Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Circuit Theory

Doctoral Thesis

September 2016 Michal Borský

Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Circuit Theory

Robust recognition of strongly distorted speech

Doctoral Thesis

Author: Michal Borský
Supervisor: Doc. Ing. Petr Pollák, CSc.
Branch of study: Electrical Engineering Theory
September 2016

ABSTRACT

The automatic speech recognition systems have become a part of our daily lives. People often rely on virtual personal assistants in smartphones, use their voice to control intelligent devices in cars and smart homes or communicate with automatic dialogue systems in call-centres. Since these systems often suffer from a performance drop in realistic acoustic conditions which are characterized by strong distortions, a large portion of research still must be focused on robust front-end algorithms and acoustic modelling methods for distorted speech recognition. This thesis is focused on these compensation methods working at the level of front-end processing and acoustic modelling, whose aim is to compensate the degradation introduced by a distant microphone, noisy environments and a lossy compression.

The techniques for noisy and distant speech recognition studied in this thesis were focused on front-end noise suppression techniques, feature normalization techniques, acoustic model adaptations and discriminative training. Said techniques were evaluated in three different car conditions and two different public environments. The experiments have proved, that extended spectral subtraction can bring significant improvement even for the state-of-the-art systems in public environments with a strong noise and for a far-distance microphone recordings.

The evaluation of compressed speech recognition examined the degrading effects of lossy compression on fundamental frequency, formants and smoothed LPC spectrum and for standard MFCC and PLP features used for ASR. The low-pass filtering and the areas of very low energy in a spectrogram were identified as the two main reasons of degradation. The practical experiments evaluated the contributions of specific feature extraction setups, combinations of normalization and compensation techniques, supervised and unsupervised adaptation and discriminative training methods and finally the matched training. The largest contributions were gained from the application of adaptation techniques, subspace GMM and discriminative training.

A novel algorithm named *Spectrally selective dithering* (SSD) was proposed within this thesis, which compensated the effect of spectral valleys. The contribution of said algorithm was verified for both GMM-HMM and DNN-HMM speech recognition systems for Czech and English and for a GMM-HMM system for German. The practical experiments proved that the proposed algorithm can lower *WER* for all languages with GMM-HMM systems. Concerning DNN-HMM system, a significant contribution was achieved only for Czech.

ABSTRAKT

Systémy automatického rozpoznávania reči prenikli do mnohých oblastí nášho života. Stále častejšie sa spoliehame na personálnych asistentov pri obsluhe mobilných zariadení, hlasové ovládanie spotrebičov v domácnosti alebo navigácie v automobile, prípadne komunikujeme s automatickými dialógovými systémami v call centrách. Pri nasadení týchto systémov do reálnych akustických podmienok vyznačujúcich sa zvýšenou úrovňou rušenia sa však stretávame s výrazným poklesom ich úspešnosti. Je preto stále nevyhnutné, aby sa značná časť výskumu zaoberala robustnými metódami zpracovania rečového signálu a akustického modelovania. Táto práca analyzuje metódy pracujúce na úrovni predzpracovania signálu a akustického modelovania v úlohách rozpoznávania nahrávok zo vzdialeného mikrofónu, hlučného prostredia auta a po aplikácii ztrátovej kompresie.

Techniky pre kompenzáciu vplyvu vzdialeného mikrofónu a hlučného prostredia boli zamerané na algoritmy odstraňovania šumu, normalizácie príznakov, adaptácie akustického modelu a nakoniec vplyv diskriminatívnych techník trénovania akustického modelu. Ich prínos bol ohodnotený pre tri rôzne autové prostredia s vysokým SNR a dve rôzne verejné prostredia s vysokou úrovňou aditívneho a konvolučného zkreslenia. Praktické experimenty ukázali, že použitie rozšíreného spektrálneho odčítania prináša výrazne zlepšenie aj pre súčasné systémy v prípade, že nahrávky pochadzajú z verejného prostredia s výrazným šumom a vzdialeného mikrofónu.

Vplyv ztrátového kodéru bol analyzovaný na úlohách odhadu základného tónu, formantových kmitočtov, vyhladeného LPC spektra a nakoniec pre štandardné MFCC a PLP príznaky používané pre systémy rozpoznávania reči. Táto analýza odhalila, že hlavné príčiny zhoršenia sú nízko-pásmová filtrácia a oblasti s prakticky nulovou energiou vo spektru, nazývané tiež spektrálne údolia. Následné praktické experimenty analyzujú vplyv špecifického nastavenia pri extrakcií príznakov, kombinácie normalizačných a kompenzačných techník, riadenej a neriadej adaptácie, diskriminatívneho a prispôsoveného trénovania. Najväčší prínos bol dosiahnutý s pomocou adaptácie akustického modelu, subspace GMM a diskriminatívneho trénovania.

Táto práca navrhuje nový algoritmus s názvom *Spektrálne selektívne zašumovanie*, ktorý kompenzuje spektrálne údolia. Prínos tohoto algoritmu bol zkúmaný pre GMM-HMM a DNN-HMM systémy pre český, anglický a nemecký jazyk. Exprimenty potvrdili jeho prínos pre GMM-HMM systémy pre všetky jazyky. Štatistický významný prínos pre DNN-HMM systém bol potvrdený len pre český jazyk.

ACKNOWLEDGEMENTS

This thesis was done at Czech Technical University in Prague[1]. I would like to express my thanks to my supervisor Petr Pollák for introducing me to the field of digital signal processing and automatic speech recognition. I have started my scientific work under him with my Master's thesis and then followed to my PhD. studies. It was also thanks to his insistence that I was introduced to Linux and HTK. Thank You for giving me a chance.

My thanks goes to the PhD. colleagues I have encountered during my studies. In particular, to Petr Mizera for our discussions on everything related to our research, introducing me to KALDI and your catchy enthusiasm to always work harder; to Honza Sedlák for our discussions on everything that was unrelated to our studies. Thank you guys for being my friends.

My thanks also goes to other members of our small lab, namely Václav Hanžl and Zdeněk Horčík. Mr. Hanžl who provided me with an endless well of advices on everything related to Linux and ASR and Mr. Horčík for your knowledge of computer hardware and for helping us build our recognition cluster. I have learnt a lot. My thanks also go to the head our our department, Pavol Sovka, I really liked your DPS lectures.

And finally, I am forever thankful to my whole family. It was only thanks to you, mom and dad, that I could start my university studies. I would not be here if it wasn't for you.

[1]The research described in this thesis was supported by CTU Grants SGS12/143/OHK3/2T/13 "Algorithms and Hardware Realizations of Digital Signal Processing" and SGS14/191/OHK3/3T/13 "Advanced Algorithms of Digital Signal Processing and their Applications".

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AM	Acoustic Model
ASR	Automatic Speech Recognition
bMMI	Boosted Maximum Mutual Information
CMLLR	Constrained Maximum Likelihood Linear Regression
CMN	Cepstral Mean Normalization
CMS	Cepstral Mean Subtraction
CMVN	Cepstral Mean and Variance Normalization
DNN	Deep Neural Networks
DT	Discriminative Training
ESS	Extended Spectral Subtraction
fMLLR	feature Maximum Likelihood Linear Regression
GMM	Gaussian Mixture Models
HMM	Hidden Markov Models
HTK	Hidden Markov Model Toolkit
LDA	Linear Discriminant Analysis
LM	Language Model
LPC	Linear Predictive Coding
LTI	Linear Time-Invariant
LVCSR	Large Vocabulary Continuous Speech Recognition
MAP	Maximum A-posteriori Probability
MFCC	Mel-Frequency Cepstral Coefficients
MLE	Maximum Likelihood Estimation
MLLR	Maximum Likelihood Linear Regression
MMI	Maximum Mutual Information
MP3	MPEG-1 Audio Layer III
MPE	Minimum Phone Error
PAC	Perceptual Audio Coding
PER	Phone Error Rate
PERR	Phone Error Rate Reduction
PERC	Phone Error Rate Contribution
PLP	Perceptual Linear Prediction

SA	Speaker Adapted
SBR	Spectral Band Replication
SI	Speaker Independent
SSD	Spectrally Selective Dighering
WER	Word Error Rate
WERR	Word Error Rate Reduction

CHAPTER 1

INTRODUCTION

Human to machine interaction has become a common form of communication in the current world. This interaction was historically provided by devices such as a keyboard, a mouse or a touchscreen but the current trend is to replace these with automatic speech recognition (ASR) systems in situations which require a more natural form of communication. It is becoming fairly common nowadays, that people make use of personal assistants built in their smartphones, control the intelligent appliances in their homes and offices with the voice or communicate with automatic dialogue systems in the call centres. Also, the automatic transcription systems are being used to create subtitles for television broadcasts, to index the audio archives or to transcribe personal recordings. This progress came with the introduction of advanced signal processing and machine learning algorithms as well as due to the massive increase in available data and computational power.

However, the variability of the speech greatly increases the difficulty of these tasks. The speaker-based variability is natural and carries additional paralinguistic information, but has no relevance to the content. The general tendency is to remove this information by normalizing the vocal tract parameters or using a speaker adaptation. For example, an ASR system based on artificial neural networks intended for the large vocabulary continuous speech recognition (LVCSR) task which is designed for clear acoustic conditions and is adapted to a particular speaker can achieve the accuracy as high as 95%. However, even these modern systems still struggle with conditions regularly encountered in real-life situations which in turn limits their usability. The recording conditions introduce additional variability that degrades the signals quality and we often say that the speech is distorted. The degrading conditions are generally divided into two main groups.

- **Environmental**: The most relevant factors in this group are the types and the levels of noises present during recording. The additive noises are present for nearly all public environments such as a street, a driving car or an auditorium. The convolutional noises occur for recordings done in closed spaces, where the sounds is reflected back to the speaker as echoes or reverberations.

- **Channel**: The relevant factors in this group are the types and positions of a microphone, employed coding and compression. Each microphone has a specific transfer function which alters the spectrum of speech. The second important thing is its position, whether the microphone is close to the speaker or not. Third, the signals are often coded and compressed in order to reduce their size for further transmission and storage, which introduces unwanted compression artifacts.

The task of the robust speech recognition is to reduce the impact of the above mentioned adverse acoustic conditions by removing, or at least suppressing, their effects. The common options include suitable pre-processing algorithms or extracting robust features. Another option is to employ robust ASR architectures which make use of the multi-conditional training, parallel model combination or acoustic model adaptation. The additive noise is often removed in the spectral domain while the convolution noise is often suppressed in logarithmic-spectral domain. Distortions introduced by coding and compression are harder to address as they often remove "information" rather than add a new one. The purpose of this thesis is to contribute to research focused on distorted speech recognition, namely for the distant, a noisy car and compressed speech. It explores algorithms working at the level of signal processing, feature extraction and acoustic model training and proposes a compensation method designed for compressed speech which works at the level of front-end processing. The thesis is structured as follows.

- Chapter 2 provides an overview of an ASR system and focuses more closely on the front-end signal processing and acoustic modelling blocks. These are limited to the techniques applicable for robust speech recognition, as they are the main research topics of this thesis.
- Chapter 3 presents the set research goals and introduces the questions this thesis attempts to answer.
- Chapter 4 describes the resources used in the experimental part and presents two optimization steps for constructing the baseline acoustic model.
- Chapter 5 focuses on noisy car and distant speech recognition from the public environment and evaluates the contribution of the front-end processing and acoustic modelling techniques in this task.
- Chapter 6 focuses on compressed speech recognition. It begins with the theoretical description of the compression, introduces the distortions and evaluates their effects in an ASR system. This analysis lays the groundwork which is used to design a novel compensation method proposed by this thesis.
- Chapter 7 presents a novel compensation method named *Spectrally Selective Dithering* (SSD) and demonstrates its contribution for Czech, English and German. The experimental part concludes with a comparison of SSD against a perceptually-motivated compensation technique called *Spectral Band Replication* (SBR).
- Chapter 8 summarizes the findings from the previous chapters and draws the final conclusions and outlines the directions for the future research in these areas.

CHAPTER 2

AUTOMATIC SPEECH RECOGNITION

This chapter aims to introduce two state-of-the-art ASR architectures together with their relative strengths as well as their weaknesses. The special focus is given to methods which directly contribute to the acoustic model quality and which were optimized for the distorted speech recognition. Typical architecture of an ASR system can be described by a simple block scheme in Figure 2.1.

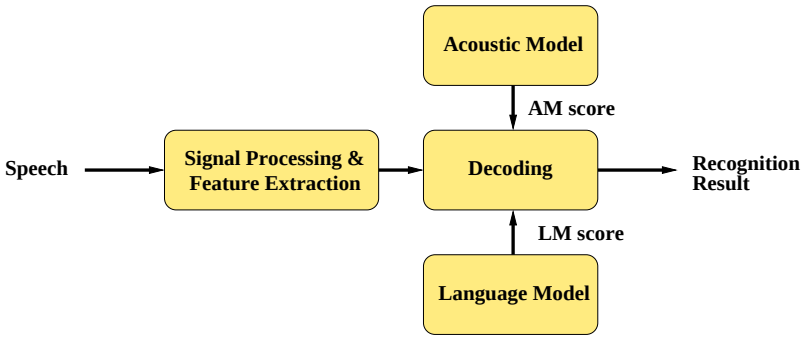

Figure 2.1: General stochastic ASR system

The task of the signal processing and feature extraction block is to take the audio signal on the input, preprocess the signal and extract the feature vectors that are suitable for the following acoustic modelling. It usually involves transforming the speech from the time domain into the frequency domain and enhancing its quality. The acoustic model (AM) block takes the feature vectors as an input and outputs the acoustic score for a set of fixed, usually subword, phonetic units. The knowledge about phonetics is essential at this step and represents one of the most important a priori decisions that determines the AM quality. The purpose of the language model (LM) block is to estimate the probability of generating the hypothesized word sequence given the set of all possible word sequences.

This task is highly domain-dependent and thus we often see specialized LMs developed for a specific domain. The decoding block combines the acoustic score for the given acoustic observations and the language score for given word sequence and outputs the most likely word sequence as the recognition result. Within this thesis I focus on the improvements to the blocks of *Speech Processing and Feature Extraction* and the *Acoustic Model*. The following sections describe a typical recognition system and focus in more details on robust methods of AM creation which were explored in the experimental part of this thesis.

2.1 Stochastic ASR

Nowadays, the two different architectures are used for ASR. The first one is based on the combination of Gaussian Mixture Models (GMM) and Hidden Markov Models (HMM). The purpose of the GMM is to statistically model the speech variability while the HMM is a probabilistic finite state machine that can model the varying speech length. In the past, this approach played a dominant role in the ASR field and produced first state-of-the-art systems that were capable of delivering the needed performance to pass the bar for commercial use. These systems often used Mel-frequency cepstral coefficients (MFCC) as input feature vectors, speaker-adapted AM and a statistical n-gram LM. Their rise in popularity begun in the early 90's but their appeal slowly faded away since the introduction of "new" models based on artificial neural networks around 2010. Nevertheless, most of the experiments in this thesis were done using a GMM-HMM system and therefore their description will be more thorough than that of neural networks.

The second architecture is based on "modern" artificial neural networks and a completely new field of *deep learning* has been created since their popularization. These discriminative hierarchical models have surpassed conventional GMM-HMM systems and replaced them as the state-of-the-art ASR systems for practically every recognition task. This rapid shift started due to the progress made in several key areas. First, the computational power has become more available and its power has increased massively with deployment of parallel processing units such as GPUs. Second, the amount of available data has increased as well. There was also another reason for a resurgence of neural nets. The original multilayer nets had their parameters initialized randomly and then trained using back-propagation algorithm. This approach often led to the problem of vanishing or exploding gradient. This problem was effectively solved by introducing the pre-training step which initialized the parameters, often one layer at a time, see [1]. Current trends in deep learning include many different architectures some of which are : deep neural networks (DNN), convolutional neural networks or recurrent neural networks. The description in this thesis will focus on DNN-HMM as it was the only architecture used for practical experiments.

Let us now define a statistical speech recognizer that is independent of the used AM architecture. Lets assume a process that generates the sequence of acoustic observations $\boldsymbol{O} = \{o_1, o_2, ...o_T\}$ given the sequence of words $\boldsymbol{W} = \{w_1, w_2, ...w_N\}$. The primary goal of the speech recognizer is to answer the question "*What is the most likely sequence*

of words $\hat{\boldsymbol{W}}$ *given the sequence of acoustic observation* $\boldsymbol{O}$ *for our model defined by its set of parameters* $\boldsymbol{\Theta}$*?*". This problem can be formulated in a mathematical way as the conditional probability by using the formula:

$$\hat{\boldsymbol{W}} = \arg\max_{\boldsymbol{W}}, P(\boldsymbol{W}|\boldsymbol{O}, \boldsymbol{\Theta}). \tag{2.1}$$

As it is not possible to estimate the probability of $P(\boldsymbol{W}|\boldsymbol{O}, \boldsymbol{\Theta})$ directly, we can use Bayesian rule and rewrite Eq. (2.1) into the form :

$$\hat{\boldsymbol{W}} = \arg\max_{\boldsymbol{W}}, \frac{P(\boldsymbol{O}|\boldsymbol{W}, \boldsymbol{\Theta})P(\boldsymbol{W}|\boldsymbol{\Theta})}{P(\boldsymbol{O}|\boldsymbol{\Theta})}, \tag{2.2}$$

which contains two terms in the numerator and a single term in the denominator. The denominator term $P(\boldsymbol{O}|\boldsymbol{\Theta})$ represents the a priori probability of the observation sequence $\boldsymbol{O}$, which is constant for all hypothesis $\hat{\boldsymbol{W}}$, and thus can be omitted from the equation. The term $P(\boldsymbol{O}|\boldsymbol{W}, \boldsymbol{\Theta})$ is now the probability of the observation sequence $\boldsymbol{O}$ given the word sequence $\boldsymbol{W}$ and the $P(\boldsymbol{W}|\boldsymbol{\Theta})$ is the probability of the word sequence $\boldsymbol{W}$ which is now independent of the observation. The Eq. (2.2) can be simplified if we further assume that the set of model parameters $\boldsymbol{\Theta}$ comprises of acoustic parameters $\boldsymbol{\Theta}_{AM}$ and language parameters Θ_{LM} which are independent of each other. The conditional probability $P(\boldsymbol{O}|\boldsymbol{W}, \boldsymbol{\Theta})$ is now assumed to be dependent only on the acoustic parameters, while the probability of $P(\boldsymbol{W}|\boldsymbol{\Theta})$ is now assumed to be dependent only on the language parameters. Then, we can then rewrite the Eq. (2.2) to a new form:

$$\boldsymbol{W} = \arg\max_{\boldsymbol{W}}, P(\boldsymbol{O}|\boldsymbol{W}, \boldsymbol{\Theta}_{AM})P(\boldsymbol{W}|\boldsymbol{\Theta}_{LM}). \tag{2.3}$$

In this form, the acoustic observations are determined solely by the acoustic parameters and the probability of words is determined solely by the language parameters. This expansion allowed us to define the statistical recognizer on a basis of two separate models.

