A. C. Surendran, and E. E. Jan: *Spatially selective sound capture for speech and audio processing.* Speech Communication, 13:207–222, 1993. 59

[51] Frey, B. *et al.*: *ALGONQUIN: Iterating Laplace's method to remove multiple types of acoustic distortion for robust speech recognition.* In *Proc. of 7th European Conference on Speech Communication and Technology, September 3–7, Aalborg, Denmark*, pages 901–904, 2001. 39

[52] Fu, Z., F. Fan, and J. Huang: *Dual-microphone noise reduction for mobile phone application.* In *Proc. of 38th International Conference on Acoustics, Speech, and Signal Processing, May 26–31, Vancouver, Canada*, pages 7239–7243, 2013. 66, 67

[53] Fujimoto, M. and T. Nakatani: *Feature enhancement based on generative-discriminative hybrid approach with GMMs and DNNs for noise robust speech recognition.* In *Proc. of 40th International Conference on Acoustics, Speech, and Signal Processing, April 19–24, Brisbane, Australia*, 2015. 157, 172, 202

[54] Gales, M. J. F.: *Model-based techniques for noise robust speech recognition.* Ph.D. thesis (University of Cambridge), 1995. 34, 37

[55] Gales, M. J. F.: *Maximum likelihood linear transformations for HMM-based speech recognition.* Computer, Speech and Language, 12:75–98, 1998. 32, 33

[56] Gauvain, J. L. and C. H. Lee: *Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains.* IEEE Transactions on Speech and Audio Processing, 2:291–298, 1994. 32

[57] Gehring, J. *et al.*: *Extracting deep bottleneck features using stacked auto-encoders.* In *Proc. of 38th International Conference on Acoustics, Speech, and Signal Processing, May 26–31, Vancouver, Canada*, 2013. 26

[58] Gemmeke, J. F., T. Virtanen, and K. Demuynck: *Exemplar-based joint channel and noise compensation.* In *Proc. of 38th International Conference on Acoustics, Speech, and Signal Processing, May 26–31, Vancouver, Canada*, 2013. 24

[59] Gillespie, B. and L. E. Atlas: *Acoustic diversity for improved speech recognition in reverberant environments.* In *Proc. of 27th International Conference on Acoustics, Speech, and Signal Processing, May 13–17, Orlando, USA*, pages 557–560, 2002. 59

[60] Gong, Y.: *Speech recognition in noisy environments: A survey.* Speech Communication, 16:261–291, 1995. 22

[61] Gong, Y.: *A method of joint compensation of additive and convolutive distortions for speaker-independent speech recognition.* IEEE Transactions on Speech and Audio Processing, 13:975–983, 2005. 34, 38

[62] Gong, Y.: *A method of joint compensation of additive and convolutive distortions for speaker-independent speech recognition.* IEEE Transactions on Speech and Audio Processing, 13:975–983, 2005. 37

[63] González, J. A., A. M. Peinado, and A. M. Gómez: *MMSE feature reconstruction based on an occlusion model for robust ASR.* Advances in Speech and Language Technologies for Iberian Languages, pages 217–226, 2012. 41

[64] González, J. A. *et al.*: *Efficient MMSE estimation and uncertainty processing for multienvironment robust speech recognition.* IEEE Transactions on Audio, Speech, and Language Processing, 19, 2011. 98

[65] González, J. A. *et al.*: *MMSE-based missing-feature reconstruction with temporal modeling for robust speech recognition.* IEEE Transactions on Audio, Speech, and Language Processing, 21:624–635, 2013. 18, 40, 93, 102, 108, 162, 163, 165

[66] González-López, J. A.: *Reconocimiento Robusto de Voz con Datos Perdidos o Inciertos.* Ph.D. thesis (University of Granada), 2013. 32, 40, 41

[67] Graves, A. and N. Jaitly: *Towards end-to-end speech recognition with recurrent neural networks.* In *Proc. of International Conference on Machine Learning, June 21–26, Beijing, China*, pages 1764–1772, 2014. 118

[68] Graves, A., A. Mohamed, and G. Hinton: *Speech recognition with deep recurrent neural networks.* In *Proc. of 38th International Conference on Acoustics, Speech, and Signal Processing, May 26–31, Vancouver, Canada*, 2013. 118

[69] Greensted, A.: *The Lab Book Pages: Delay Sum Beamforming.* http://www.labbookpages.co.uk/audio/beamforming/delaySum.html, 2012. XI, 58