- $P(\boldsymbol{O}|\boldsymbol{W}, \boldsymbol{\Theta}_{AM})$ - determines the probability of acoustic observation given the set of acoustic parameters combined in the **Acoustic model**.
- $P(\boldsymbol{W}|\boldsymbol{\Theta}_{LM})$ - determines the probability of words given the set of language parameters combined in the **Language model**. As this thesis is not orientated on language modelling, the LM will be discussed only very briefly further in the text.

If we assume that each word can be modelled by a sequence of subword units, then this scheme can be fragmented even further. This approach is practically always used as the number of words in any language is too high to model. The most common set of phonetic units consists of either simple phones or their context-dependent variants biphones, triphones etc. The problem of estimating the parameters for the acoustic model and the language model comprises two great research areas of speech recognition. The

following text will focus on signal preprocessing and feature extraction algorithms and on the AM training algorithms, whose purpose is to estimate the $\boldsymbol{\Theta}_{AM}$ parameters.

2.2 Feature Extraction

The process of feature extraction involves transforming the speech signal into the form that is more suitable for acoustic modelling and decoding. These features should provide good discriminability between phonetic units, the vector has to be compact so that the whole extraction is fast and the features should be robust against speaker and environmental variability. However, the latest trend for recognizers based on neural nets is to use short snipets of raw speech which usually several hundred $[ms]$ long. This approach was shown to bring comparable or even better results in certain recognition tasks such as distorted speech recognition [2, 3]. However, a closer look at the first few layers reveals, that these nets effectively emulate known extraction schemes [4] and thus a general overview of the standard extraction techniques is helpful to understand the underlying principles.

2.2.1 Speech Production Model

Since practically all popular parametrization schemes for ASR exploit the speech production and perception knowledge, this section will begin with introducing the speech production model. The proposed compensation algorithm for lossy compressed speech recognition exploits this model as well as thus its beneficial to have a point of reference.

Speech production is a process in which the stream of air exiting the lungs passes through the glottal area (vocal folds), enters the vocal tract area (which consists of the laryngeal, the oral and the nasal cavity) and finally exits through the nose and the mouth as sound. The distinctions between different articulated sounds (phonemes) are determined by different parameters of all components. The vocal tract can be fully described as a tube with a time-varying cross-section [5] that behaves like a passive resonator with multiple resonating frequencies. These frequencies, which are called formants, are determined by the shape of the vocal tract. The voice source characteristics are more complicated. Vocal folds modulate the airflow exiting the lungs and produce the voice source signal. If the vocal folds are fully open, the air stream is characterized as a turbulent airflow which produces unvoiced consonants. If the vocal folds are tightly stretched, the air stream builds up under the glottis and the folds open and close periodically. This mode produces a quasi-periodic source signal and creates voiced phones (vowels and voiced consonants).

The block scheme in Figure 2.2 describes this process as a linear time invariant (LTI) system. In this model, the voice source signal represents the driving signal $x[n]$ and the vocal tracts characteristics are modelled as an all-pole filter with the impulse response $h[n]$. This model can be expanded further by introducing the lip radiation filter with an

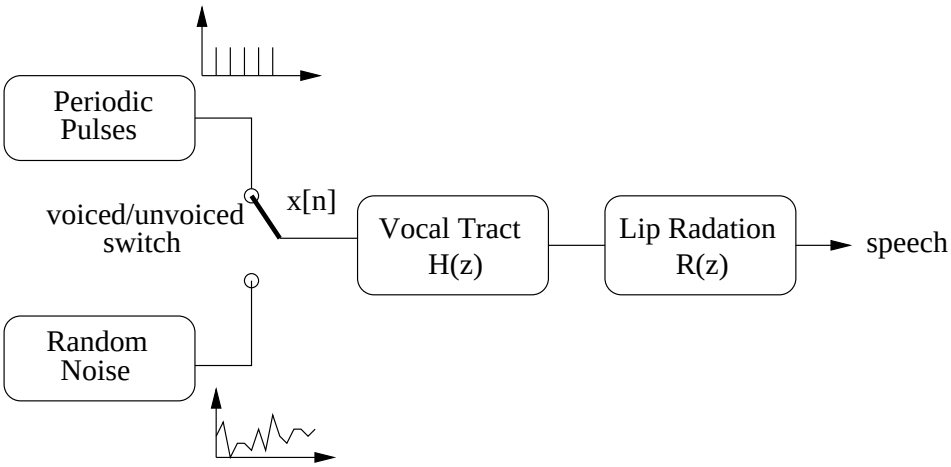

Figure 2.2: The linear time-invariant model of speech production

impulse response $r[n]$. The speech $y[n]$ is then written as:

$$y[n] = x[n] * h[n] * r[n], \tag{2.4}$$

and frequency responses of vocal tract and lip radiation filters have the form of:

$$H[z] = \frac{G}{1 - \sum_{k=1}^{N} a_k z^{-k}}, \qquad R[z] = R_0(1 - z), \tag{2.5}$$

where G is the gain, N is the order of the filter and a_k are the coefficients of the filter. Further analysis of $H(k)$ and $R(k)$ reveals, that the vocal tract dominates as a primary source of information about the content. As a consequence, the standard features are designed to remove the voice source parameters of the the speech signal and model only the vocal tract characteristics.

2.2.2 Cepstral-based Features

The most popular features currently used are designed to model the frequency envelope of above mentioned vocal tract and lip radiation filters using the short-time Fourier transform (STFT). Figure 2.3 illustrates the block schemes for Mel-Frequency cepstral coefficients (MFCC) [6] and perceptual linear prediction cepstral coefficients (PLP) [7].

The process of their extraction is similar to a certain degree. The MFCC algorithm first splits the speech into short, quasi-stationary frames with an overlap. Then the weighting window (usually Hamming) is applied to attenuate the spectral leakage. In the next step, the energy spectrum is computed for windowed frames using the formula for STFT. The Mel filter bank is then applied to integrate energy in each critical frequency band. Each filter has a magnitude frequency response that is triangular in shape and their central frequencies are placed equidistantly on a non-linear mel-frequency axis f_{mel} defined by the Eq. (2.6). This characteristic emulates the non-linear frequency resolution

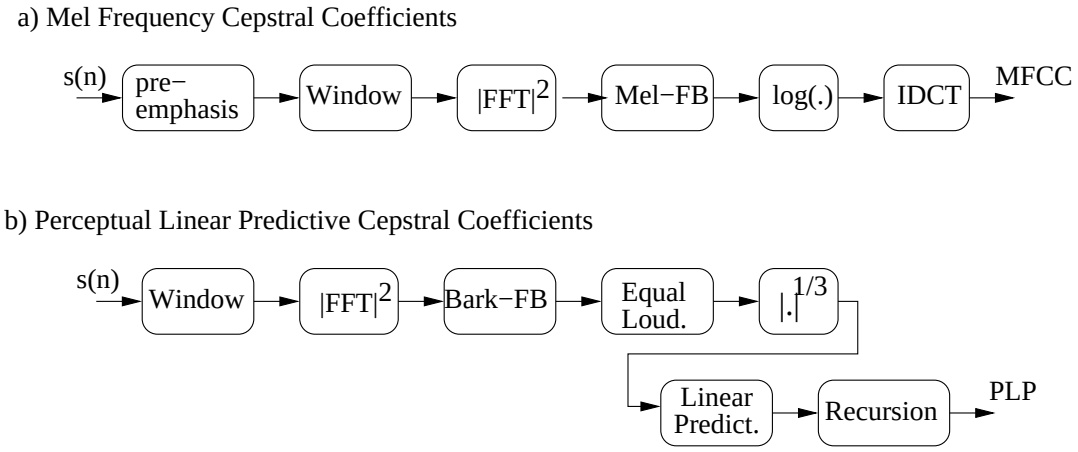

Figure 2.3: The parametrization scheme of MFCC and PLP cepstral features

of the hearing. Finally, the logarithm is applied and the discrete cosine transform is used to extract MFCCs. However, the current trends in feature extraction also favour mel-scale log-filter bank features. Their computation is identical to standard MFCCs, but the process is truncated before the application of the cosine transform. Multiple works demonstrated their superior quality over MFCCs [8, 9], especially in conjunction with an AM based on neural networks.

$$f_{mel} = 2595 \log_{10}\left(1 + \frac{f_{Hz}}{700}\right). \tag{2.6}$$

$$f_{bark} = 6 \ln\left(\frac{f_{Hz}}{600} + \sqrt{\left(\frac{f_{Hz}}{600}\right)^2 + 1}\right). \tag{2.7}$$

The PLP computation begins with the same steps of segmentation, windowing and computing the energy spectrum and then a trapezoidal filter bank is applied. The filters are placed equidistantly on the bark-frequency scale f_{bark} defined by the Eq. (2.7). The next steps are different as PLPs were designed to emulate sound perception more closely. The equal loudness block simulates the perceived intensity and the cubic root compression transforms the intensity into loudness. The energy spectrum is represented by the linear prediction coefficients which are then transformed into the PLP cepstral coefficients.

2.2.3 Temporal Context Information

Described features can accurately capture the spectral envelope in a short window in which the signal is assumed to be static, but the temporal information about the neighbouring content is usually added to the static features. The standard approach is to add 1^{st} and 2^{nd} order dynamic parameters as the dynamic and acceleration cepstral

features can be more robust against convolutional distortions. The dynamic parameters are computed using the formula

$$\Delta c_k[n] = \frac{\sum_{m=1}^{M} m(c_{k+m}[n] - c_{k-m}[n])}{2\sum_{m=1}^{M} m^2}, \tag{2.8}$$

where $c_k[n]$ is the kth cepstral coefficient and m is the length of the derivation window which is usually set to 2. The acceleration coefficients are computed by reusing the formula (2.8) and substituting the static coefficients for dynamic ones from the previous step. More recent solutions favour concatenating several neighbouring static feature vectors into a single high-dimensional feature vector. A factorization method in then applied in order to reduce the vector dimensionality and to decorrelate the features. The context length varies highly as there is no clear consensus for its proper value. If we assume the average vowel duration in a fluent, continuous speech is between 100 to 200 ms on average [10], and the segmentation step was set to 10 ms, then we arrive at the conclusion that 7 preceding and following vectors provide sufficient temporal context. However, these values are highly speaker, language, dialect and context dependent. A second thing to consider is that vowels are generally longer in duration than consonants. Another solution is to derive dynamic MFCCs directly from dynamic spectrum [11].

The common approach is to reduce vectors' dimensionality by the application of a linear discriminant analysis (LDA) [12], heteroscedastic linear discriminant analysis [13] or some other factorization technique [14]. The principle of these techniques is to transform the input feature vector into the space of output vectors and to truncate it at N principal components. All of these methods also serve the purpose of decorrelating the features in a vector, which has been also shown to improve the performance of standard ceptral features in the presence of noise [15, 16, 17].

2.3 GMM-HMM Acoustic Model

The GMM is a statistical generative model that can very effectively model the *static* cepstral features, while HMM is a statistical model that is able to model the *temporal dynamics* of speech. Thus, the fusion of these two components creates a model capable of describing both spectral and temporal characteristics of the speech. The use of GMM-HMM for speech modelling involves selecting an appropriate structure. This decision is usually done expertly and depends on the type of modelled speech units.

Figure 2.4 illustrates a classical composite GMM-HMM model that is used in ASR. The typical GMM-GMM structure for subword units consists of 3 emitting states $\{s1, s2, s3\}$, the *entry* and *exit* states. The model is fully described by the transition matrix $\mathbf{A} = \{a_{ij}\}$, which defines the probability of moving from state i to state j, and the emitting functions

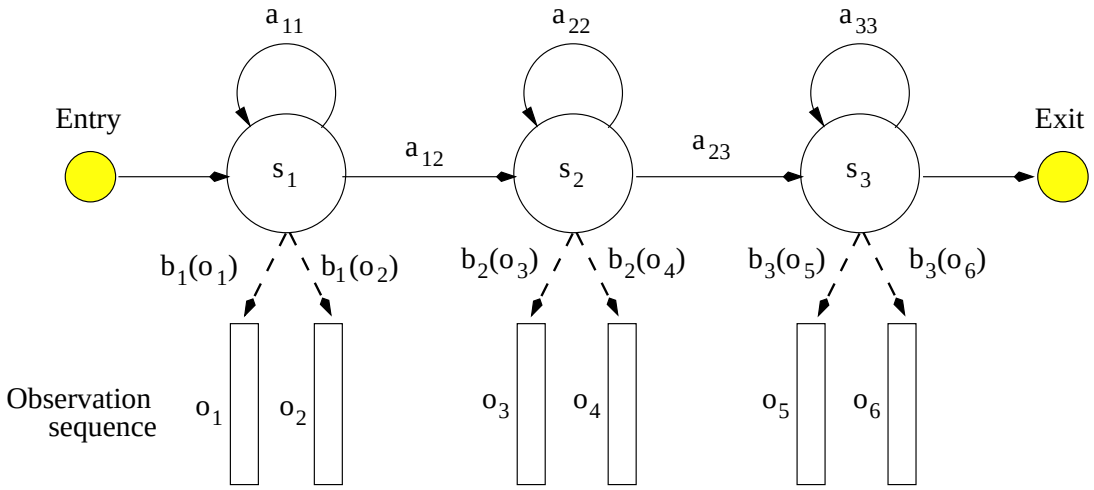

Figure 2.4: A classical left-to-right HMM with 3 emitting states s_i defined by its state transitional probabilities a_{ij} and observation emission probabilities $b_i(\mathbf{o}_t)$

$b_i(\boldsymbol{o}_t)$. The model usually lacks backward transitions as only forward and state-repeating transitions are allowed. Each state is assigned its emitting function $b_i(\boldsymbol{o}_t)$ which estimates the probability of the observation vector $\boldsymbol{o}_t$ being generated by the state i and can be expressed as

$$b_i(\boldsymbol{o}_t) = \sum_{m=1}^{M} c_m \mathcal{N}\left(\boldsymbol{o}_t; \mu_{im}, \boldsymbol{\Sigma}_{im}\right). \tag{2.9}$$

The set of acoustic model parameters $\boldsymbol{\Theta}_{AM} = \{c_m, \mu_{im}, \boldsymbol{\Sigma}_{im}\}$ are the weight, the mean and the covariance matrix of a multivariate normal distribution $\mathcal{N}(\mu_{im}, \boldsymbol{\Sigma}_{im})$. Given the defined GMM-HMM, the probability of generating the state sequence $\mathbf{S} = s_1, s_2, ..., s_k$ is dependent only on the transition probabilities and the observation probability for frame $\boldsymbol{o}_t$ is dependent only on the emission probability $b_i(\boldsymbol{o}_t)$ of the corresponding state i. The total likelihood of generating the observed sequence of acoustic features $\boldsymbol{O} = \{\boldsymbol{o}_1, \boldsymbol{o}_2, ..., \boldsymbol{o}_T\}$ is then expressed as

$$P(\boldsymbol{O}|\boldsymbol{\Theta}_{AM}) = \sum_{s_1,...,s_k} \prod_{t=1}^{T} a_{s_t|s_{t-1}} b_{s_t}(\boldsymbol{o}_t), \tag{2.10}$$

where $a_{s_t|s_{t-1}}$ represents the state transition probability $p(s_t|s_{t-1})$. The described acoustic model is fully defined by the set of acoustic model parameters $\boldsymbol{\Theta}_{AM}$ that need to be inferred from the training data. Several learning schemes already known in the field of the machine learning have been adopted for this purpose, while many others have been proposed specifically for ASR. Historically, the most common method for the AM parameters' estimation was based on Maximum Likelihood estimation (MLE). However, this conventional approach has several drawbacks, some of which stem from the unmet assumptions for HMM when used for modelling the human speech, others stem from the assumptions of the MLE itself. This fact can cause the MLE to yield suboptimal results

in terms of classification accuracy. As a result, discriminative training algorithms have taken over as the principal training algorithms. Their main advantage is the fact that they don't make any assumptions about the distribution of training data.

2.3.1 Generative Training

A generative training algorithm based on MLE was implemented for ASR systems using a very efficient Baum-Welch algorithm. The algorithm is derived from the Expectation Maximization (EM) algorithm that aims is to maximize the likelihood of training data generation using the given model. The whole training procedure can be divided into two essential steps; the E-step and the M-step. The goal during the E-step is to estimate the likelihood of generating observed data given the current set of model parameters and new model parameters are estimated during the M-step. Its main advantage is the fact that if the training data truly belong to the class of presumed distribution, the generative training leads to optimal parameter estimation [18]. Given the set of training observation $\boldsymbol{O}$, their corresponding transcriptions $\boldsymbol{W}$ and the set of unknown acoustic model parameters $\boldsymbol{\Theta}_{AM}$, the MLE approach attempts to maximize the function

$$F_{MLE}(\boldsymbol{\Theta}_{AM}) = \sum_{t=1}^{T} logP_{\boldsymbol{\Theta}_{AM}}(\boldsymbol{o}_t|M_t), \tag{2.11}$$

where M_t is the model corresponding to the correct transcription w_t. The maximization of the likelihood function is generally done in practice by mentioned Baum-Welch algorithm, which is repeated in cycles until a larger than set difference between "old" and "new" parameters is achieved. The new model parameters are then re-estimated using the formulas which can be found in [19]. However, it is important to note that Baum-Welch algorithm leads only to the local maximum of $P(\boldsymbol{O}|\boldsymbol{\Theta}_{AM})$, which means that the training results are dependent on the initial conditions.