[70] Grézl, F. *et al.*: *Probabilistic and bottle-neck features.* In *Proc. of 32nd International Conference on Acoustics, Speech, and Signal Processing, April 16–20, Honolulu, USA*, pages 757–760, 2007. 26

[71] Griffiths, L. and C. Jim: *An alternative approach to linearly constrained adaptive beamforming.* IEEE Transactions on Antennas and Propagation, 30:27–34, 1982. 61

[72] Hendriks, R. C., R. Heusdens, and J. Jensen: *MMSE based noise PSD tracking with low complexity.* In *Proc. of 35th International Conference on Acoustics, Speech, and Signal Processing, March 14–19, Dallas, USA*, 2010. 47, 169

[73] Hermansky, H., D. P. W. Ellis, and S. Sharma: *Tandem connectionist feature extraction for conventional HMM systems.* In *Proc. of 25th International Conference on Acoustics, Speech, and Signal Processing, June 5–9, Istanbul, Turkey*, pages 1635–1638, 2000. 26

[74] Hermansky, H., B. A. Hanson, and H. Wakita: *Perceptually based linear predictive analysis of speech*. In *Proc. of 10th International Conference on Acoustics, Speech, and Signal Processing, March 26–29, Tampa, USA*, pages 509–512, 1985. 4, 25

[75] Hermansky, H. and N. Morgan: *RASTA processing of speech*. IEEE Transactions on Speech and Audio Processing, 2:578–589, 1994. 25

[76] Heymann, J. *et al.*: *BLSTM supported GEV beamformer front-end for the 3rd CHiME challenge*. In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 13–17, Scottsdale, USA*, 2015. 43, 51, 63

[77] Higuchi, T. *et al.*: *Robust MVDR beamforming using time-frequency masks for online/offline ASR in noise*. In *Proc. of 41st International Conference on Acoustics, Speech, and Signal Processing, March 20–25, Shanghai, China*, 2016. 141, 142, 149

[78] Hilger, F. and H. Ney: *Quantile based histogram equalization for noise robust speech recognition*. In *Proc. of 7th European Conference on Speech Communication and Technology, September 3–7, Aalborg, Denmark*, pages 1135–1138, 2001. 28

[79] Himawan, I.: *Speech recognition using ad-hoc microphone arrays*. Ph.D. thesis (Queensland University of Technology), 2010. XI, 53, 59, 74

[80] Hinton, G.: *Training products of experts by minimizing contrastive divergence*. Neural Computation, 14:1771–1800, 2002. 117

[81] Hinton, G.: *A Practical Guide to Training Restricted Boltzmann Machines*. UTML TR 2010-003, 2010. 117, 121, 165, 168

[82] Hinton, G. *et al.*: *Deep neural networks for acoustic modeling in speech recognition*. IEEE Signal Processing Magazine, 29:82–97, 2012. 8, 109, 112, 114, 116, 117

[83] Hinton, G., S. Osindero, and Y. W. Teh: *A fast learning algorithm for deep belief nets*. Neural Computation, 18:1527–1554, 2006. 109, 112

[84] Hinton, G. and R. Salakhutdinov: *Reducing the dimensionality of data with neural networks*. Science, 313:504–507, 2006. 113, 116

[85] Hinton, G. *et al.*: *Improving neural networks by preventing co-adaptation of feature detectors*. http://arxiv.org/abs/1207.0580, 2012. 115

[86] Hirsch, G.: *FaNT - Filtering and noise adding tool, http://dnt.kr.hsnr.de/download.html.* 2005. 133

[87] Hochreiter, S.: *The vanishing gradient problem during learning recurrent neural nets and problem solutions.* Uncertainty, Fuzziness and Knowledge-Based Systems, 6:107–116, 1998. 114

[88] Hochreiter, S. *et al.*: *Gradient Flow in Recurrent Nets: the Difficulty of Learning Long-Term Dependencies.* IEEE Press, 2001. 112

[89] Hochreiter, S. and J. Schmidhuber: *Long short-term memory.* Neural Computation, 9:1735–1780, 1997. 118

[90] Hori, T. *et al.*: *The MERL/SRI system for the 3rd CHiME challenge using beamforming, robust feature extraction, and advanced speech recognition.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 13–17, Scottsdale, USA*, 2015. 51, 86

[91] Hout, J. and A. Alwan: *A novel approach to soft-mask estimation and log-spectral enhancement for robust speech recognition.* In *Proc. of 37th International Conference on Acoustics, Speech, and Signal Processing, March 25–30, Kyoto, Japan*, pages 4105–4108, 2012. 43, 87, 141, 142

[92] Hu, Y. and Q. Huo: *Irrelevant variability normalization based HMM training using VTS approximation of an explicit model of environmental distortions.* In *Proc. of 11th Annual Conference of the International Speech Communication Association, August 27–31, Antwerp, Belgium*, pages 1042–1045, 2007. 35

[93] Hughes, T. B. *et al.*: *Performance of an HMM speech recognizer using a real-time tracking microphone array as input.* IEEE Transactions on Speech and Audio Processing, 7:346–349, 1999. 50

[94] Ito, N. *et al.*: *Designing the Wiener post-filter for diffuse noise suppression using imaginary parts of inter-channel cross-spectra.* In *Proc. of 35th International Conference on Acoustics, Speech, and Signal Processing, March 14–19, Dallas, USA*, pages 2818–2821, 2010. 63

[95] Jalalvand, S. *et al.*: *Boosted acoustic model learning and hypothesis rescoring on the CHiME-3 task.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 13–17, Scottsdale, USA*, 2015. 51, 61

[96] Jensen, J. *et al.*: *MMSE estimation of complex-valued discrete Fourier coefficients with generalized gamma priors.* In *Proc. of 9th International Conference of Spoken Language Processing, September 17–21, Pittsburgh, USA*, 2006. 90

[97] Jeub, M. *et al.*: *Noise reduction for dual-microphone mobile phones exploiting power level differences.* In *Proc. of 37th Int. Conference on Acoustics, Speech, and Signal Processing, March 25–30, Kyoto, Japan*, pages 1693–1696, 2012. 66, 79, 80, 81, 170

[98] Johnson, D.: *Composing Music With Recurrent Neural Networks (hexahedria).* http://www.hexahedria.com/2015/08/03/composing-music-with-recurrent-neural-networks/, 2015. 109

[99] Johnson, D. H. and D. E. Dudgeon: *Array Signal Processing.* New Jersey: Prentice Hall, 1993. 51, 56, 57

[100] Kamath, S. and P. Loizou: *A multi-band spectral subtraction method for enhancing speech corrupted by colored noise.* In *Proc. of 27th International Conference on Acoustics, Speech, and Signal Processing, May 13–17, Orlando, USA*, pages IV–4164, 2002. 30, 83

[101] Kim, C. and R. M. Stern: *Feature extraction for robust speech recognition based on maximizing the sharpness of the power distribution and on power flooring.* In *Proc. of 35th International Conference on Acoustics, Speech, and Signal Processing, March 14–19, Dallas, USA*, pages 4574–4577, 2010. 4, 26

[102] Kum, J. M., Y. S. Park, and J. H. Chang: *Speech enhancement based on minima controlled recursive averaging incorporating conditional maximum a posteriori criterion.* In *Proc. of 34th International Conference on Acoustics, Speech, and Signal Processing, April 19–24, Taipei, Taiwan*, pages 4417–4420, 2009. 46

[103] Lefkimmiatis, S. and P. Maragos: *A generalized estimation approach for linear and nonlinear microphone array post-filters.* Speech Communication, 49:657–666, 2007. 63, 141

[104] Leggetter, C. and P. Woodland: *Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models.* Computer, Speech and Language, 9:171–185, 1995. 32

[105] Leonard, R.: *A database for speaker-independent digit recognition.* In *Proc. of 9th International Conference on Acoustics, Speech, and Signal Processing, March 19–21, San Diego, USA*, pages 328–331, 1984. 130, 132

[106] Li, B. and K. C. Sim: *Improving robustness of deep neural networks via spectral masking for automatic speech recognition.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 8–12, Olomouc, Czech Republic*, pages 279–284, 2013. 110

[107] Li, J. *et al.*: *An overview of noise-robust automatic speech recognition.* IEEE/ACM Transactions on Audio, Speech, and Language Processing, 22:745–777, 2014. 16, 18, 22, 24, 26, 28, 32, 35, 36, 40

[108] Li, J. *et al.*: *High-performance HMM adaptation with joint compensation of additive and convolutive distortions via vector Taylor series.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 9–13, Kyoto, Japan*, 2007. 18, 38