2.3.2 Discriminative Training

The principle of discriminative training (DT) techniques, in comparison to the generative ones, is the effort to minimize the recognition error directly instead of maximizing the observation likelihood. This is generally achieved by formulating an objective function which is directly relevant to the actual classification and is able to "discriminate" against the model parameters which are likely to confuse the classification. This can be expressed in the form of multiple competing hypotheses, when both the correct and incorrect classifications are used for actual training.

However, DT suffers from a set of problems in conjunction with the ones already mentioned for MLE. The addition of incorrect classifications to the learning criteria function for DT expands the original homogeneous polynomial criterion function to the rational one. Authors in [20] proposed a new optimization method called Extended Baum-Welch,

which was later on [21] successfully extended for continuous density HMMs. The extensive computational load, the second major problem of DT, was satisfactory resolved by the use of lattice-based training framework [22]. Another problem of DT in general is a poor test data generalization, when the discriminative models tend to work well on training data but relatively poorly on unseen test data. One of the possible solution is to employ an AM scaling factor κ to increase the amount of confusable training data or use a "weak" unigram LM for lattice generation.

The most prevalent DT methods include entropy-related Maximum Mutual Information (MMI), boosted Maximum Mutual Information (bMMI), Large Margin Estimation, Conditional Maximum Likelihood or classification-based Minimal Classification Error, Minimal Word Error (MWE) and Minimal Phone Error (MPE) [23, 24].

Maximum Mutual Information

MMI estimation was first proposed from the point of view of information theory [25], when the goal of the parameter estimation was to maximize the mutual information $I(\boldsymbol{O}, \boldsymbol{W})$ between observations $\boldsymbol{O}$ and their transcriptions $\boldsymbol{W}$. It was later proved [26] that MMI outperforms the MLE if observation data has different than assumed distribution. This is naturally true for any real-world signals. Given the same model parameters and observations as for MLE, the MMI criteria has the form of:

$$F_{MMI}(\boldsymbol{\Theta}_{AM}) = \sum_{t=1}^{T} log \frac{P_{\boldsymbol{\Theta}_{AM}}(\boldsymbol{o}_t|\boldsymbol{M}_t)P(\boldsymbol{w}_t)}{\sum_{\hat{\boldsymbol{w}}} P_{\boldsymbol{\Theta}_{AM}}(\boldsymbol{o}_t|\boldsymbol{M}_{\hat{\boldsymbol{w}}})P(\hat{\boldsymbol{w}})}\,, \tag{2.12}$$

where the numerator is the traditional MLE and the denominator is the summation over all possible word sequences (correct and incorrect) defined in the recognition task. The $\boldsymbol{M}_{\hat{\boldsymbol{w}}}$ is the composite model corresponding to the word sequence $\hat{\boldsymbol{w}}$, $P(\boldsymbol{w}_t)$ and $P(\hat{\boldsymbol{w}})$ represents the word sequence probability given by a stochastic language model. The maximization policy is evident from the form of the objective function; the numerator representing the correct word hypothesis must increase (the same as ML), while the denominator representing any possible words hypotheses must decrease. The MMI algorithm deals with the generalization problem by interpolating ML and MMI criteria functions, which is known as H-criterion [27] or I-Smoothing [22].

Minimum Phone/Word Error

The Minimum Word Error was first proposed in [28], where the focus was on minimizing the estimation of training set errors. The MWE was thus defined to maximize the expected

word accuracy with the objective function of:

$$F_{MWE}(\boldsymbol{\Theta}_{AM}) = \sum_{t=1}^{T} log \frac{\sum_{\boldsymbol{w}} P_{\boldsymbol{\Theta}_{AM}}(\boldsymbol{o}_t|M_{\hat{\boldsymbol{w}}})P(\hat{\boldsymbol{w}})RawAccuracy(\hat{\boldsymbol{w}})}{\sum_{\hat{\boldsymbol{w}}} P_{\boldsymbol{\Theta}_{AM}}(\boldsymbol{o}_t|M_{\hat{\boldsymbol{w}}})P(\hat{\boldsymbol{w}})}, \tag{2.13}$$

where the $RawAccuracy(\hat{\boldsymbol{w}})$ is the measure of the number of correctly transcribed words in the word sequence $\boldsymbol{w}$. The MWE function (2.13) gives the weighted average of correct words over all possible word sequences $\hat{\boldsymbol{w}}$, which is in fact the metric used to estimate the standard error rate. If $\kappa \to \infty$, the maximization of MWE criterion leads the minimization of the error rate. The Minimum Phone Error uses the same objective function as Eq. (2.13), but the formula is defined for the phone error instead. The details on $RawAccuracy(\hat{\boldsymbol{w}})$ computation can be found in [28]. In addition to already mentioned problems, the application of MWE or MPE can easily result in over-training, thus the I-Smoothing was proposed and shown to be necessary in order for MPE to outperform MLE.

Boosted-Maximum Mutual Information

The standard MMI objective function got later extended in [29] by introducing a term similar to one that is used in Minimum Phone Error. In bMMI the objective function has the form of:

$$F_{bMMIE}(\boldsymbol{\Theta}_{AM}) = \sum_{t=1}^{T} log \frac{P_{\boldsymbol{\Theta}_{AM}}(\boldsymbol{o}_t|M_t)P(\boldsymbol{w}_t)}{\sum_{\hat{\boldsymbol{w}}} P_{\boldsymbol{\Theta}_{AM}}(\boldsymbol{o}_t|M_{\hat{\boldsymbol{w}}})P(\hat{\boldsymbol{w}}) \exp(-b * RawAccuracy(\hat{\boldsymbol{w}}))}, \tag{2.14}$$

where b is the boosting factor from which the technique got its name. The purpose the term is to boost the likelihood of the sentence containing errors and thus to produce more confusable data.

2.4 DNN-HMM Acoustic Model

Figure 2.5 illustrates an example of a hybrid DNN-HMM system with a feed-forward architecture, where the temporal dynamics of speech is modelled by HMM and the DNN is used to model the observation probabilities within a static frame. The actual structure of DNN in the figure is composed of an input, an output and 4 hidden layers with different number of units in each layer. A unit j in each hidden layer employs a non-linear activation function to map the total sum of inputs from the preceding layer to the output that is sent to the next layer. The unit input x_j at the current layer is computed as a weighted linear combination as per the Eq. (2.15), where b_j is the bias, y_i is the output from the unit i in the preceding layer and w_{ij} is the weight of a connection from unit i to unit j. The most common activation function include the logistic (sigmoid) function, hyperbolic tangent or

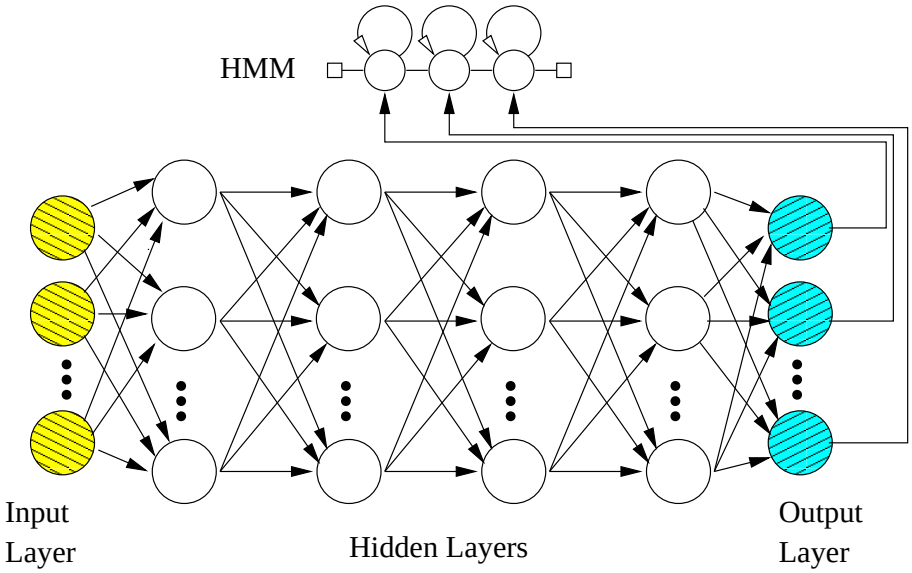

Figure 2.5: An example of a DNN-HMM with an input layers, four hidden layers and an output layer.

rectified linear function. The recognition experiments in this thesis were conducted with a DNN-HMM system with a sigmoid activation function (2.15). The output value of j^{th} unit represents the probability $P_{dnn}(j, \mathbf{o})$ that the observation vector $\mathbf{o}$ belongs to class j which can be done by using the "soft-max" function as per Eq.(2.16).

$$x_j = b_j + \sum_i y_i w_{ij} \quad , \quad y_j = \frac{1}{1 + e^{-x_j}} \, . \tag{2.15}$$

$$P_{dnn}(i|\mathbf{o}) = \frac{e^{x_j}}{\sum_{j=1}^{C} e^{x_j}} \, . \tag{2.16}$$

Currently, the most popular DNN recognition frameworks are built to model HMM states of context dependent phones, called senons. This approach, also called continuous density DNN-HMM, is very similar to previous state-of-the-art HMM-GMM systems which contributed heavily to its rapid development as large number of previously applicable processes and methods were easily transferable to this newer framework. In the DNN-HMM, the output layer of DNNs is trained to estimate the conditional state posterior probabilities $p(s_t = i|\mathbf{o}_t)$ given the observation $\mathbf{o}_t$. The most popular DNN training method is to employ the error back-propagation algorithm in conjunction with a gradient descent optimization method. However, the algorithm suffers from the problem of vanishing or exploding gradient that occurs mostly due to the practise to initialize the parameters randomly. This problem has been effectively solved by introducing improved deep learning algorithms. This section has only touched on the problems of DNN training and has not touched on the problems of DNN feature extraction optimization. A more thorough description on neural networks for ASR can be found in [30].

2.5 Language Model and Decoding

The purpose of the LM is to estimate the probability of generating the hypothesised word sequence $P(\boldsymbol{W})$, which can be further decomposed using the chain rule as:

$$P(\boldsymbol{W}) = \prod_{i=1}^{n} P(w_i|w_1, ..., w_{i-1}), \tag{2.17}$$

where $P(w_i|w_1, ..., w_{i-1})$ is the probability that word w_i is spoken given the previously uttered word sequence $w_1, ..., w_{i-1}$. The past word sequence is also called the history. The purpose of the language model is to provide the recognizer with an adequate estimate of $P(w_i|w_1, ..., w_{i-1})$. However, it is practically infeasible to create a model with large history given all possible word sequences. As a consequence, the current state-of-the-art approach is to employ *n-gram* models which limit the history length down to $n-1$ number of words. The recognition experiments presented in this thesis were done using a $bigram$ LM, which simplifies the formulated probability to the form of:

$$P(\mathbf{W}) = P(w_1) \prod_{i=1}^{n} P(w_i|w_{i-1}). \tag{2.18}$$

The optimization of LM was not part of my research topics and thus it is not discussed further. The final component of any speech recognizer is the decoding block, which combines the probability scores of acoustic and language models and outputs the most likely word sequence $\hat{\mathbf{W}}$. Although it is computationally infeasible to search the whole recognition space for the optimal solution, it can still be can be very effectively solved by utilizing the dynamic programming and $Viterbi$ algorithm. Their application in speech recognition greatly simplifies the decoding process by utilizing the optimality principle which postulates, that the optimal path through a directed graph is equivalent to taking optimal partial paths between the nodes. The optimality principle ensures that the likelihood for each state at each stage t can be computed by means of a simple recursion. Besides the described recursion, the Viterbi algorithm requires additional steps of recursion initialization, termination and path-backtracking. Another advantage of the algorithm is that it does not need to keep track of all partial paths leading to stage $t + 1$. It is important to realize that the described procedure can be applied to both GMM-HMM as well as DNN-HMM architectures.

2.6 ASR's Robustness

The current ASR systems perform very well under clean acoustic conditions and with high quality signals. However, when the recordings come from a noisy environment or the speech signal is distorted on its way from a microphone to the ASR engine, the system performance can drop significantly. The types of conditions which can result in

performance drops are numerous, but the the ones explored within this thesis are as follows.

- **Far microphone**: In order to provide a natural speech input for voice controlled devices in smart-homes and offices, the common practice is to make use of distant microphones with omnidirectional characteristics. These microphones are usually embedded in devices themselves or in the walls or a ceiling and lead to the presence of various kinds of additive and convolution distortions such echoes and reverberations. Also, an attenuation of distant speech is rather high. The methods for robust far microphone recognition studied in this thesis include methods of noise subtraction, feature vector normalization and robust acoustic modelling.
- **Driving Car** - While the voice controlled devices in intelligent cars (i.e. a navigation) also rely on the use of omnidirectional middle-distance microphones, the acoustic conditions are very different from the smart-homes. The environment is practically reflectionless and thus the recordings contain very little convolution distortion. On the other hand, both the running engine and the aerodynamic noise of the air introduce strong additive noises. Each has its own specific spectral characteristics and requires a special tailoring to compensate. The methods for robust driving car recognition studied in this include methods of noise subtraction, feature vector normalization and robust acoustic modelling.
- **Lossy Compression** - The perceptual audio coders contain a psychoacoustic block that exploits the imperfection of human hearing in order to code the speech signal into a low-bitrate digital stream. Certain temporal and spectral parts of the signal that are considered inaudible and therefore redundant are removed. Unlike the previously discussed situations, compression introduces unwanted information and removes the desired information from the signal at the same time. As a result, this process introduces multiple non-linear distortions which are hard to describe using the standard signal processing theory, but are known to severely degrade the performance of ASR systems. The methods for robust lossy compression recognition include methods of noise addition, matched training and robust acoustic modelling.

2.6.1 Robust Front-End Processing

The robustness of ASR can be solved at front-end processing level at first and there are generally three options [31]. The preprocessing algorithms which modify the signal before the actual parametrization, robust feature extraction schemes or methods which modify the already extracted features. The preprocessing algorithms are usually designed to be independent of used features and their purpose is to target specific distortions, which limits their usage for other conditions. On the other hand, robust parametrizations are often just extensions or modifications to the existing algorithms and their deployment is not conditioned so strictly by the presence of a specific distortion. Finally, the methods which modify the extracted features are largely independent of the used features and also the conditions.

Spectral Subtraction

Spectral Subtraction represents one of the oldest, yet still used, method for compensating additive noise. The principle idea is to estimate the spectral characteristics of noise and to subtract it directly from a signal in the frequency domain [32]. There are two main approaches on how to estimate noise characteristics : employ a voice activity detector and estimate noise spectrum from silence segments [33] or use the spectral minims for estimation [34]. The study on spectral subtraction done in [35] showed that its application can actually decrease ASR's performance on clean and slightly noisy speech due to the introduction of non-linearities, but it also reported improvements for artificially added car noise for SNR $\leq$ 6 dB. I use within this thesis the extended spectral subtraction [36] (ESS) which is an iterative version of spectral subtraction that works without the need of a voice activity detection and uses the principle of deriving the noise spectrum by tuning the gain of a matched Wiener filter. This method is based on the initial assumption that noise changes its characteristics more slowly than speech does. As a result, this algorithm works well for background stationary noises or for the noise with slow changes. Significant improvements from using ESS for removing noises introduced in a public place (a shop) for SNR $\leq$ 20 dB were shown in [37]. A comprehensive summary on various modifications to this technique was done in [38] and the overall conclusion was that the technique can bring both subjective and objective performance improvements in case the initial conditions about the noise characteristics are met.

Cepstral Mean and Variance Normalization

Cepstral Mean Normalization (CMN) is a well established technique for robust speech recognition. The principle of is based on the assumption that the average cepstrum of real speech $\overline{\mathbf{c}}_{\mathbf{x}}$ that contains stationary convolutional distortions added by a channel h can be expressed in the form:

$$\overline{\boldsymbol{c}}_x = \frac{1}{N}\sum_{i=1}^{L-1} \boldsymbol{c}_s[i] + \boldsymbol{c_h} = \overline{\boldsymbol{c}}_{\boldsymbol{s}} + \boldsymbol{c}_h, \tag{2.19}$$

where $\overline{\boldsymbol{c}}_s$ is the average cepstrum of clean speech computed from L number of segments and $\boldsymbol{c}_h$ is the cepstrum of the channel. If we further assume that $\overline{\boldsymbol{c}}_s \to 0$ if $L \to \infty$, then we can approximate the average cepstrum of real speech with cepstrum of the channel, which means that $\hat{\boldsymbol{c}}_x \approx \boldsymbol{c}_h$. Thus, the aim of CMN can be easily explained as a process of removing convolutional distortion by subtracting its contribution $\mathbf{c}_h$ in cepstral domain. Although it is a fairly simple method it has been proved to provide robustness against the environmental and channel distortions and speaker variability. Cepstral mean and variance normalization (CMVN) is technique that normalizes the mean and variance of cepstra to give zero mean and unit variance. Even though there is no precise theoretical explanation for normalizing the cepstral variance, it is know to remove additive noises and to normalize the speaker variability. If the statistics are accumulated over time from a suitably long window, this approach is often called cepstral mean subtraction (CMS).

The application of CMS for telephone band recognition was done in [40]. The study used 4 s sliding window and reported about 15% relative improvement. Significant relative improvements of 25.5% for noisy car recognition were reported in [41].

2.6.2 Robust Acoustic Modelling

The robustness of ASR can be also solved at the acoustic modelling level. The most common approach is to use an AM adaptation technique whose task is to adapt the AM parameters or transform feature vectors prior to decoding to specific environmental conditions with "a little" amount of adaptation data. The most common methods are Maximum A-Posteriori Probability (MAP), Maximum Likelihood Linear Regression (MLLR) and feature Maximum Likelihood Linear Regression (fMLLR). The second option is to use the described adaptation techniques during the training process to reduce the variability in the training data. Although adaptive training is concerned with removing all variability present in the training set, a method called noise adaptive training [42] aims to obtain a "pseudo-clean" AM that specifically lacks the acoustic variability. The third approach is based on increasing the overall robustness AM. The conventional system based on GMM can contain several hundred thousands mixtures. As a result, subspace Gaussian Mixture Model (SGMM) has been proposed as an alternative approach in which the model parameters are typically initialized from the clustered Universal Background Model (UBM) and then shared. The result is a situation when a trained SGMM system has typically less parameters than a standard GMM system [43]. Likewise, DT has been studied for noisy speech recognition as well and the authors have concluded that its usage can increase the overall robustness of the system. However, it was also shown that the application of DT suffers from poor generalization and is thus not always applicable for distorted speech, especially if the acoustic conditions in the training set differ greatly from the test set. Despite this obvious disadvantage, the improved AM quality often outweighs the generalization problem.