[109] Li, J. *et al.*: *HMM adaptation using a phase-sensitive acoustic distortion model for environment-robust speech recognition.* In *Proc. of 33rd International Conference on Acoustics, Speech, and Signal Processing, March 30–April 4, Las Vegas, USA*, pages 4069–4072, 2008. 38

[110] Li, J., M. L. Seltzer, and Y. Gong: *Improvements to VTS feature enhancement.* In *Proc. of 37th International Conference on Acoustics, Speech, and Signal Processing, March 25–30, Kyoto, Japan*, pages 4677–4680, 2012. 36

[111] Liang, X. *et al.*: *Semantic object parsing with local-global long short-term memory.* In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition, June 27–30, Las Vegas, USA*, 2016. 109

[112] Liao, H.: *Uncertainty Decoding for Noise Robust Speech Recognition.* Ph.D. thesis (University of Cambridge), 2007. XI, 4, 5

[113] Liao, H. and M. J. F. Gales: *Adaptive training with joint uncertainty decoding for robust recognition of noisy data.* In *Proc. of 32nd International Conference on Acoustics, Speech, and Signal Processing, April 16–20, Honolulu, USA*, pages 389–392, 2007. 35

[114] Lim, J. S. and A. V. Oppenheim: *Enhancement and bandwidth compression of noisy speech.* Proceedings of the IEEE, 67:1586–1604, 1979. 29, 31, 79

[115] Lin, L., W. H. Holmes, and E. Ambikairajah: *Subband noise estimation for speech enhancement using a perceptual Wiener filter.* In *Proc. of 28th International Conference on Acoustics, Speech, and Signal Processing, April 6–10, Hong Kong*, 2003. 79

[116] Lin, M., Q. Chen, and S. Yan: *Network in network.* arXiv:1312.4400v3, 2014. 111

[117] Lippmann, R. P.: *Speech recognition by machines and humans.* Speech Communication, 22:1–15, 1997. 2, 190

[118] Lollmann, H. W. and P. Vary: *Post-filter design for superdirective beamformers with closely spaced microphones.* In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, October 21–24, New York, USA*, pages 291–294, 2007. 63

[119] López-Espejo, I. *et al.*: *Feature enhancement for robust speech recognition on smartphones with dual-microphone.* In *Proc. of 22nd European Signal Processing Conference, September 1–5, Lisbon, Portugal*, pages 21–25, 2014. 178, 179, 207

[120] López-Espejo, I. *et al.*: *A deep neural network approach for missing-data mask estimation on dual-microphone smartphones: Application to noise-robust speech recognition.* Lecture Notes in Computer Science, 8854:119–128, 2014. 123, 179, 207

[121] López-Espejo, I. *et al.*: *Dual-channel spectral weighting for robust speech recognition in mobile devices.* Submitted to Digital Signal Processing. 178, 179, 207

[122] López-Espejo, I. *et al.*: *Dual-channel VTS feature compensation for noise-robust speech recognition on mobile devices.* IET Signal Processing, 11:17–25, 2017. 124, 179, 207

[123] López-Espejo, I. *et al.*: *Deep neural network-based noise estimation for robust ASR in dual-microphone smartphones.* Lecture Notes in Computer Science, 10077:117–127, 2016. 179, 207

[124] Lu, X. G. *et al.*: *Speech enhancement based on deep denoising auto-encoder.* In *Proc. of 14th Annual Conference of the International Speech Communication Association, August 25–29, Lyon, France*, pages 436–440, 2013. 109

[125] Ma, N. *et al.*: *Exploiting synchrony spectra and deep neural networks for noise-robust automatic speech recognition.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 13–17, Scottsdale, USA*, 2015. 51, 52, 58, 64, 111

[126] Macho, D. *et al.*: *Evaluation of a noise-robust DSR front-end on Aurora databases.* In *Proc. of 7th Int. Conference of Spoken Language Processing, September 16–20, Denver, USA*, pages 17–20, 2002. 31

[127] Marro, C., Y. Mahieux, and K. U. Simmer: *Analysis of noise reduction and dereverberation techniques based on microphone arrays with postfiltering.* IEEE Transactions on Speech and Audio Processing, 6:240–259, 1998. 63, 149

[128] Marsland, S.: *Machine Learning: An Algorithmic Perspective.* CRC Press, 2009. XIII, 115

[129] Martin, R.: *Spectral subtraction based on minimum statistics.* In *Proc. of 7th European Signal Processing Conference, September, Edinburgh, Scotland*, pages 1182–1185, 1994. 87

[130] Martin, R.: *Noise power spectral density estimation based on optimal smoothing and minimum statistics.* IEEE Transactions on Speech and Audio Processing, 9:504–512, 2001. 46, 169

[131] Martin, R.: *Speech enhancement based on minimum mean-square error estimation and Supergaussian priors.* IEEE Transactions on Speech and Audio Processing, 13:845–856, 2005. 90

[132] McCowan, I. A.: *Robust Speech Recognition using Microphone Arrays.* Ph.D. thesis (Queensland University of Technology), 2001. 53, 55, 62

[133] McCowan, I. A. and H. Bourlard: *Microphone array post-filter based on noise field coherence.* IEEE Transactions on Speech and Audio Processing, 11:709–716, 2003. 63

[134] McCowan, I. A., A. Morris, and H. Bourlard: *Improving speech recognition performance of small microphone arrays using missing data techniques.* In *Proc. of 7th Int. Conference of Spoken Language Processing, September 16–20, Denver, USA*, pages 2181–2184, 2002. 51

[135] Mikolov, T. *et al.*: *Recurrent neural network based language model.* In *Proc. of 11th Annual Conference of the International Speech Communication Association, September 26–30, Makuhari, Japan*, pages 1045–1048, 2010. 6

[136] Molau, S., M. Pitz, and H. Ney: *Histogram based normalization in the acoustic feature space.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 9–13, Madonna di Campiglio, Italy*, 2001. 26, 28

[137] Moreno, P.: *Speech Recognition in Noisy Environments.* Ph.D. thesis (Carnegie Mellon University), 1996. 18, 35, 39, 105, 153

[138] Moreno, P. J., B. Raj, and R. M. Stern: *A vector Taylor series approach for environment-independent speech recognition.* In *Proc. of 21st International Conference on Acoustics, Speech, and Signal Processing, May 7–10, Atlanta, GA*, pages 733–736, 1996. 18, 35, 99, 103, 108, 153

[139] Narayanan, A. and D. L. Wang: *Ideal ratio mask estimation using deep neural networks for robust speech recognition.* In *Proc. of 38th International Conference on Acoustics, Speech, and Signal Processing, May 26–31, Vancouver, Canada*, 2013. 43, 107, 110, 119, 120, 123

[140] Nelke, C. M., C. Beaugeant, and P. Vary: *Dual microphone noise PSD estimation for mobile phones in hands-free position exploiting the coherence and speech presence probability.* In *Proc. of 38th International Conference on Acoustics, Speech, and Signal Processing, May 26–31, Vancouver, Canada*, pages 7279–7283, 2013. 66, 81

[141] Ng, A. *et al.*: *Convolutional neural network.* http://ufldl.stanford.edu/tutorial/supervised/ConvolutionalNeuralNetwork/. 119

[142] Obuchi, Y.: *Multiple-microphone robust speech recognition using decoder-based channel selection.* In *Workshop on Statistical and Perceptual Audio Processing, Jeju, Korea*, 2004. 50

[143] Omologo, M. *et al.*: *Experiments of hands-free connected digit recognition using a microphone array.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, Santa Barbara, USA*, pages 490–497, 1997. 50

[144] Paliwal, K. K., B. Schwerin, and K. Wójcicki: *Speech enhancement using a minimum mean-square error short-time spectral modulation magnitude estimator.* Speech Communication, 54:282–305, 2012. 30

[145] Paliwal, K. K. and K. Yao: *Robust Speech Recognition Under Noisy Ambient Conditions.* Elsevier, 2010. 15

[146] Parihar, N. and J. Picone: *Aurora working group: DSR front end LVCSR evaluation AU/384/02.* Technical Report, Institute for Signal and Information Process, Mississippi State University, 2002. 38

[147] Paul, D. and J. Baker: *The design of Wall Street Journal-based CSR corpus.* In *Proc. of 2nd International Conference of Spoken Language Processing, October, Alberta, Canada*, pages 899–902, 1992. 135

[148] Pearce, D. and H. G. Hirsch: *The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions.* In *Proc. of 6th International Conference of Spoken Language Processing, October 16–20, Beijing, China*, pages 29–32, 2000. 26, 31, 130, 134, 179, 207

[149] Peinado, A. M. and J. C. Segura: *Speech Recognition over Digital Channels.* Wiley, 2006. XI, 15, 16, 30, 31, 107, 136, 144, 153

[150] Petersen, K. B. and M. S. Pedersen: *The Matrix Cookbook.* Technical University of Denmark, 2008. 37, 91, 99, 186