Maximum A-Posteriori Adaptation

The MAP adaptation [44] is based on the definition of an ASR system as a Bayesian classifier with a zero-one loss function. In paper [45], the authors introduced the re-estimation formulas for HMMs by addressing the problem of a priori distributions for the HMM parameters, which they stated can be adequately represented as a product of Dirichlet and normal-Wishart densities. Given the vector of model parameters $\boldsymbol{\Theta}$, the set of observation vectors $\boldsymbol{O}$, and using the well-known Bayes' theorem, the formula for MAP estimated vector of model parameters $\boldsymbol{\Theta}_{MAP}$ can be written as:

$$\boldsymbol{\Theta}_{MAP} = \arg\max_{\boldsymbol{\Theta}} P(\boldsymbol{O}|\boldsymbol{\Theta})P(\boldsymbol{\Theta}), \tag{2.20}$$

where the $P(\boldsymbol{\Theta})$ represents the a priori information about parameters probability distribution, also known as the *informative prior*. To solve the Eq. (2.20), it is assumed

that $\boldsymbol{O}$ is the set of independent observations and the parameters are from the assumed distributions. The new model parameters $\boldsymbol{\Theta}_{MAP}$ can be estimated by the standard EM algorithm. It is important to note that the speaker-independent model is used to get a priori probability distribution. This means, that less data is needed to estimate the new model parameters when compared to standard re-training. In case the information prior is not taken into account, or is not present, the MAP estimation takes the form of standard ML estimation. Otherwise the new model parameters are estimated as the weighted average of a priori information and ML estimation. One disadvantage of MAP adaptation is the large amount of data needed to satisfactory update the old parameters. This problem becomes especially pressing in the case of complex AMs (e.g. triphone-based AMs) with a large number of parameters.

Maximum Likelihood Linear Regression Adaptation

The MLLR adaptation method uses the maximum likelihood estimation for finding the optimal transformation to fit the general model on the adaptation data [46]. The solution to data problem for MAP was introduced in [47], where the author proposed to cluster the parameters for similar models into groups (regression classes) and to find a linear transformation for the whole group. In a strict sense of speaking, the MLLR can be applied both in constrained and unconstrained version and on the model parameters or feature vectors, summarized in [48]. The constrained MLLR (CMLLR) uses the same transformation matrices for all model parameters and can be applied to both the AM or feature vectors. If CMLLR is applied to feature vectors, it is usually referenced as the feature MLLR. The formula introduced for mean vectors $\hat{\boldsymbol{\mu}}$ update, using the previously stated conditions is:

$$\hat{\boldsymbol{\mu}} = \boldsymbol{A}_{\Omega}\boldsymbol{\mu} + \boldsymbol{b}_{\Omega}, \tag{2.21}$$

where the $\boldsymbol{A}_{\Omega}$ is the transformation matrix and $\boldsymbol{b}_{\Omega}$ is the bias vector, both for the regression class Ω. The formula for covariance matrix is:

$$\hat{\boldsymbol{\Sigma}} = \boldsymbol{H}_{\Omega}\boldsymbol{\Sigma}\boldsymbol{H}_{\Omega}^{T}, \tag{2.22}$$

where now $\boldsymbol{H}_{\Omega}$ is the desired transformation matrix. Standard MLLR was studied in [49], where the authors used clean trained AMs for recordings from a car environment. The recognition task consisted of a simple digit recognition and showed that using two regression classes (speech vs. noise) in MLLR can yield up to 87.1% *WERR*. A slightly different approach was take in [50], where the authors experimented with multi-channel-based MLLR and MAP adaptations. The authors evaluated the performance of AMs for each feature stream (static, dynamic, acceleration) separately and adapted the whole vector based on the best-performing one. Their evaluation set contained artificially noisy recordings with various SNR levels. This splitting approach brought 22% absolute improvement while the standard MLLR+MAP approach achieved only 8% improvement. This work further demonstrated that static parameters are more prone to degradation due to the presence of noise than the dynamic ones. The authors in [51] investigated

the performance of MLLR in a distant speech recognition and reported 15% *WERR*. Yet another example of MLLR on Aurora2 database can be seen in [52], where the authors experimented with pooling data into either a global cluster or SNR specific clusters. Interestingly, the SNR specific MLLR outperformed the generic MLLR in 4 out of 8 environments (car, street, airport, station). A slight modification to the existing CMLLR was proposed in [53]. The authors called it noisy CMLLR and evaluated its performance against the standard MLLR and CMLLR on artificially noisy recordings which contained operations room and car noises. The CMLLR adaptation proved to outperform the standard MLLR while the proposed noisy CMLLR achieved even slightly better results. The CMLLR achieved better *WER* for the car and 20 dB operation room environments but lagged behind for 14 dB room subset. Another modification to the standard MLLR was presented in [54] where the authors dealt with adapting a noisy AM to another noise types from Aurora4 database. The proposed method consisted of MLLR and CMLLR adaptations in conjunction with uncertainty decoding and the authors reported 6% absolute improvement over the SPLICE UD reference.

Speaker Adaptive Training

Speaker-adaptive training (SAT) is based on the assumption that the variability in the training data is caused not only due to the phonetic content, but also by the variability among speakers and environmental conditions. The purpose of the SAT is to remove this variability from the the SI model by the application of the before-mentioned adaptation techniques during the training process and to create a more general (canonical) AM. Thus, this is a completely opposite process from the classical adaptation which creates a SA model. However, the canonical contains very little information about speakers or acoustic conditions and thus it is necessary to adapt it prior to decoding. The MLLR and fMLLR adaptations are used most often for this purpose and my SAT setup made also use of fMLLR.

Discriminative Training

The authors in [55] studied the MMI trained models on isolated and connected digits recognition tasks in noisy environments and reported approx. 3% improvements for both tasks over the standard MLE. This work also demonstrated that MMI training can be used for a key word-spotting task as the article reported approx. 7.5% error rate reduction. Another results with DT models for Aurora2 corpus have been presented in [56], where the authors used MMI training scheme for the same multi-condition training task. Their work reported up to 11% relative improvements.

The study on MWE and minimum divergence (MD) training algorithms for both clean and multi-condition training for Aurora2 set was presented in [57]. This work reported 41% and 35% relative improvements of MWE and MD trained models for the clean task. They also demonstrated that the increased train-test mismatch lowers the contribution of DT. The highest *WERR* of 45.45% was achieved with MWE algorithm and digits

recognition task for the 20 dB SNR test set. However, the relative improvements for 0 db and -5 dB evaluation sets were only 6.25% and 3.01% respectively, which further demonstrates the limits of DT for distorted speech.

A more realistic scenario of DT models working with real-life recordings from a domestic environment was presented in [58]. This work evaluates the performance of MMI and bMMI trained AMs for 2^{nd} CHiME challenge. The evaluation tasks consisted of multiple subsets with different SNR levels -6, -3, 0, 3, 6, 9 dB and the authors studied the performance of both MFCC and PLP features. The best absolute improvement of 5.52% was achieved for MFCC features and bMMI criteria. The improvement for PLP features was only 3.75% using the same setup. Interestingly, the best performing MFCC system outperformed the PLP system by more than 10% (41.12% vs 52.62%). These results were achieved for features without any special noise suppression. However, the authors also showed that the additional noise suppression can lower the absolute *WER* down to 33.71%. Another interesting thing to note was the fact that bMMI always outperformed the standard MMI.

CHAPTER 3

GOALS OF THE THESIS

The development towards a fully informational society requires a better integration of machines into our lives and creating a more natural form of communication with them. The general objective of this thesis is to study existing methods and to find novel methods for robust recognition of strongly distorted speech. The situations include signals recorded with far distance microphones, in a noisy car environment and compressed speech. The focus will be given to techniques working at the level of acoustic model creation and front-end processing. The motivation for this research can be formulated as follows.

The recognition of recordings from a distant microphone is analysed for its application in the so-called smart homes which is based around the idea of using voice controlled appliances and controlling home faculties remotely. The second practical application is for the transcription of lectures and conference speeches recorded in auditoriums, where the microphone is usually placed at a distance from the speaker. The recognition of recordings from a car environment is analysed for two primary reasons. The first one is to provide human-to-machine interface for the voice controlled devices which include on-board navigation systems and other systems for controlling the car faculties. The second reason is more general as the conversations and phone calls made in cars also suffer from specific acoustic distortions which limits their usability for further processing.

Concerning the compressed speech, the algorithm widely known as MP3 belongs to the group of perceptual audio coders whose worldwide popularity is mainly historical as it appeared in the period of the rapid growth of the Internet and media sharing that came with it. It was developed primarily for the multimedia, namely for video and music storage and distribution [59], but it has seen successful use for speech encoding as well. Only music professionals, phoneticians, and audiophiles have always avoided using it. However, various studies have proved that even expert listeners can't distinguish between

original and encoded files for bitrates higher than 256kbps [60]. Also, people tended to use much lower bitrates because even highly compressed speech which containing audible distortions was perceived by human listeners as intelligible. Recently, professional studios and many broadcasters are leaving the MP3 coding tools and prefer formats that are better suited for speech (e.g. Speex or FLAC). However, a lot of speech data has already been compressed and archived utilizing the MP3 format, which makes the task of MP3 speech recognition a true research challenge. This fact led me to decision to study compensation methods which would enable the automatic processing of MP3 compressed recordings. Particular ideas analysed within this thesis can be formulated as follows.

- Signals recorded with far distance microphones suffer from additive noises, strong echoes and reverberations. Home environments often introduces only weak additive noises but public places introduce strong additive and convolution noises. *What is the contribution of front-end compensation methods for these situations? What is the contribution of acoustic modelling techniques? How much do these two environments differ in terms of ASR performance?*

- Signals recorded in a running car suffer from a strong additive noise caused by the running engine and the aerodynamic noise. Both get stronger as the driving speed increases. *What is the contribution of signal pre-processing methods for a running car ASR? What is the contribution of acoustic modelling techniques? How much do the differing driving conditions matter in terms of ASR performance?*

- The principal idea of MP3 compression is based on removing the imperceptible parts of the signal. *What are the primary distortions introduced by the compression and how do they affect the standard cepstral-based features? It is possible that the distortions are located at certain parts of the speech more often that at others?*

- The compression introduces non-linear distortions which corrupt signals spectra and the extracted features. *It is possible to optimize the feature extraction parameters such as the window length/step? Do the standard compensation and feature normalization methods improve the performance? Which features are better suited for this task?*

- Common way of improving ASR performance in adverse conditions is to employ either matched training or adapt the general purpose models to specific conditions. *What is the contribution of using the bitrate specific in comparison to general-purpose AM? Can the AM adaptation reduce this mismatch?*

- Theoretical and practical works on distorted speech recognition demonstrated, that adding noise to speech signal can improve ASR performance. *Can these ideas be extended further for MP3 speech?*

- Recognition systems based on neural networks have displayed much greater robustness against adverse environmental conditions than their GMM predecessors. However, these systems are discriminatory by their nature and thus purely data reliant, unlike the GMMs. *Can the DNN-HMM system outperform the GMM-HMM system? Can the DNN-HMM system still contribute from any feature-level compensation methods such as the ones studied in this thesis?*

Goals of the Thesis

On the basis of the above mentioned discussion, the principal goals of the thesis can be summarized as follows :

- to get acquainted with current state-of-the-art ASR systems and robust methods of front-end processing and acoustic model creation,
- to assembled an ASR system and to design appropriate evaluation tasks for a distant microphone, a running car and a compressed speech recognition,
- to analyse the contribution of the front-end noise suppression techniques, AM adaptation and discriminative training algorithms in the case of strongly distorted distant microphone and a driving car speech recognition,
- to analyse the contribution of various feature extraction setups, front-end compensation techniques, AM adaptation, discriminative training and to compare the GMM-HMM and DNN-HMM systems in the task of non-linearly distorted compressed speech recognition,
- to optimize the setup of the studied techniques for given recognition tasks,
- to design a novel compensation technique for compressed speech recognition and to verify its contribution using the assembled framework.

CHAPTER 4

BASELINE ASR SYSTEM

Although this thesis is focused on separate topics of robust speech recognition, most of the experiments shared used software, databases and recognition toolkits. This section summarizes these common resources. The initial studies were realized using the HTK Toolkit, the later ones were done using newly issued KALDI toolkit. The description of DNN-HMM system is provided in the particular section as it was used only for last experiments with compressed speech recognition. The following sections describe initial optimizations of AM creation. The first analysis compared the differences of using a triphone AM with a high number of tied-states and low number of added mixtures against an AM with a low number tied-states and a higher number of mixtures. The second analysis compared two different clustering strategies, an automatic and a knowledge based one, for a MLLR based AM adaptation.

4.1 Software tools

Following software was used to perform the described research.

- **HTK** [61] is a toolkit that was widely used for HMM construction and manipulation. It provides all of the necessary ASR utilities, but I used it only for AM training and decoding with basic AM techniques before switching to KALDI.

- **KALDI** [62] is a modern, widely popular ASR toolkit. It supports most of the current state-of-the-art AM techniques and ASR architectures. I used it for more advanced GMM-HMM training and for building DNN-HMM systems.

- **CtuCopy** [63] is our internal tool for feature extraction and speech enhancement that supports file formats usable for other ASR tookits such as HTK or KALDI. It was developed in our research group and is offered for free on our websites. I used it as a primary feature extraction tool for all experiments.

- **LAME** [64] is a free, high-quality MP3 encoder that uses improved psychoacoustic model and supports multiple compression features. It also gets used in many other third party encoders which broadens the relevance of presented results. My compression setup always used a constant bitrate, free format bitstream and the highest audio quality on output.

- **SoX** [65] is a command line utility which was used to convert MP3 coded speech back to PCM quality.

- **FFmpeg** [66] is a command line utility for high quality media manipulation. I used it to compresses speech into MPEG-4 (AAC) format and then decompress it back to PCM quality. I worked with the highest quality encoder *libfdk_aac* and a constant bitrate.

- **Praat** [67] is a free tool widely used for phonetic analysis. I used it to extract the pitch and formant contours.

4.2 Databases

Major portion of performed analyses were done on Czech recordings from the SPEECON and CZKCC databases. The databases contain recordings from acoustically clean conditions as well as recordings from acoustically adverse environments. This attribute made it possible to analyse real-life distortions and I didn't need to rely on adding artificial noises. The compressed speech analysis was also done for English and German languages to demonstrate that the negative effects are language independent. Foreign databases used for this purpose are introduced later in the corresponding sections.

The studies were performed using a Czech phoneme set which consisted of 44 monophones and a single silence model which also served as a garbage model for all other non-speech events, summarized in Table 4.1.

Table 4.1: Monophone set for Czech

Class	Phonemes
Non-speech	silence
Vowels	a, á, e, é, i, í, o, ó, u, ú, swa
Diphthongs	au, eu, ou
Consonants Voi.	b, d, ď, dz, dž, g, h, j, l, m, mv, n, ň, ng, r, ř, z, ž
Consonants Unvoi.	f, v, s, š, ch, /ř, p, t, ť, k, c, č

4.2.1 SPEECON

The adult part of Czech SPEECON database [68] contains utterances from 580 speakers recorded under different conditions, i.e. in offices, home environments, at public places, or in a car. Each speaker recorded utterances with an overall length of about 20 minutes. The database was recorded in 4 channels with different microphone types and positions, summarized in Table 4.2. The signals were sampled with $f_s = 16$ kHz rate and 16-bit precision and coded in the PCM format. All utterances were manually annotated and the actual pronunciation was written down along with possible mispronunciations or non-speech events.

The data for the subsets were selected as follows. The "Clean" set comprised signals from the office and entertainment environment with a switched-off background audio. The recordings in this subset were characterized by a weak background noise and were used to train a general purpose AM and for compressed ASR. The "Public" set comprised signals from the hall public environment, which was characterized by strong convolution distortions, and an open public environment which was characterized by strong background noises. The "Noisy Car" set comprised recordings from a standing car with a switched-on engine and a running car. The usage of CS0, CS2 and CS3 channels made it possible to analyse the influence of additive and convolution noises for distant microphone recognition as significant channel distortions appeared only in signals from CS2 and CS3 channels, as it is summarized in Table 4.2.

Table 4.2: Description of channels in SPEECON

Channel	Microphone	Type	Position	Level of Distortion
CS0	Sennheiser ME104	headset	2 cm	-
CS2	Sennheiser ME64	middle-talk	0.5-1 m	++
CS3	MBF HAUN	far-talk	3 m	+++

4.2.2 CZKCC

CZKCC database contains utterances from 710 speakers recorded in three different car brands and varying driving conditions. Each speaker recorded utterances with an overall length of about 40 minutes, pauses included. Three different microphones were used, two in a medium distance and one in close headset distance, as it is summarized in Table 4.3. The recordings were sampled at $f_s = 44.1$ kHz rate, coded with 16-bit precision and saved in the PCM format. The signals were later downsampled to 16 kHz in order to merge the data with SPEECON signals. Special care was taken to ensure that the acoustic conditions of selected signals were similar to the conditions found in SPEECON. All utterances were manually annotated and the actual pronunciation was written down along with possible mispronunciations or non-speech events.

The data for different subsets were selected as follows. The "Clean" set comprised recordings from a standing car with a switched-off engine and were used to train a general

Table 4.3: Description of channels in CZKCC

Channel	Microphone	Type	Position	Distortion
CS0	Sennheiser ME102	headset	2cm	-
CS2	AKG Q400 MK3T	middle-talk	dashboard	+++
CS3	Peiker ME27	middle-talk	rear-view mirror	+++

purpose AM and for compressed AS. The "Noisy car" set comprised recordings from a standing car with a switched-on engine and also speech from a running car. CZKCC database did not contain signals to match the public environments from SPEECON. The data from CZKCC was used only to train the general purpose AM but not for the actual evaluation on this task. Table 4.4 summarizes the selected data from both databases.