[151] Pfeifenberger, L. *et al.*: *Multi-channel speech processing architectures for noise robust speech recognition: 3rd CHiME challenge results.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 13–17, Scottsdale, USA*, 2015. 64, 149

[152] Povey, D. *et al.*: *The Kaldi speech recognition toolkit.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 11–15, Waikoloa, USA*, 2011. 135, 138

[153] Prudnikov, A., M. Korenevsky, and S. Aleinik: *Adaptive beamforming and adaptive training of DNN acoustic models for enhanced multichannel noisy speech recognition.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 13–17, Scottsdale, USA*, 2015. 51, 52, 63

[154] Qian, Y. *et al.*: *Very deep convolutional neural networks for noise robust speech recognition.* IEEE/ACM Transactions on Audio, Speech, and Language Processing, 24:2263–2276, 2016. 109, 119

[155] Raj, B., M. L. Seltzer, and R. M. Stern: *Reconstruction of missing features for robust speech recognition.* Speech Communication, 48:275–296, 2004. 40

[156] Rangachari, S. and P. C. Loizou: *A noise-estimation algorithm for highly non-stationary environments.* Speech Communication, 48:220–231, 2006. 46, 169

[157] Reynolds, D. A.: *Gaussian Mixture Models.* Encyclopedia of Biometric Recognition, 2008. 93

[158] Roweis, S. T.: *Factorial models and refiltering for speech separation and denoising.* In *Proc. of 8th European Conference on Speech Communication and Technology, September 1–4, Geneve, Switzerland*, pages 1009–1012, 2003. 40

[159] Segura, J. C. *et al.*: *Model-based compensation of the additive noise for continuous speech recognition. Experiments using the AURORA II database and tasks.* In *Proc. of 7th European Conference on Speech Communication and Technology, September 3–7, Aalborg, Denmark*, 2001. 85, 105, 124, 125, 153

[160] Seltzer, M. L.: *Microphone Array Processing for Robust Speech Recognition.* Ph.D. thesis (Carnegie Mellon University), 2003. 50, 53, 58, 62

[161] Seltzer, M. L., B. Raj, and R. M. Stern: *Likelihood-maximizing beamforming for robust hands-free speech recognition.* IEEE Transactions on Speech and Audio Processing, 12:489–498, 2004. 50

[162] Seltzer, M. L. and R. M. Stern: *Subband likelihood-maximizing beamforming for speech recognition in reverberant environments.* IEEE Transactions on Audio, Speech, and Language Processing, 14:2109–2121, 2006. 50

[163] Seltzer, M. L., D. Yu, and Y. Wang: *An investigation of deep neural networks for noise robust speech recognition.* In *Proc. of 38th International Conference on Acoustics, Speech, and Signal Processing, May 26–31, Vancouver, Canada*, 2013. 109, 115, 125

[164] Shao, Y. *et al.*: *A computational auditory scene analysis system for speech segregation and robust speech recognition.* Computer Speech & Language, 24:77–93, 2010. 25

[165] Shinoda, K. and C. H. Lee: *Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains.* IEEE Transactions on Speech and Audio Processing, 9:276–287, 2001. 32

[166] Simmer, K. U., S. Fischer, and A. Wasiljeff: *Suppression of coherent and incoherent noise using a microphone array.* Annals of telecommunications, 7/8:439–446, 1994. 63

[167] Siohan, O., C. Chesta, and C. H. Lee: *Joint maximum a posteriori adaptation of transformation and HMM parameters.* IEEE Transactions on Speech and Audio Processing, 9:417–428, 2001. 32

[168] Siohan, O., T. A. Myrvoll, and C. H. Lee: *Structural maximum a posteriori linear regression for fast HMM adaptation.* Computer, Speech and Language, 16:5–24, 2002. 32

[169] Sivasankaran, S. *et al.*: *Robust ASR using neural network based speech enhancement and feature simulation.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 13–17, Scottsdale, USA*, 2015. 51

[170] Sreenivas, T. V. and P. Kirnapure: *Codebook constrained Wiener filtering for speech enhancement.* IEEE Transactions on Speech and Audio Processing, 4:383–389, 1996. 79

[171] Statista: *Global smartphone sales by operating system from 2009 to 2015 (in millions).* https://www.statista.com/statistics/263445/global-smartphone-sales-by-operating-system-since-2009/, 2016. XI, 1