Table 4.4: Summary of used sets for distorted speech recognition

Evaluation Task	SPEECON	CZKCC
Far-microphone	Office, Entertainment, Public Place Hall, Public Place Open	Car engine off
Driving Car	Car engine on, running car	car engine on, running car
Compression	office, entertainment, car engine off	car engine off

4.3 Common ASR Framework

The initial experiments were realized using a progressively refined ASR framework, that is always described in a corresponding section. The remaining experiments were realized using a common ASR framework that was built partially upon this basic framework and is described below in order to avoid repetitions. This setup is always referred to as common later in the text.

4.3.1 Feature Extraction

The 13-dimensional PLP and MFCC features were computed from the signals using CtuCopy and with a window length of 32 ms and 16 ms shift. The vector was then normalized using CMN or CMVN. Five preceding and successive feature vectors were spliced onto the central vector and it was then transformed into 40 dimensional decorrelated vector via LDA and then transformed using MLLT. These features were used in my experiments as baseline features.

4.3.2 Acoustic Modelling

The process of acoustic model creation started with training a monophone AM using the Viterbi training algorithm. The monophone AM was then expanded into context-dependent crossword triphones and state-tied. The quality of this baseline AM was later improved using SAT, a combination of a UBM and SGMM and discriminative training using either the MMI, bMMI or MPE criteria. The final step involved using one of the discussed adaptation technique; MLLR, fMLLR or MAP; during the decoding. My SAT setup was based on fMMLR. The adaptation was always performed in an unsupervised fashion in two steps. In the first pass, I used the baseline SI model to obtain the phonetic transcription from which the linear transformations were estimated. During the second pass, these transforms were applied to get the final output transcriptions. It must be noted, however, that fMMLR also served the purpose of channel and environment adaptation in the case of mismatched recognition.

4.3.3 Language Model and Decoding

An internally developed LM [69] with a 340k vocabulary was used for the Czech. It was created using publicly available resources of the Czech National Corpus [70]. Most of the experiments were performed with the bigram LM as the contribution of using the trigram LM was marginal but the additional computational costs were great. The phoneme recognition task was also done on the bigram LM trained on local newspapers "Lidové Noviny". However, some analyses were performed for connected digits or voice commands recognition tasks and the LM in these cases was just a simple zero-gram infinite loop grammar. The decoding was done using *HVite* and *HDecode* from the HTK toolkit or one of the decoders from KALDI toolkit, depending on the stage of AM refinement.

4.3.4 Evaluation Criteria

The results of word level recognition were evaluated by word error rate (WER) and the word error rate reduction ($WERR$) criteria using the formulas:

$$WER = \frac{S + D + I}{N} \times 100 \ [\%] \tag{4.1}$$

$$WERR = \frac{WER_{base} - WER_{imp}}{WER_{base}} \times 100 \ [\%], \tag{4.2}$$

where S, D, I and N represents the number of substituted, deleted, inserted and the total number of words respectively. WER_{imp} represents the improved error rate and WER_{base} the base error rate against which the relative improvement was computed. The result of phoneme recognition was evaluated by phone error rate (PER) and the phone error rate reduction ($PERR$) criteria using analogous formulas. In addition, the recognized phonetic

transcription was remapped into three phonetic classes: voiced consonants, unvoiced consonants and vowels, and the phone error rate contribution for a particular phonetic class ($PERC_{cl}$) was computed using the following formula:

$$PERC_{cl} = \frac{PER_{cl}}{PER_{all}}, \ [\%] \tag{4.3}$$

where PER_{all} and PER_{cl} and were the total phone error rate and phone error rate for a particular class respectively.

4.4 State-Tying for GMM-HMM

This section describes an optimization analysis for a triphone-based GMM-HMM systems in the case of training data deficiency. This process involves reducing the total number of AM parameters $\mathbf{\Theta}_{AM}$ by clustering "similar" phoneme states. The most common approach is to employ the tree-based clustering algorithm with the help of the phonetic trees. The first option is to use a rather low number of tied-states with an addition of a high number of mixtures. It means using strict clustering conditions for context-dependent phonemes or using of a monophone AM without context dependency. An example of a recognition system which employed a monophone AM with up to 100 Gaussian mixtures per state was presented in [71]. The system was developed for the purpose of transcribing broadcast news. Alternatively, the system can be built to contain a high number of tied-states in conjunction with a small number of mixtures per state. In practise, this options means employing context-dependent AM trained with relaxed clustering conditions. For example, the system built for the purpose of online TV captioning presented in [72] contained approximately 5000 tied-states with 8 mixtures per state. Another example of this approach was examined in [73].

The ASR initial system for this analysis was created with the following setup. The full training set contained signals from 190 speakers with an overall length of 51 hours and contained 15392 different triphones. The full training set was then reduced in order to achieved the desired effect of data deficiency. A special focus was given to ensure that the selected signals in the reduced sets contained as much of a phonetically rich content as possible. This approach degraded the training sets quality only marginally as they contained a very similar number triphones. Only the quantity of data in it was significantly reduced. The details about the particular training sets are summarized in the Table 4.5.

The feature vector consisted of standard MFCC features complemented by dynamic and acceleration features and then normalized using the CMVN. The AM was trained using Baum-Welch algorithm. Five different stopping thresholds (denoted as *TB_xxx*) during the state-tying process were selected and the minimal occupation count for each leaf was set to 100 frames in order to avoid the problem of insufficient training data after the state-tying. Roughly 400 questions were defined, asking only about the immediate

Table 4.5: Training sets for optimized state-tying

	Data	Rich signals	Triphones
Full	51h	4.5h	15392
Reduced_A	10.8h	4.5h	13451
Reduced_B	8.9h	2.6h	12131

left and right context. The total number of generalized triphones, same as the number of mixture added, depended on state-tying conditions, but all systems were created to contain approximately 30k Gaussians in total. The evaluation set contained 15 minutes of recordings and the evaluation task consisted of a connected digits and voice control commands recognition and the whole system was built using the HTK toolkit.

The initial one mixture AMs achieved roughly the same results for the full training set, as is summarized in Figure 4.4. As the number of mixtures increased, the overall *WER* dropped, but the *TB_360* and *TB_720* models outperformed more strictly tied models for both recognition tasks. Table 4.6 summarizes the best achieved results for each training set and recognition task. The lowest *WER* of 2.4% and 1.89% was achieved for TB_720 setup in the digit and command tasks respectively. The second lowest values were then achieved for TB_360 setup and the TB_3800 setup proved to be the worst. Generally speaking, more complex AM (meaning models which were tied "less" strictly and thus allowed for more "free" parameters) with less added Gaussian mixtures achieved better results that less complex models with more added mixtures. This conclusion was true for both recognition tasks. This trend could be attributed to the fact that the training set contained enough data which allowed for a proper training of more complex models. On the other hand, the less complex models showed signs of overtraining very early in the process, especially for the digit task.

Table 4.6: Results for full training set for optimized state-tying

Set	TB Thresh.	Init. Gauss.	Mixt.	End Gauss.	*WER* [%]	
					Digits	Comm.
Tri_360	360	6665	5	33225	2.56	1.35
Tri_720	720	3893	7	27251	2.15	1.35
Tri_1800	1800	1926	15	28890	2.29	1.62
Tri_2800	2800	1410	20	28200	2.59	2.16
Tri_3800	3800	1138	22	25036	2.56	2.16

Table 4.7: Results for reduced sets for optimized state-tying

	Reduced A		Reduced B	
	Digits	Comm.	Digits	Comm.
TB_360	2.69	2.16	3.23	2.43
TB_720	**2.4**	**1.89**	**2.83**	**1.89**
TB_1800	2.96	2.43	4.31	1.89
TB_2800	3.36	2.43	4.58	2.43
TB_3800	3.5	3.5	3.5	2.7

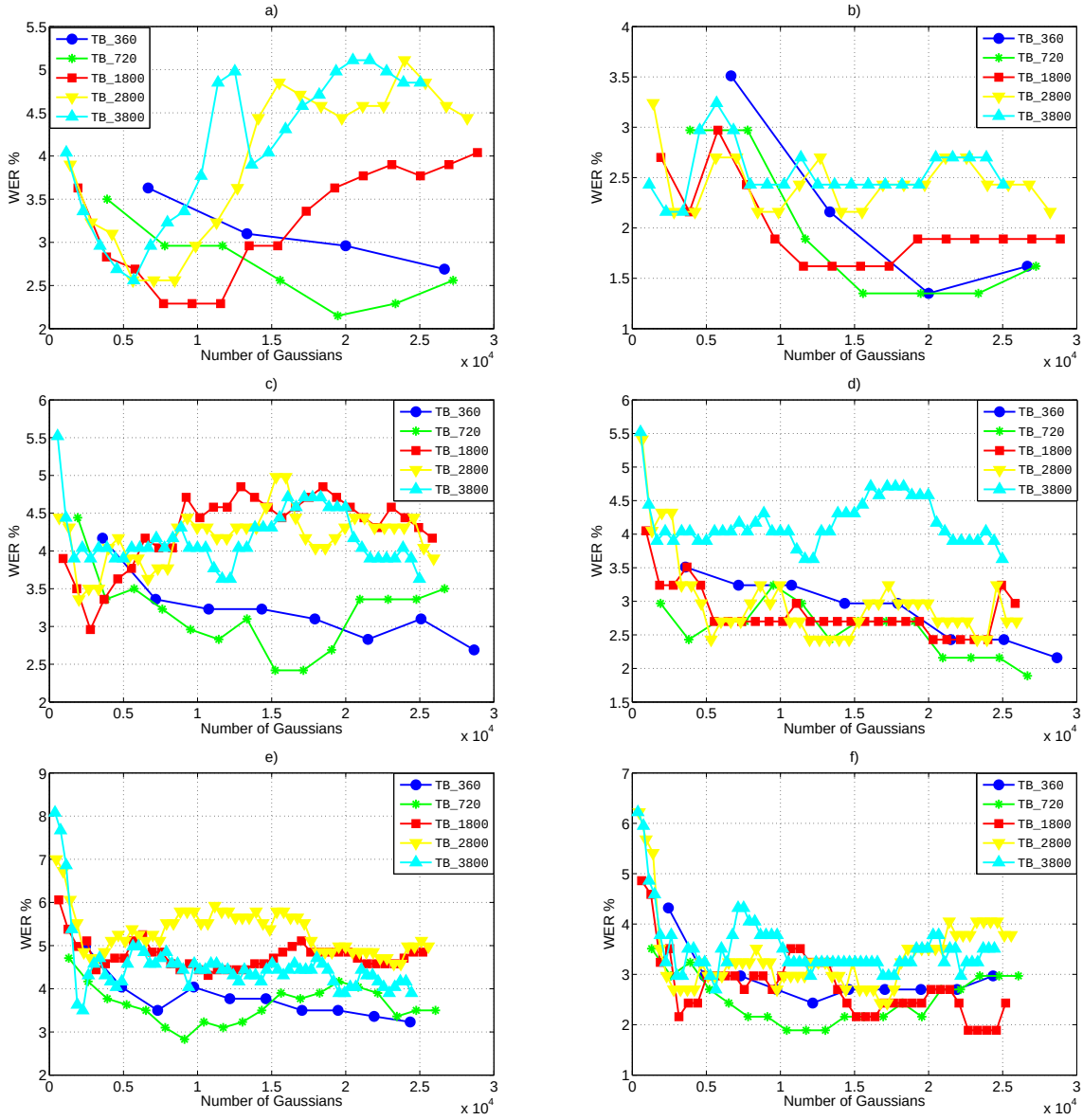

Figure 4.1: Results for : a) Digits full set; b) Commands full set,
c) Digits set A; d) Commands set A,
e) Digits set B f) Commands - set B.

The results for the reduced training sets are summarized in Table 4.7. The best results were achieved for TB_720 setup for both recognition tasks and the rest of the previously discussed conclusion proved to be true for the reduced training sets as well. Figure 4.4 summarizes the *WER* for an increasing number of Gaussians. It shows that the less strictly state-tied AMs achieved worse results for initial one mixture models. As the number of mixtures increased, the *WER* for *TB_360* and *TB_720* setups achieved comparatively better results. This trend occurred when the AMs reached more than 15k Gaussians.

The performed analysis demonstrated that using more tied-states and less mixtures is an optimal solution to parameter reduction as opposed to having less free states and more mixtures. As a result, AMs in all further experiments built using this strategy.

4.5 Clustering in MLLR Adaptation

This section proceeds with the optimization analysis of a GMM-HMM system with the help of phonetic trees. It describes a comparison analysis between the automatic and knowledge based clusterings for MLLR adaptation. Automatic clustering is based on an objective similarity measure between different phonetic models. The knowledge-based clustering exploits the division of phonemes into classes based on their manner of production. In this case, the phonemes with similar articulation and vocal tract characterization were assumed to be acoustically similar and thus were placed into the same regression class. It was shown in [49] that knowledge-based clustering with only two defined classes, one for speech phonetic units and one for non-speech units, may bring considerable improvement to accuracy in real-life conditions with a high level of background noise.

Table 4.8: Used knowledge-based triphone classes

Class	**Phonemes**
Class 1	silence
Class 2	(*-a+*),(*-á+*),(*-e+*),(*-é+*),(*-i+*),(*-í+*), (*-o+*),(*-ó+*),(*-u+*),(*-ú+*),(*-swa+*)
Class 3	(*-au+*),(*-eu+*),(*-ou+*)
Class 4	(*-l+*),(*-r+*),(*-j+*)
Class 5	(*-m+*),(*-n+*),(*-ň+*), (*-ng+*),(*-mv+*)
Class 6	(*-f+*),(*-v+*),(*-s+*),(*-z+*),(*-š+*),(*-ž+*), (*-ch+*),(*-h+*),(*-r+*),(*-ř+*),(*-/ř+*),
Class 7	(*-p+*),(*-b+*),(*-t+*),(*-d+*),(*-ť+*),(*-ď+*), (*-k+*),(*-g+*)
Class 8	(*-c+*),(*-dz+*),(*-č+*),(*-dž+*)

Table 4.9: Further division of vowels

	A) Lip Position
Class 2.1a Class 2.2a	(*-a+*),(*-á+*),(*-e+*),(*-é+*),(*-i+*),(*-í+*), (*-o+*),(*-ó+*),(*-u+*),(*-ú+*)
	B) Tongue Movement
Class 2.1b Class 2.2b	(*-e+*),(*-é+*),(*-i+*),(*-í+*), (*-a+*),(*-á+*), (*-o+*),(*-ó+*),(*-u+*),(*-ú+*)

This analysis extended the cited approach further by defining multiple regression classes according to basic phonetic categorization of generally recognized Czech phonemes [74]. Since this division was too coarse for the purpose of this study, the vowels were further divided into the following classes: vowels, nasals, liquids, fricatives, plosives and affricates and special classes were added for the diphthongs and a silence. Since it was proved that the acoustic representation of a triphone is mostly determined by its middle monophone [75], all triphones with the same middle monophone were clustered into a same class. The described strategy resulted in a fairly straightforward composition with a total number of 8 expertly determined regression classes summarized in Table 4.8. This *Basic*

setup was also expanded further as vowels in *Class 2* were split into two classes, according to either the lip position or tongue movement, summarized in Table 4.9. These setups are denoted *10a* and *10b* further in the text.

The feature vector consisted of standard MFCC features complemented by the dynamic and acceleration features and then normalized using the CMVN. The training set contained 51 hours of speech. The AM was trained to contain 15k states and six mixtures per state. The adaptation was performed in a supervised, speaker-specific fashion. The transforms were estimated using a set of 170 utterances for a speaker on average, with an overall length of about 4 minutes. The number of regression classes for automatic division was gradually increased, starting from 2 and ending at 32. The performance was evaluated in the LVCSR task on the set of 275 utterances containing only whole sentences with an overall length of 27.5 minutes. The decoding was done with *HDecode* decoder and a trigram LM with 340k vocabulary.

Table 4.10: Results for both MLLR clustering strategies

	Automatic						Knowledge-based		
Num. of Classes	2	4	8	12	16	32	8	10a	10b
WER[%]	22.1	20.3	20.1	20.5	20.6	20.4	20	20.3	20.4
WERR[%]	16.7	19.7	20.2	19.3	19.5	19.8	20.9	20.1	19.9
Baseline	25.67 %								

Table 4.10 summarizes the obtained results. The best results with automatic clustering were achieved for 8 regression classes and the *WERR* reached 20.25% over the baseline system. The *WER* then began to rise after a slight decline for 12 classes, but 32 classes proved to be a performance ceiling, since no improvement past this value was measured. Also, the highest *WERR* was measured for speakers with a relatively high *WER* before adaptation and likewise speakers with relatively low baseline *WER* showed only small values of relative improvement. The best overall results for knowledge-based clustering were achieved with the *Basic* setup, which yielded an average absolute reduction in *WER* of 5.69% over the baseline system. Both *10a* and *10b* setups yielded very similar *mbox WER* reductions, 5.35% and 5.26% respectively. Also, all studied manual clusterings achieved better results than the automatic one, albeit only very slightly. One interesting thing was a high value of variance in *WER* for the baseline system, when the difference between the best and the worst speaker was 29.84% in absolute.

The most limiting factor of this study was that the number of manually determined classes and their composition had to be optimized according to the amount of adaptation data. The more I had for adaptation the finer division was required in order to achieve the optimal MLLR performance. As a result, all further experiments which employ the MLLR adaptation use an automatic regression classes construction approach as the improvements of using manual division was too small to justify the necessary design adjustments and optimizations.

CHAPTER 5

DISTANT MICROPHONE AND CAR RECOGNITION

This chapter deals with the recognition of strongly distorted speech recorded with a distant microphone or in a running car. Each degrading situation is analysed separately and the focus is given to algorithms working at the front-end processing level or acoustic modelling level. The front-end processing algorithms include Extended Spectral Subtraction, feature normalizations and the combination of LDA and MLLT. The acoustic modelling algorithms include SAT, UBM, SGMM and discriminative training. The study was performed on SPEECON and CZKCC data due to the fact that both databases contain utterances recorded simultaneously by several microphones located in different positions and conditions.