[172] Stern, R. M. and N. Morgan: *Features based on auditory physiology and perception.* Techniques for Noise Robustness in Automatic Speech Recognition, 2012. 26

[173] Stevens, S. S., J. Volkmann, and E. B. Newman: *A scale for the measurement of the psychological magnitude pitch.* The Journal of the Acoustical Society of America, 8:185–190, 1937. 5, 19

[174] Stolbov, M. and S. Aleinik: *Improvement of microphone array characteristics for speech capturing.* Modern Applied Science, 9:310–319, 2015. 52

[175] Stouten, V., H. Van Hamme, and P. Wambacq: *Effect of phase-sensitive environment model and higher order VTS on noisy speech feature enhancement.* In *Proc. of 30th International Conference on Acoustics, Speech, and Signal Processing, March 18–March 23, Philadelphia, USA*, pages 433–436, 2005. 39

[176] Stouten, V., H. Van Hamme, and P. Wambacq: *Model-based feature enhancement with uncertainty decoding for noise robust ASR.* Speech Communication, 48:1502–1514, 2006. 18, 39, 105

[177] Sun, M. *et al.*: *Speech enhancement under low SNR conditions via noise estimation using sparse and low-rank NMF with Kullback–Leibler divergence.* IEEE/ACM Transactions on Audio, Speech, and Language Processing, 23:1233–1242, 2015. 44

[178] Taghia, J. *et al.*: *An evaluation of noise power spectral density estimation algorithms in adverse acoustic environments.* In *Proc. of 36th International Conference on Acoustics, Speech, and Signal Processing, May 22–27, Prage, Czech Republic*, 2011. 49

[179] Tashev, I. *et al.*: *Sound capture system and spatial filter for small devices.* In *EUROSPEECH 2008 – 9th Annual Conference of the International Speech Communication Association, September 22–26, Brisbane, Australia, Proceedings*, pages 435–438, 2008. 3, 9, 52, 65, 145, 176, 191, 204

[180] Tashev, I., M. Seltzer, and A. Acero: *Microphone array for headset with spatial noise suppressor.* In *IWAENC 2005 – 9th International Workshop on Acoustic, Echo and Noise Control, Proceedings*, 2005. 3, 9, 52, 65, 145, 176, 191, 204

[181] Torre, A. de la *et al.*: *Histogram equalization of speech representation for robust speech recognition.* IEEE Transactions on Speech and Audio Processing, 13:355–366, 2005. 26

[182] Truax, B.: *Handbook for Acoustic Ecology.* Cambridge St. Pub., 1999. 73, 83

[183] Vaseghi, S. V.: *Advanced Digital Signal Processing and Noise Reduction (4th Edition).* John Wiley & Sons, 2008. 30, 84

[184] Veselý, K. *et al.*: *Sequence-discriminative training of deep neural networks.* In *Proc. of 14th Annual Conference of the International Speech Communication Association, August 25–29, Lyon, France*, pages 2345–2349, 2013. 138

[185] Viikki, O., D. Bye, and K. Laurila: *A recursive feature vector normalization approach for robust speech recognition in noise.* In *Proc. of 23rd International Conference on Acoustics, Speech, and Signal Processing, May 12–15, Seattle, USA*, pages 733–736, 1998. 26

[186] Vincent, E.: *Is audio signal processing still useful in the era of machine learning?* WASPAA, New York, USA, 2015. 108

[187] Vincent, E. *et al.*: *The 4th CHiME speech separation and recognition challenge.* http://spandh.dcs.shef.ac.uk/chime_challenge/, 2016. 52

[188] Vincent, P. *et al.*: *Extracting and Composing Robust Features with Denoising Autoencoders.* Technical Report 1316, 2008. 109

[189] Viterbi, A.: *Error bounds for convolutional codes and an asymptotically optimum decoding algorithm.* IEEE Transactions on Information Theory, 13:260–269, 1967. 5, 9

[190] Vu, T. T., B. Bigot, and E. S. Chng: *Speech enhancement using beamforming and non negative matrix factorization for robust speech recognition in the CHiME-3*

Challenge. In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 13–17, Scottsdale, USA*, 2015. 51

[191] Wang, X. *et al.*: *Noise robust IOA/CAS speech separation and recognition system for the third 'CHiME' challenge.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 13–17, Scottsdale, USA*, 2015. 52

[192] Wang, Y., K. Han, and D. L. Wang: *Exploring monaural features for classification-based speech segregation.* IEEE Transactions on Audio, Speech, and Language Processing, 21:270–279, 2013. 43