5.1 Distant Speech Recognition

This section describes the optimization analysis on speech recorded with a middle- and far-distance microphones. The reference results for the close-talk microphone are always presented along the results for distance microphones. The contribution of the discussed front-end processing and acoustic modelling algorithms is evaluated in one acoustically clean and two public environments, each characterized by its own specific acoustic distortions. The AM trained on acoustically clean speech was also used for mismatched recognition of public recordings in order to evaluate generalization qualities of the studied acoustic modelling techniques, especially the AM adaptation. Also, this mismatched training-evaluation scheme allowed to compare the generalization qualities of different discriminative training criteria for different acoustic conditions.

5.1.1 Acoustic Conditions

The acoustic analysis presented in this section compares the estimated SNR between each channel in three studied environments. The SNR levels discussed in this section were estimated during the database collection by a recording device and are included in the annotation files. The distant microphone recordings from acoustically clean environment, commonly denoted as "Quiet" further in the text, were characterized by a relatively low additive and convolutional noises, especially if compared to the public environments. The SNR histograms for each channel are summarized in Figure 5.1. More than 20 dB difference in SNR between close and far distance speech prove that far channel data had significantly worse quality even for this environment. This degradation was most likely caused by attenuation of the speech signal and increased additive and convolution distortions. There was also another interesting thing to note. The distributions for CS2 and CS3 shifted their central positions (which was to be expected) but they also became more tilted towards the lower SNR values. The CS0 channel was tilted slightly towards the higher SNR values, while the CS2 and CS3 started to approximate a lognormal distribution rather than the expected normal one. This observation proved that robust ASR is necessary even for acoustically clean environments in case the recording come from a distant microphone.

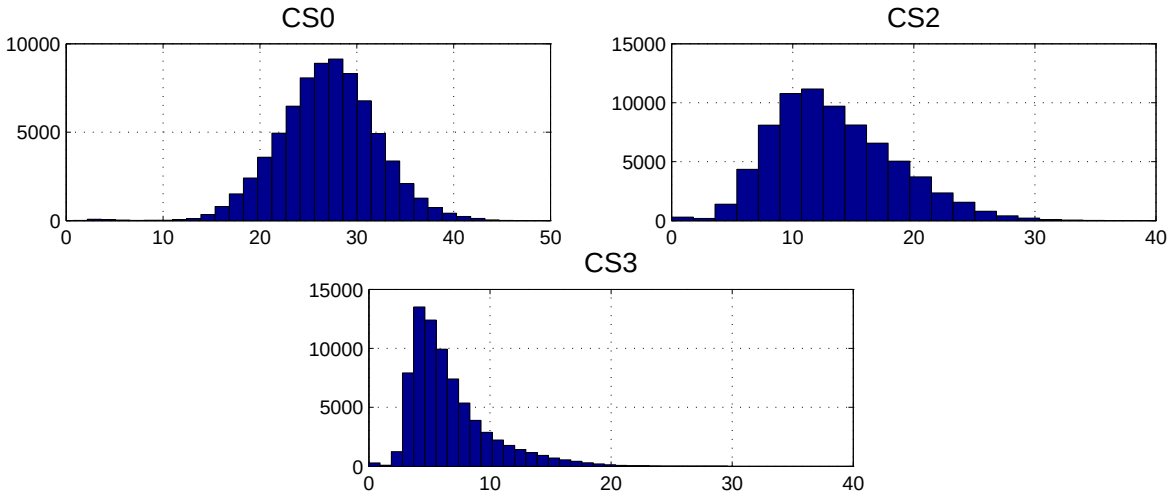

Figure 5.1: SNR histograms for all channels for Quiet Environment

The following analysis on far distance microphone recognition were focused on recordings from a public environment. Public environment was characterized by a naturally present additive and convolution noises. The selected recordings were divided further into two distinct subsets. The recordings denoted as PubHall later in the text were recorded in a closed space and thus contained a relatively strong convolutional distortion caused by the echoes and reverberations. The SNR histograms for the PubHall are illustrated in Figure 5.2. The SNR analysis revealed, that the CS0 channel contained only small amount of noise, which was to be expected. The SNR levels for CS2 and CS3 channels were much lower as we once again observed about 20 dB drop. Overall, the SNR distributions were only marginally worse and closely resembled the Quiet environment for all channels.

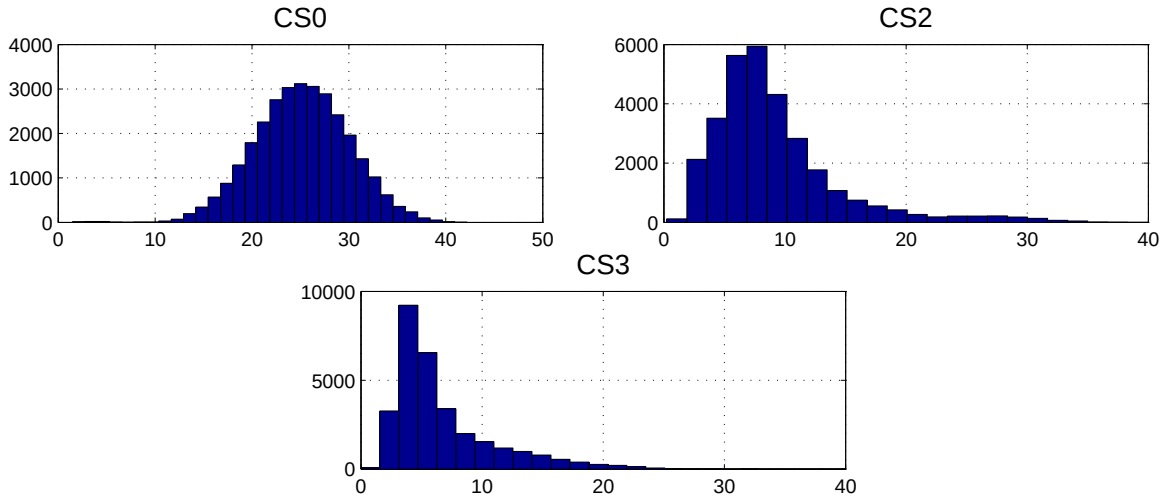

Figure 5.2: SNR histograms for all channels for Public Hall Environment

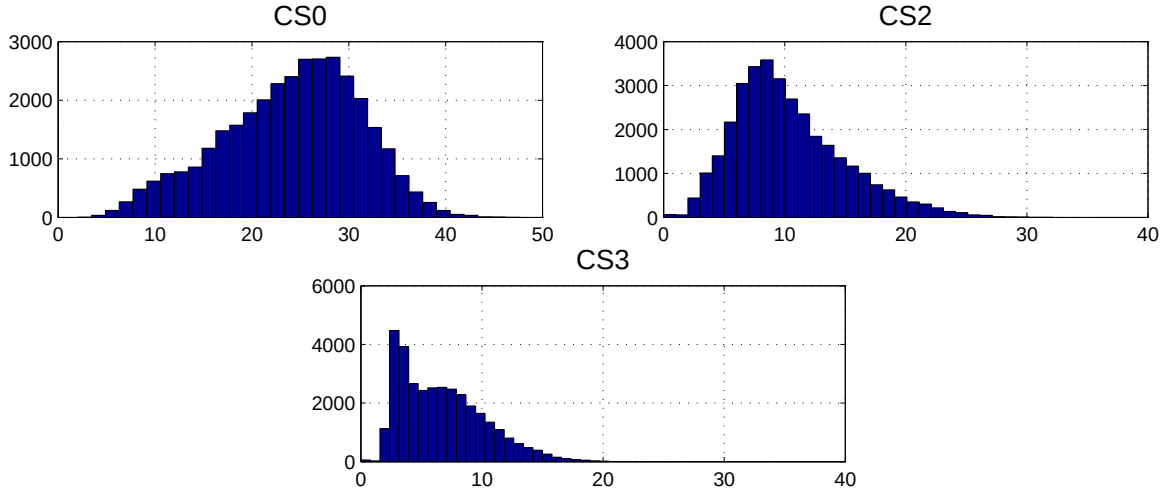

Figure 5.3: SNR histograms for all channels for Public Open Environment

The final evaluation was realized with the data denoted as PubOpen which were recorded in an open environment, such a street. These recordings contained stronger additive noises which are naturally present in these types of situations. The SNR histograms for the PubOpen are illustrated in Figure 5.2. This environment displayed non-Gaussian distributions for all analysed channels. This trend was especially pronounced for CS3 channel which displayed a high peak in the histogram around 3 dB value. This peak indicated a strong and consistent distortion present during the recording session for a large potion of signals. Table 5.1 summarizes the parameters of the estimated Gaussian distributions for all environments. It can be noted that CS3 channel displayed very similar statistics across the channels, while the values for CS0 and CS2 were significantly different across the environments. This analysis indicated that using ASR in these condition was most likely to be rather difficult. However, it is also important to realize that the presented SNR values were just estimates and thus might not be 100% accurate.

Table 5.1: Statistical parameters for distant microphone ($\mu \pm \sigma$) [dB]

Env.	CS0	CS2	CS3
Quiet	27 ± 4.66	13.4 ± 4.45	6.7 ± 3.06
PubHall	25.2 ± 4.9	9.2 ± 5.4	6.7± 4.2
PubOpen	24.1 ± 7.2	10.4 ± 4.6	6.7 ± 3.5

5.1.2 Front-End Processing for Distant Speech

This section analysed the performance of the optimized ASR presented in the previous chapter in conjunction with the ESS and CMS techniques for Quiet environment with distant microphones. The intended application was for the so-called "Smart Home". The deployment of ASR system for voice command control in such application leads to the usage of middle or far distance microphones, which are usually embedded in devices themselves or in the walls and ceiling of the house. This requirement disables the usage of directional microphones. When a microphone with omnidirectional characteristics is used, especially with far distance placement typically, it leads to the inevitable presence of various kinds of noises of rather high levels. The purpose of ESS was to compensate the lower SNR which is typical for far microphone speech in general. This technique was chosen because it works without the need of a voice activity detector and the authors in [49] also proved that it contributed reasonably well to speech recognition in very noisy environments. Another advantage of this technique is the fact that it can also suppress non-stationary noise when its spectral characteristics changes more slowly then the characteristics of speech. The noise cancellation based on ESS was implemented with the following parameters:

- integration constant p = 0.95,
- realized in magnitude domain,
- applied before the filter-bank.

The second studied technique was a simple CMS. Although the general principle of CMN/CMS is clear and simple, the practical implementations differ. The CMN estimates the average cepstrum over the whole utterance which leads to a various number of samples over which the average is estimated. Another drawback is the possibility to apply CMN only in off-line applications. On the other hand, the CMS accumulates the necessary statistics and averages the cepstrum over the sliding window of a limited length which makes it possible to use this techniques for both on-line and off-line applications. This section analysed two approaches to CMS computation. Firstly, it was the standard computation of a moving average (MA) over the long-time window of a given length. The second approach was the computation on the basis of recursive exponential averaging (EA). There were, however, several decision that had to be made. The key one was the length of a long-time window over which an average cepstrum was computed. Particular authors work with various lengths of this window from 1 s up to values above 10 s. Figure 5.4 illustrates the averaging results for both solutions. There are several important

things that can be taken from them. The long-time window should be definitely longer than 1 s, but the results for windows longer than 5 s began to be near the same.

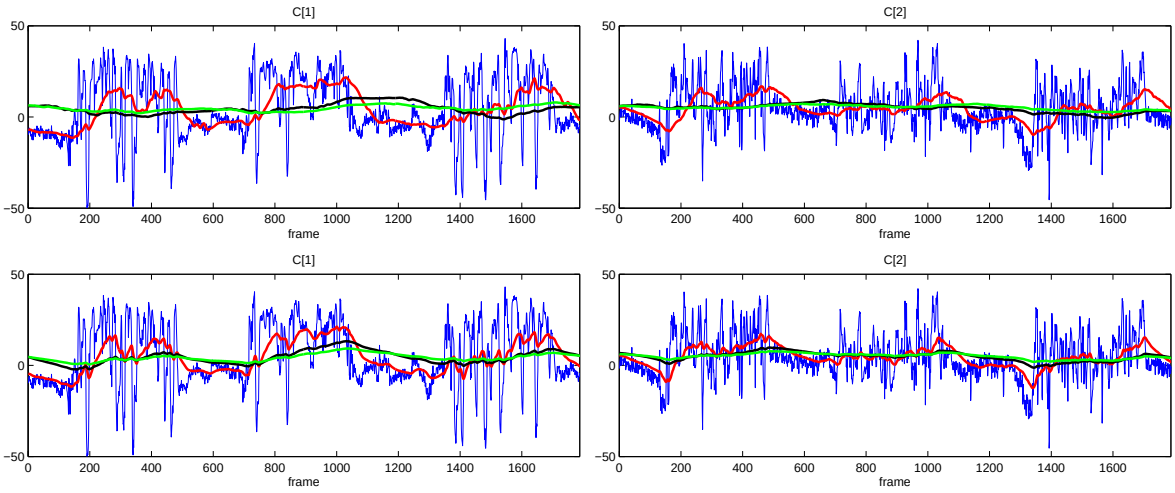

Figure 5.4: Illusration of CMS with EA/MA averaging and smoothing constants 1 s (red), 5 s (black), 10 s (green)

Since all of the signals were recorded simultaneously using different microphones, channel distortion could be quantified basically by an Euclidean cepstral distance computed between the reference CS0 signal and CS2/CS3 signals computed either from complete cepstral vector with coefficient c_0 (*CD0*) or just from the coefficients $c_1 \div c_L$ (*CD1*). Table 5.2 shows results estimated from subset of about 2000 utterances. The trend observed for both CMS methods was the decrease in the (*CD0*) and (*CD1*) as the averaging time windowed increased in length. The (*CD1*) distance was consistently lower for independent CMS system than for the combined system, regardless of the channel. The differences were however very small.

Table 5.2: Cepstral distance for various parametrizations

			CS2		CS3	
			CD0	CD1	CD0	CD1
CS0	×	CS_x	$\mu \pm \sigma$	$\mu \pm \sigma$	$\mu \pm \sigma$	$\mu \pm \sigma$
mfcc	×	*mfcc*	42.57 ±*11.11*	37.57 ±*11.30*	54.91 ±*16.33*	49.11 ±*18.87*
mfcc	×	*mfcc_ESS*	41.99 ±*11.32*	38.79 ±*11.52*	54.69 ±*17.04*	50.36 ±*18.70*
mfcc	×	*mfcc_exp1*	48.23 ±*14.02*	43.44 ±*15.46*	54.25 ±*16.80*	48.26 ±*19.57*
mfcc	×	*mfcc_exp5*	45.82 ±*12.45*	41.06 ±*13.14*	52.93 ±*15.74*	46.93 ±*18.29*
mfcc	×	*mfcc_exp10*	45.27 ±*12.14*	40.48 ±*12.71*	52.60 ±*15.55*	46.57 ±*18.07*
mfcc	×	*mfcc_b1*	49.25 ±*14.26*	44.59 ±*15.62*	55.02 ±*17.01*	49.15 ±*19.71*
mfcc	×	*mfcc_b5*	46.07 ±*12.39*	41.41 ±*12.84*	53.10 ±*15.52*	47.18 ±*17.92*
mfcc	×	*mfcc_b10*	45.51 ±*12.15*	40.77 ±*12.67*	52.77 ±*15.55*	46.78 ±*18.03*

The ASR system was built with the following setup. The feature vector consisted of standard MFCC features, complemented by their 1^{st} and 2^{nd} order dynamics. The ESS was applied during the MFCC computation and before the application of the filterbank and the CMS was applied afterwards. Equivalent time constants for both methods

Table 5.3: Summary of used parametrization setups

Param.	ESS	T [s]		
mfcc	no	-		
mfcc_ESS	yes	-		
mfcc_b/mfcc_exp	no	1	5	10
mfcc_ESS_b/mfcc_ESS_cms	yes	1	5	10

were set to 1, 5, and 10 s. Together with ESS, 14 different feature extraction setups summarized in the Table 5.3 were analysed. The training sets contained signal with an overall length of about 51 hours for all channels. The final triphone-based AM contained approximately 7k tied states and 14 mixtures per state. The systems were always trained in matched conditions and no adaptation was employed as the purpose was to evaluate the performance of these front-end processing techniques. The evaluation was done on a small vocabulary recognition task of 468 different commands. The utterances had a single word or multiple words structure and they contained potentially used commands for household appliances. The evaluation set had an overall length of about 15 minutes and the system was constructed using HTK.

In the first experiment the performance was evaluated for a system with a standalone CMS and the results are summarized in Table 5.4. In this case, a clear improvement was observed for all setups on the CS2 channel, while the CS0 showed the degradation in accuracy. The results for CS3 channel were mixed. The time constant of 5 seconds for both averaging methods proved to be unfit. The EA/MA methods with time constant 1/10 seconds performed better and decreased the error rate when compared to reference feature extraction.

Table 5.4: Results for reference and standalone CMS

	CS0		CS2		CS3	
Param	WER	WERR	WER	WERR	WER	WERR
mfcc	1.89	*0*	9.73	*0*	33.51	*0*
mfcc_b10	2.43	*-28.57*	4.59	*52.86*	31.89	*4.83*
mfcc_b5	2.43	*-28.57*	4.86	*50.05*	37.57	*-12.11*
mfcc_b1	2.16	*-14.28*	5.41	*44.39*	32.7	*2.41*
mfcc_exp10	2.97	*-57.14*	5.95	*38.84*	32.7	*2.41*
mfcc_exp5	2.16	*-14.28*	5.41	*44.39*	40	*-19.36*
mfcc_exp1	2.16	*-14.28*	4.59	*52.82*	33.24	*0.8*

The second analysis compared the system without any noise suppression and the system with either ESS or the combination of both ESS and CMS. The results with standalone ESS and the combined system are summarized in Table 5.5. The application of a standalone ESS increased the robustness only in the case of CS3 channel. In both the CS0 and CS2 channels, the additive noise from the background was rather small, $SNR_{CS0} = 27$ dB and $SNR_{CS2} = 13.4$ dB. The most likely explanation was that the introduction of non-linearities and musical tones degraded the speech quality, which resulted in the increase of *WER*. The final analysis employed the combination of both methods. The

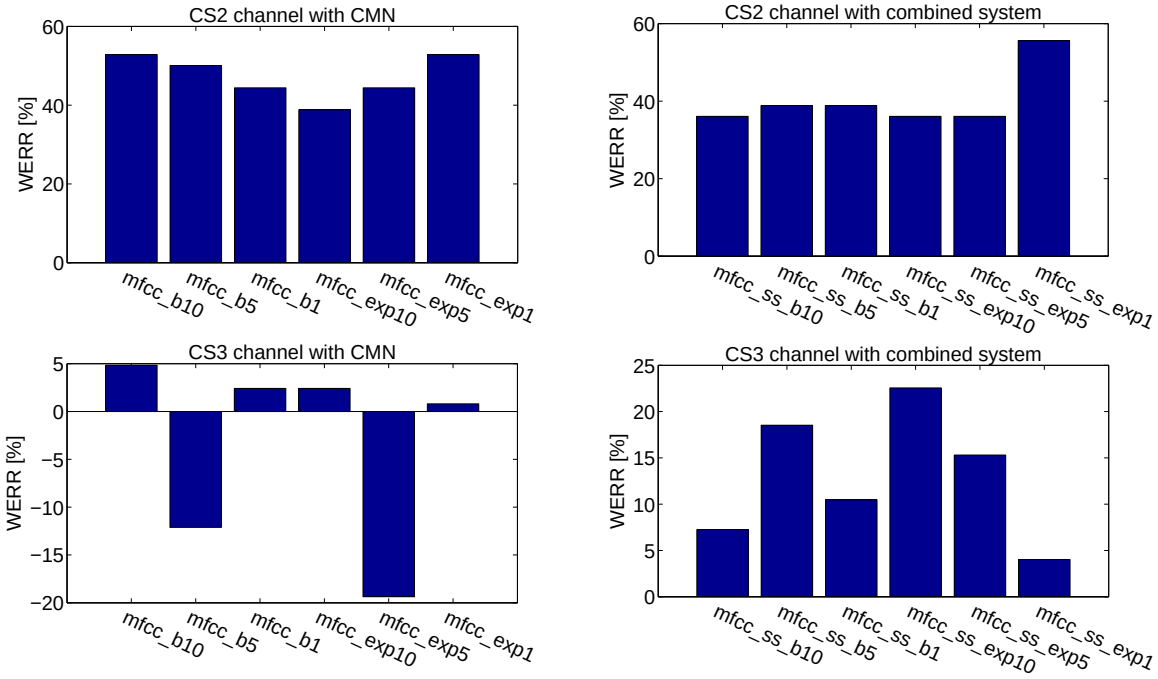

Figure 5.5: *WERR* for various parametrizations

combined system proved to be the most effective when an improvement was reached for any setup and for both noisy channels. Even a slight decrease of 0.27% in *WER* for CS0 channel was observed. Figure 5.5 illustratively summarizes the principal results and the main conclusions can be summarized as follows.

Table 5.5: Results for ESS compesanted and combined system

	CS0		CS2		CS3	
Param	WER	WERR	WER	WERR	WER	WERR
mfcc_ESS	2.43	*-28.57*	12.7	*-30.52*	29.46	*12.08*
mfcc_ESS_b10	2.43	*-28.57*	6.22	*36.07*	31.08	*7.25*
mfcc_ESS_b5	2.16	*-14.28*	5.95	*38.84*	27.3	*18.53*
mfcc_ESS_b1	2.43	*-28.57*	5.95	*38.84*	30	*10.47*
mfcc_ESS_exp10	3.24	*-71.42*	6.22	*36.07*	25.95	*22.56*
mfcc_ESS_exp5	1.62	*14.28*	6.22	*36.07*	28.38	*15.3*
mfcc_ESS_exp1	1.89	*0*	4.32	*55.60*	32.16	*4.02*

- The contribution of CMS was overall positive for CS2 channel. The *WERR* for *mfcc_exp1/10* and *mfcc_block1/10* reached up to 52.8%.
- The contribution of CMS also positive for CS3 channel, but only with the highest 10 s and the lowest 1 s integration constants. The *WERR* reached up to 4.8%.
- This combination of ESS and CMS achieved the best results and decreased the error rates for both CS0 and CS3 channels. Specifically for CS2 channel, the highest *WERR* of 55.6% was obtained for *mfcc_ESS_exp1* and for CS3 channel the highest *WERR* of 22.5% was obtained for *mfcc_ESS_exp10*.

- The sole application of ESS proved to yield negative improvement for the close-talk channel. This problem is rather known and is tied to the introduction of non-linearities and music tones. However, the combination of ESS and CMS showed improvement even for CS0 channel for one specific setup.

5.1.3 Acoustic Modelling for Distant Microphone

The previous analyses concluded that the combination of ESS and cepstrum normalization could bring addition improvements for heavily distorted CS2 and CS3 channels. Further analysis was focused on extending the validity of these experiments with more advanced systems. The recognition system was built using the common ASR framework described in Chapter 4, while it also included the addition of ESS and was extended for all three far distance microphone environments. However, there were only two AMs. The first AM was trained for the Quiet environment and a common AM was trained for both public subsets. The choice to train a single public AM was influenced mainly by the lack of data. Only the evaluation set was split into two subset, PubHall and PubOpen. The amount of training and evaluation data is summarized in Table 5.6. It is also important to realize that the amount of data remained the same for all three channels. The evaluation was done on a standard LVCSR task with a bigram LM and 340k vocabulary. The potential application of these systems is for the recognition of speeches made in auditoriums or in open public spaces.

Table 5.6: Summary of data sets for distant microphone

	Quiet	PubHall	PubOpen
Train	72h	43.5h	
Test	2.7h	0.5h	0.55h
OOV	1.4%	0.7%	0.5%

The results for the Quiet environment are summarized in Table 5.7. This recognition task represented an ideal situation where the acoustic conditions during the recording were good. Thus, this system set the bar for the subsequent ASR systems that were analysed in this thesis. This framework was also used for all other acoustically degraded conditions in the mismatched training-evaluation scheme. The initial system achieved the *WER* of 23.74% and the final MPE trained models performed at 14.01% and the actual difference between the *Baseline* and MPE trained models was only 9.7% absolute, which corresponded to 41% *WERR*. The results for the CS2 were only slightly worse when the *Baseline* system performed at 28.89% and the MPE models at 15.39%, which corresponded to 46.7% *WERR*. Finally, the results for the CS3 channel were much worse. The *Baseline* system achieved 54.14% error rate while the MPE models performed at 34.32%, which corresponded to 36.6% *WERR*. The highest average *WERR* across the channels was achieved by the application of SAT, 18.46% on average. The second highest relative reduction of 15.4% was observed for the combination of UBM and SGMM, then the MPE training criteria and finally the usage of LDA. The MPE training criteria has achieved slightly better overall results than the bMMI (0.55% on average) for all studied

channels. The direct comparison between the channels showed that the the absolute *WER* difference for the baseline AMs between CS0 and CS2/CS3 channels was 5.15% and 30.4% respectively. The same difference for MPE models dropped down to 1.38% and 20.31%. It is also interesting to note that the CS2 channel contributed the most from practically all studied modelling techniques (aside from the UBM+SGMM, which was highest for CS3), while the lowest *WERR* between each subsequent refinement was observed for CS3 channel on average. This results documented a decreasing robustness of studied techniques against the distortions in CS3 channel.

The application of ESS increased the error rates for all studied channels and evaluation sets. This conclusion held true even for weakly refined *Baseline* AMs. However, the *WERR* had an overall decreasing tendency for higher channels. Thus, it could be concluded that the application of ESS was not recommended for the clean acoustic conditions as the introduction of additional music tones brought more harm than the removal of noises. These results also indicated that the actual degradation to the speech wave recorded with a distant microphone in clean conditions was not necessarily tied to the introduction of additive noises.

Table 5.7: Results for Quiet environement

			Adapted			
Channel	Baseline	LDA	SAT	SGMM	bMMI	MPE
CS0	23.74	21.8	18.19	16.14	14.25	14.01
CS2	28.89	26.37	19.99	16.49	15.74	15.39
CS3	54.14	51.37	43.84	36.16	34.75	34.32
CS0+ESS	26.03	24.12	20.29	17.89	15.21	14.99
CS2+ESS	30.67	28.24	21.99	17.99	17.00	16.21
CS3+ESS	54.86	53.21	47.98	40.72	37.03	35.65

The result for the PubHall environment with matched training for the progressively refined AM are summarized in Table 5.8. The results for the *Baseline* system were considerably worse in comparison to the Quiet subset. The CS0 channel performed at 36.78%, the CS2 channel at 66.36% and the CS3 channel at 82.29%. The MPE models performed at 20.91%, 43.21% and 63.5%. These final error rates corresponded to 43.1%, 34.9% and 22.8% *WERR*. Since these relative improvements over the initial systems had a decreasing tendency, which documented the limitations of the studied techniques in the case of strong convolution and additive noises. It was also interesting to compare the performance for each channel in the Quiet and PubHall conditions. The relative difference between the CS0 microphone and the MPE models reached 33%. This was a considerable performance drop. However, the relative difference between the CS2 was much worse, it reached 64.4% while the relative difference for the CS3 microphone was 46%. This observation lead to the conclusion that the CS3 channel for Quiet environment must have contained a certain level of distortion that was present for CS2 only marginally. However, once the distortion got stronger due to the nature of the environment (PubHall is a closed space with sound reflections), even the CS2 microphone which was about 0.75 m away from the speaker started picking up these reverberations and the performance dropped more significantly than for CS3 channel.

Table 5.8: Results for PubHall environment with matched training

			Adapted			
Channel	Baseline	LDA	SAT	SGMM	bMMI	MPE
CS0	36.78	35.13	29.16	23.63	21.40	20.91
CS2	66.36	61.16	54.81	45.37	43.74	43.21
CS3	82.29	78.82	73.40	65.19	62.74	63.50
CS0+ESS	38.12	36.42	29.65	24.57	21.58	21.44
CS2+ESS	64.55	61.42	54.63	47.27	43.92	44.00
CS3+ESS	79.73	78.20	71.87	63.91	60.87	60.85

The results with the mismatched training are summarized in Table 5.9. The mismatched AM performed actually better for the CS0 microphone than the matched models. This performance drop for CS0 channel could be explained as follows. The mismatched data contained recordings from all two public subsets which have been shown to have different SNR levels and distributions. As a consequence, the matched AMs were less suitable for these recordings than the mismatched Quiet AM which was trained on more acoustically similar and homogeneous recordings. On the other hand, the advantage of matched training was clear for the CS2 and CS3 channels where the absolute difference in *WER* reached 5.1% and 3.5% respectively. These results again demonstrated a decreasing capability of the studied modelling techniques to deal with strongly distorted speech from CS3 channel.

A second interesting thing was the contribution of the ESS technique. The contribution for CS0 was statistically insignificant for the mismatched training and only marginal for matched training. The CS2 recordings contributed from its application with matched training only marginally for the *Baseline*, and *SAT* models. The *WERR* for these stages reached 2.7% and 0.3%. However, a significant improvement for the CS2 channel and mismatched training was observed for all AM refinement stages, where even the MPE trained models contributed by 9% *WERR*. Also, the CS3 channel displayed a consistent improvement from the ESS application with matched training, when the *WERR* reached 3.1%, 0.8%, 2.1%, 2%, 3% and 4.2% respectively. This observation further proved that the application ESS was able to bring improvement for the far distance microphone recognition in case of a strong distortion.

Table 5.9: Results for PubHall environment with mismatched training

			Adapted			
Channel	Baseline	LDA	SAT	SGMM	bMMI	MPE
CS0	38.16	35.09	28.80	22.51	19.35	20.02
CS2	73.77	69.09	58.69	51.54	47.97	48.28
CS3	83.77	82.18	76.98	68.45	67.28	67.02
CS0+ESS	38.92	36.02	28.62	23.50	19.13	19.79
CS2+ESS	66.84	64.33	55.20	48.06	43.43	43.92
CS3+ESS	81.88	80.45	75.09	67.79	65.19	66.62

The result for the PubOpen environment with matched training are summarized in Table 5.10. The results for the *Baseline* system were again much worse in comparison

to the Quiet subset. The CS0 channel performed at 40.18%, the CS2 channel at 51.23% and CS3 at 64.64%. The MPE criteria again outperformed the bMMI criteria. The error rates reached 24.29%, 30.47% and 44.48%, which corresponded to 39.5%, 40.7% and 31.2% *WERR*. These relative improvements were higher on average than the ones for the PubHall environment. However, the CS0 results were worse than their PubHall counterparts but the CS2 and CS3 channels achieved considerably better results. These two observation demonstrated that the studied techniques were much more robust against the additive noises, which dominated the PubOpen environment, than against the convolution noises which dominated the PubHall environment. This conclusion was especially important, if we take into account that the estimated SNR levels for both public subsets were very similar (difference was within 1.5 dB). However, the contribution of the ESS was only marginal in this case, unlike in the previous case, as a statistically significant contribution was observed only for CS2 channel. No improvement was observed for CS2 channel.

Table 5.10: Results for PubOpen environment with matched training

			Adapted			
Channel	Baseline	LDA	SAT	SGMM	bMMI	MPE
CS0	40.18	37.23	30.54	26.21	23.53	24.29
CS2	51.23	47.46	38.53	31.70	30.43	30.47
CS3	64.64	58.88	51.88	46.40	44.80	44.48
CS0+ESS	41.03	36.38	30.36	26.25	23.53	24.60
CS2+ESS	52.76	48.69	38.84	33.06	31.96	31.65
CS3+ESS	64.92	59.16	52.68	45.56	44.36	43.56

The results for the PubOpen environment with mismatched training are summarized in Table 5.11. Matched training brought a statistically significant improvement only for the CS3 channel and some improvement for CS2 channel. Once again, the matched models performed worse for CS0 channel than the mismatched models. The application of ESS was found to be beneficial only in the case of CS3 channel.

Table 5.11: Results for PubOpen environment with mismatched training

			Adapted			
Channel	Baseline	LDA	SAT	SGMM	bMMI	MPE
CS0	41.38	36.16	29.64	25.22	23.97	23.62
CS2	50.53	46.89	37.35	32.14	30.30	29.55
CS3	71.44	69.16	57.12	48.96	47.08	47.24
CS0+ESS	41.34	35.98	30.40	25.71	24.64	24.42
CS2+ESS	50.04	45.58	37.70	32.09	30.30	29.25
CS3+ESS	68.24	66.88	56.36	47.60	46.48	45.80

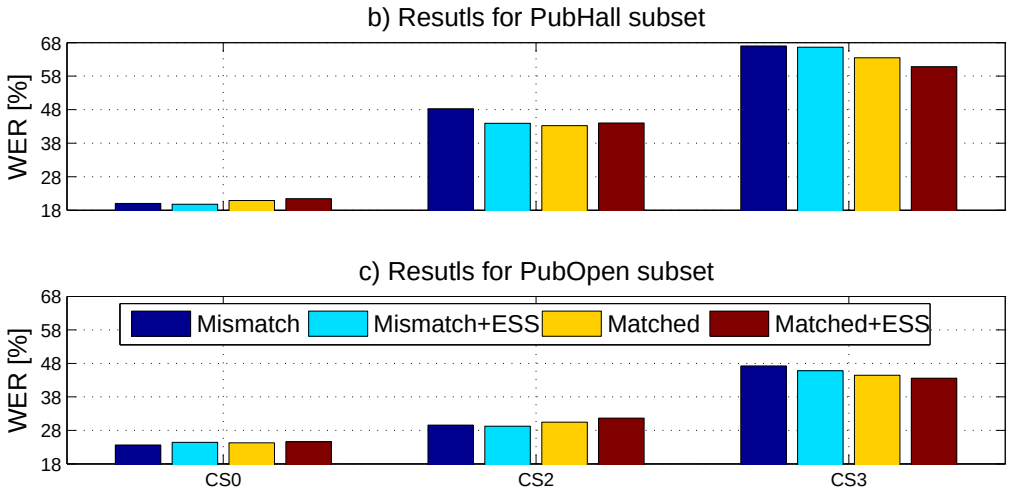

Figure 5.6: Summary for MPE trained AMs for all environments

5.1.4 Summary

The results for the final MPE models for all environments are illustratively summarized in Figure 5.6 and the main conclusions can be summarized as follows.

- The recording from the close-talk microphone achieved good results for all three environments which leads to conclusion that using a close-talk microphone greatly reduces the difficulty of using an ASR system even in noisy environments.
- Regarding the PubOpen environment, CS2 microphone was also found to produce reasonably good recordings for ASR. Also, the contribution of the studied techniques was the highest for this channel. However, the distortions introduced by CS3 channel were too severe. Thus, it can be concluded that using a microphone in more than about 1 m distance from a speaker is not advised for GMM-HMM systems.
- Regarding the PubOpen environment, only the CS0 microphone was found to be suitable for ASR as the convolution noises degraded even the CS2 channel greatly.
- The studied techniques proved to be more robust against the distortions in PubOpen environment than against the distortions in PubHall environment. Given the nature of distortions, the studied AM techniques were more robust against strong additive distortions than against strong convolution distortions.
- The application of SAT and SGMM proved to yield the greatest relative improvement. The MPE showed slightly better results on average than the bMMI criteria. The only exception was the CS3 channel for PubHall environment, where the bMMI performed better, which was also the worst performing subset. Thus, MPE proved to be a more robust discriminative training than bMMI.
- The application of the ESS brought significant and consistent improvements for the far distant CS3 microphone for both public environments and with both matched

and mismatched training. The CS0 did not contribute from its application while the CS2 channel showed improvement only for the mismatched training case.

5.2 Noisy Car Recognition

This section describes the optimization analysis on speech recorded in a running car in different traffic conditions. The analysis also compared three different microphones, a head-set microphone and two middle-distance microphones. The recognition system was built using the common ASR framework described in Chapter 4, while it also included the addition of ESS. The AM trained on Quiet environment data was also used for mismatched recognition of car recordings in order to evaluate generalization qualities of the studied techniques, especially the AM adaptation. The amount of training and evaluation data was not evenly spread among the channels as the CZCKCC database contained much more CS2 and CS3 recordings in comparison to CS0. SPEECON contained the same amount for all channels. Since it was practically unreasonable to train a specific AM for each velocity and driving environment separately, the training was done on the full car set. The information about each setup is summarized in Table 5.12. The evaluation task consisted of a standard LVCSR task with a bigram LM and 340k vocabulary. The evaluation set was divided into three different subsets: car driving in a *City*, *Country* or a *Highway*. The information about the subsets is summarized in Table 5.12.