[193] Wang, Y. and D. L. Wang: *Towards scaling up classification-based speech separation.* IEEE Transactions on Audio, Speech, and Language Processing, 21:1381–1390, 2013. 43, 107, 109, 110, 119, 120, 123

[194] Wang, Z. Q. and D. L. Wang: *Robust speech recognition from ratio masks.* In *Proc. of 41st International Conference on Acoustics, Speech, and Signal Processing, March 20–25, Shanghai, China*, 2016. 43

[195] Watanabe, S. and J. T. Chien: *Bayesian speech and language processing.* Cambridge University Press, 2015. 108

[196] Weng, C. *et al.*: *Recurrent deep neural networks for robust speech recognition.* In *Proc. of 39th International Conference on Acoustics, Speech, and Signal Processing, May 4–9, Florence, Italy*, pages 5532–5536, 2014. 109, 138

[197] Weninger, F. *et al.*: *Non-negative matrix factorization for highly noise-robust ASR: To enhance or to recognize?* In *Proc. of 37th International Conference on Acoustics, Speech, and Signal Processing, March 25–30, Kyoto, Japan*, 2012. 24

[198] Werbos, P. J.: *Backpropagation through time: what it does and how to do it.* Proceedings of the IEEE, 78:1550–1560, 2002. 118

[199] Xiao, X. *et al.*: *A study of learning based beamforming methods for speech recognition.* In *Proc. of 17th Annual Conference of the International Speech Communication Association, September 8–12, San Francisco, USA*, 2016. 52

[200] Xie, J., L. Xu, and E. Chen: *Image denoising and inpainting with deep neural networks.* In *Proc. of 26th Annual Conference on Neural Information Processing Systems, December 3–8, Lake Tahoe, USA*, pages 350–358, 2012. 109

[201] Xu, Y. *et al.*: *An experimental study on speech enhancement based on deep neural networks.* IEEE Signal Processing Letters, 21:65–68, 2014. 24, 109

[202] Xu, Y. *et al.*: *A regression approach to speech enhancement based on deep neural networks.* IEEE Transactions on Audio, Speech, and Language Processing, 23:7–19, 2015. 24, 109, 115, 125

[203] Yoshioka, T. *et al.*: *The NTT CHiME-3 system: Advances in speech enhancement and recognition for mobile multi-microphone devices.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 13–17, Scottsdale, USA*, 2015. 51, 61, 85, 111

[204] Young, S. *et al.*: *The HTK Book (for HTK Version 3.4).* Cambridge University Engineering Department, 2006. XI, 7, 138

[205] Yousefian, N., A. Akbaria, and M. Rahmani: *Using power level difference for near field dual-microphone speech enhancement.* Applied Acoustics, 70:1412–1421, 2009. 66, 78, 79, 80

[206] Yu, D. and L. Deng: *Deep learning and its applications to signal and information processing.* IEEE Signal Processing Magazine, pages 145–154, 2011. 107, 119

[207] Yu, D. *et al.*: *Use of incrementally regulated discriminative margins in MCE training for speech recognition.* In *Proc. of 9th International Conference of Spoken Language Processing, September 17–21, Pittsburgh, USA*, 2006. 2, 190

[208] Yu, R.: *A low-complexity noise estimation algorithm based on smoothing of noise power estimation and estimation bias correction.* In *Proc. of 34th International Conference on Acoustics, Speech, and Signal Processing, April 19–24, Taipei, Taiwan*, pages 4421–4424, 2009. 47

[209] Zhang, J. *et al.*: *A fast two-microphone noise reduction algorithm based on power level ratio for mobile phone.* In *Proc. of 8th International Symposium on Chinese Spoken Language Processing, December 5–8, Hong Kong*, pages 206–209, 2012. 66

[210] Zhao, S. *et al.*: *Robust speech recognition using beamforming with adaptive microphone gains and multichannel noise reduction.* In *Proc. of IEEE Automatic Speech Recognition and Understanding, December 13–17, Scottsdale, USA*, 2015. 51, 52, 61, 63, 141

[211] Ziomek, L.: *Fundamentals of Acoustic Field Theory and Space-Time Signal Processing.* CRC Press, 1994. 141

[212] Zwicker, E.: *Subdivision of the audible frequency range into critical bands (frequenzgruppen).* The Journal of the Acoustical Society of America, 33, 1961. 25