Table 5.12: Summary of training sets for noisy car environments

	CS0			CS2			CS3		
	City	Country	Highway	City	Country	Highway	City	Country	Highway
Train		89h			188h			108h	
Test	6.5h	1.7h	1.4h	6.8h	6.4h	1.4h	7.2h	1.7h	1.4h
OOV	1.8%	1.5%	1.4%	1.8%	1.6%	1.4%	1.8%	1.5%	1.4%

5.2.1 Acoustic Conditions

The main difference between the analysed conditions was the driving velocity, which was the lowest in the City, higher in Country and the highest in the Highway conditions. The speed directly correlated with the level of the aerodynamic noise as well as the noise made by the running engine. Also, these noises got stronger as the speed increased. The second important factor were additional noises, which are specific for each condition (a tram in a City, a passing car in Country or Highway). Thus, it was reasonable to assume that the City recordings had the most favouring acoustic quality, the Country recordings were degraded by a moderate level of noise and the Highway recording suffered from the strongest degradation.

The SNR histograms for each evaluation set are illustrated in Figure 5.7, 5.8 and 5.9. It can be noted that unlike in the previous analysis, the distributions do not resemble

Table 5.13: Statistical parameters for noisy car ($\mu \pm \sigma$) [dB]

Env.	CS0	CS2	CS3
City	11.8 ± 5.2	10.2 ± 4.7	8.6 ± 4
Country	10.9 ± 5.1	9.7 ± 5.4	8 ± 4.1
Highway	11 ± 5.1	8.4 ± 4.1	6.3 ± 3.4

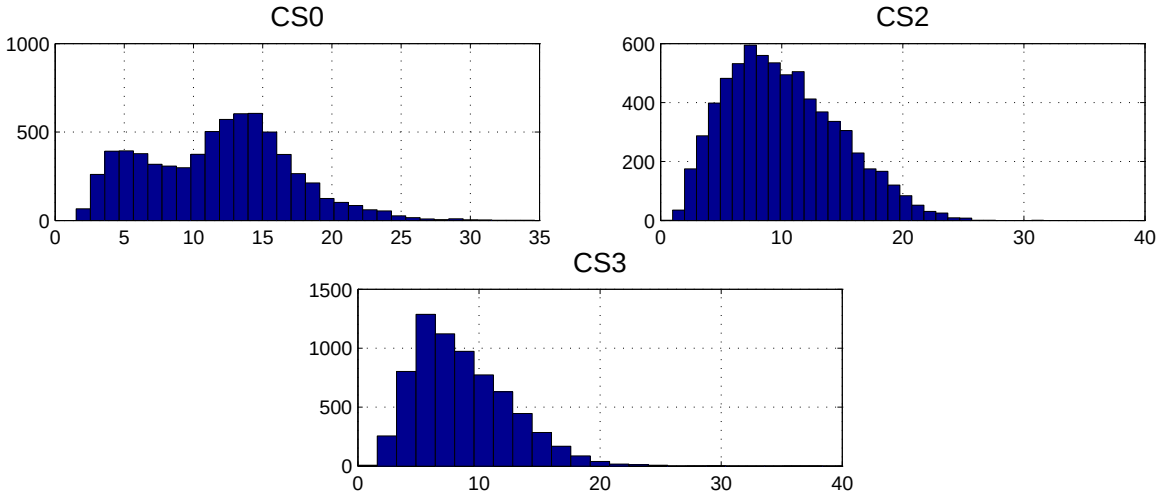

Figure 5.7: SNR histograms for all channels for City car

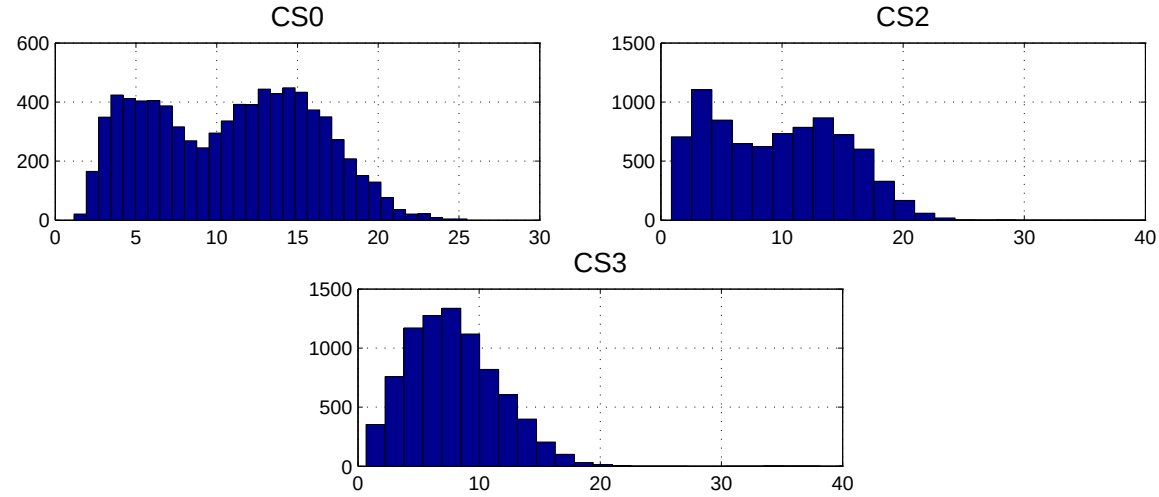

Figure 5.8: SNR histograms for all channels for Country car

the normal distributions. The CS0 channel has a bimodal Gaussian distribution, which gets more pronounced for the Country and the Highway subsets in particular. This behaviour indicated that there were two distinct sources of distortion and each had its own SNR characteristics. It is also interesting to note the corresponding peaks are located at approximately 4 dB and 14 dB levels, regardless of the environment. The 4 dB peak was more narrow and got higher as the driving velocity increased. On the other hand, the 14 dB peak was relatively wide and its corresponding distribution got flatter with increas-

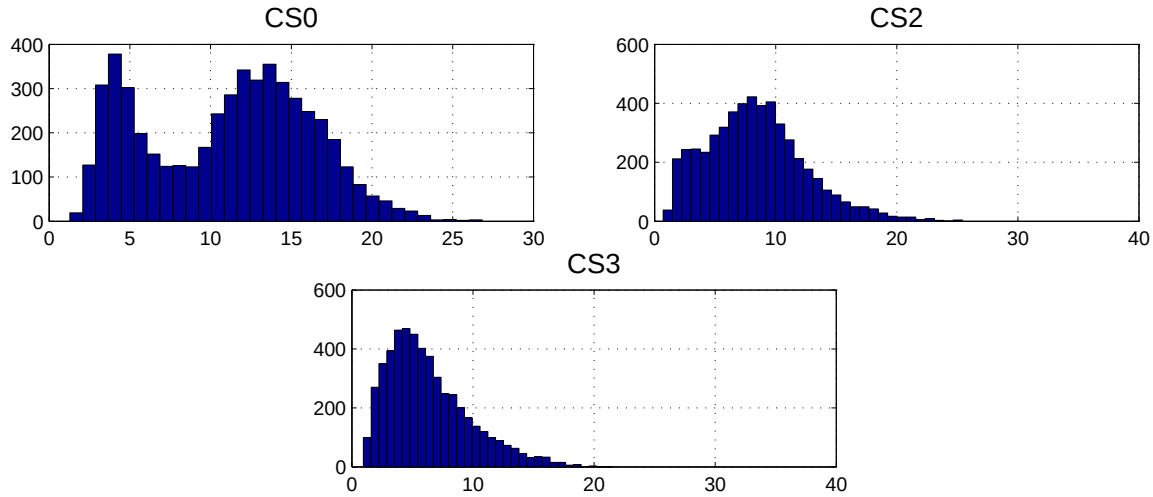

Figure 5.9: SNR histograms for all channels for Highway car

ing velocity. Only the CS3 channel displayed a previously seen lognormal distribution. Table 5.13 summarizes the parameters of the estimated Gaussian distributions. It can be also noted that the estimated parameters were relatively similar across the channels and subsets.

5.2.2 Acoustic Modelling for Noisy Car

The initial analyses evaluated the possibility of using a general AM that was trained on recordings from the Quiet environment, whose performance was evaluated in the previous section. The results for mismatched training for City subset are summarized in Table 5.14. The close talk channel performed at 33.02% with the initial AM and the MPE model performed at 22.23%, which corresponded to 32.7% *WERR*. The distant microphones CS2 and CS3 performed at very similar rates, especially for the final MPE models. The error rate reached 32.56% and 33.12% respectively, which corresponded to 32% and 36.7% *WERR*. This result would indicate that the recordings from both CS2 and CS3 microphone had very similar acoustic quality. The clean AM displayed relatively similar performance gains across the channels in this particular mismatched conditions, but the overall error rates were higher than they were for the Quiet subset. In comparison, the *WER* difference reached roughly 8% for CS0 channel, 16% for CS2 channel and approx. -1% for CS3 channel.

The results for matched training for City subset are summarized in Table 5.15.The results for the *Baseline* system differed greatly between the close talk CS0 and CS3 distant microphone. However, gradual improvements to AM's quality brought the final error rates to a very similar level. The *WERR* reached 32.5%, 42.4% and 47.6% for CS0, CS2 and CS3 channels. Interestingly, the CS3 microphone outperformed the CS2 microphone, even if only by a very slight margin. Thus, it can be concluded the position of the microphone in a car was less important as long as it was not in a close proximity to the

Table 5.14: Results for City subset with mismatched training

			Adapted			
Channel	Baseline	LDA	SAT	SGMM	bMMI	MPE
CS0	33.02	31.63	27.43	23.53	22.37	22.23
CS2	46.44	48.35	41.99	35.51	32.42	31.56
CS3	52.31	54.08	43.69	37.96	34.36	33.12
CS0+ESS	32.93	31.67	27.36	23.89	22.73	22.57
CS2+ESS	45.71	46.58	42.39	34.49	32.39	31.52
CS3+ESS	55.47	55.84	46.35	40.45	36.34	35.15

speaker's mouth. In another words, the distant car microphone simply picked up a lot of noise regardless of of its positions. However, it is also important to realize that these conclusion were reached for the least noisy, City environment. The direct comparison of the studied DT criteria showed that MPE achieved better results overall. The absolute difference in *WER* between bMMI and MPE was more than 1% for for the CS3 channel, nearly 1% for CS2 and a marginal 0.04% for CS0. Overall, this difference was greater for the distant microphones which lead to the conclusion that MPE training criteria was more suitable for the distorted speech recognition with great training-evaluation mismatch.

A consistent improvement with ESS was observed only for the weakly trained *Baseline* AMs. This trend was most notably visible for the matched training and CS2 and CS3 channels, which contain stronger additive distortions. The ESS application on the CS0 channel had either negative or very little positive impact. There was, however, an exception to this rule. The CS2 channel with matched training displayed a consistent and significant *WERR* for all stages of AM refinement. The values ranged from 4.9% for the SGMM up to 9.5% / 8% for the bMMI/MPE techniques. A less significant trend was also observed with this channel for mismatched conditions. The *WERR* for CS3 channel with matched training had a sharply decreasing tendency as it reached 14.6%, 11.4% and 1.9% for the *Baseline*, *LDA* and *SAT* models respectively. The improvements for subsequent refinement techniques models were negative.

Table 5.15: Results for City subset with matched training

			Adapted			
Channel	Baseline	LDA	SAT	SGMM	bMMI	MPE
CS0	31.76	31.14	27.22	23.83	21.41	21.45
CS2	46.65	42.19	34.36	29.48	27.73	26.88
CS3	49.12	44.10	34.27	28.86	26.21	25.74
CS0+ESS	31.50	30.83	27.29	23.95	21.30	21.49
CS2+ESS	43.05	39.46	32.57	28.11	25.33	24.88
CS3+ESS	42.85	39.59	33.64	29.44	27.38	26.60

The Quiet AM recordings were expected to be acoustically similar to the City recordings, especially for the close-talk microphone. The position of the CS0 microphone as well as its directional characteristics produced clean recordings without any significant additive or convolutional noises even in such adverse conditions. The potential mismatch

was also reduced by the application of AM adaptation and other modelling techniques. As a consequence, the final AMs were expected to perform equally well is mismatched and matched conditions. The situation for the CS2 and CS3 channels was, however, more complicated. These channels contained specific types of distortions which was present only in the car evaluation sets and not in the clean training set and only the AM adaptation could lower this mismatch. Thus, it was expected that the error rates would differ greatly for the matched and mismatched conditions.

The performed experiments reveal that the advantage of matched training was statistically insignificant for the close-talk CS0 microphone and thus proved the previously stated hypothesis. The average difference in *WER* for the best MPE trained AM was just 0.8%. Thus, it could be concluded that a general purpose AM performed sufficiently well and it was not necessary to train a car-specific AM. The results were different for the CS2 and CS3 channels where the advantage of matched training was clearly visible. A significant improvement was gained for these channels as the *WER* difference reached 4.7% for CS2 and 7.8% for CS3. There were two factors which lead me to conclude that the studied techniques were fairly robust against distortions present in a slowly moving car in a city environment. First, the absolute difference for *Baseline* AMs between matched and mismatched conditions was only marginal. Second, the *WERR* between the initial and final models had an increasing tendency. Also, these results qualitatively corresponded with conclusions reached for the PubOpen subset, which showed that additive distortions were relatively easy to deal with. These experiments also further proved the significance of domain-specific SAT, SGMM and discriminative training as three methods achieved the highest *WERR* in the matched training, in this respective order.

Table 5.16: Results for Country subset with mismatched training

			Adapted			
Channel	Baseline	LDA	SAT	SGMM	bMMI	MPE
CS0	44.80	41.70	31.82	27.11	25.64	25.47
CS2	50.89	51.55	41.53	34.72	32.69	32.06
CS3	57.41	57.71	45.32	39.22	36.28	35.10
CS0+ESS	44.91	41.41	32.45	27.64	26.27	26.09
CS0+ESS	50.69	49.94	41.83	35.39	33.50	32.44
CS0+ESS	57.14	58.61	48.37	40.72	37.03	35.65

Table 5.17: Results for Country subset with matched training

			Adapted			
Channel	Baseline	LDA	SAT	SGMM	bMMI	MPE
CS0	45.57	44.62	37.34	32.61	26.31	25.92
CS2	50.28	47.20	39.24	34.26	32.14	32.04
CS3	52.32	48.99	40.13	33.90	32.73	31.60
CS0+ESS	45.52	44.36	38.06	32.87	26.93	26.38
CS2+ESS	49.15	46.97	39.65	33.87	32.34	30.98
CS3+ESS	48.11	45.68	39.26	34.44	33.20	32.29

The results for mismatched training for Country subset are summarized in Table 5.16. The initial AMs performed at relatively similar error rates of 44.8%, 50.89% and 57.41% for CS0, CS2 and CS3 channels. The results for the CS0 channel were much worse than for the City subset, but the results for CS2 and CS3 channel were only about 5% worse. This observation showed that the background aerodynamic noise and the engine noise began to heavily influence even the close talk microphone recordings as the speed got higher. The final MPE models performed better than bMMI models and the error rates reached 25.47%, 32.06% and 35.1%, which corresponds to 43.1%, 37% and 38.9% *WERR*. If we compare these results with the City subset we will notice that the *WER* difference reached approximately 3%. These results demonstrated that the studied techniques were still reasonably robust against the car noise with the SNR of about 10 dB. Also, even if the relative improvement across the channels reached similar values, it was still the highest for the CSO, which could be explained as follows. As the acoustic conditions got worse and the amount of noise increased, the relative contribution of described modelling techniques decreased and the channel that contributed the most was the one whose acoustic conditions were most similar with the training conditions.

The results for matched training for Country subset are summarized in Table 5.17. The *Baseline* AMs performed at 45.57%, 50.28% and 52.32% while the final MPE models performed at 25.92%, 32.04% and 31.6%. This corresponded to 43.1%, 36.3% and 39.6% *WERR*. These values were once again very similar to the results for the City subset, both qualitatively and quantitatively. They also further demonstrated the reduced robustness of these techniques for more noisy CS2 and CS3 microphones. The direct comparison of the Country subset with the City subset showed that average decrease of 0.6 dB SNR between these two environment brought about 5% *WER* difference. Interestingly, the usage of matched training proved to be fairly insignificant for Country environment as only the CS3 contributed by 3.5% in absolute. The CS0 and CS2 channel displayed a minimal or even a negative improvement. These results documented the limits of the studied techniques to compensate the introduced distortions in such adverse conditions. However, the application of ESS brought significant *WERR* for distant channels and weakly trained models, which was a trend that has been previously observed for City subset as well. The *WERR* of using the ESS reached 8.8%, 7.2% and 2.2% for the *Baseline*, *LDA* and *SAT* stages and CS3 channel. Also, some improvement were observed for CS2 channel. The improvements for the CS0 channel were once again statistically insignificant for weakly refined AMs and negative for discriminative models.

Table 5.18: Results for Highway subset with mismatched training

			Adapted			
Channel	Baseline	LDA	SAT	SGMM	bMMI	MPE
CS0	44.55	41.88	33.42	28.39	26.98	27.31
CS2	61.96	58.06	50.44	42.25	39.05	39.76
CS3	64.97	63.80	50.86	43.74	41.60	40.60
CS0+ESS	44.15	41.29	32.94	28.09	26.35	26.46
CS2+ESS	58.20	56.99	50.97	42.16	39.99	38.92
CS3+ESS	64.72	65.53	53.82	45.98	42.13	41.43

Table 5.19: Results for Highway subset with matched training

			Adapted			
Channel	Baseline	LDA	SAT	SGMM	bMMI	MPE
CS0	44.61	44.80	36.96	31.48	27.24	27.64
CS2	59.19	58.56	48.14	39.44	38.99	38.70
CS3	57.94	55.16	47.95	39.25	38.53	38.46
CS0+ESS	43.98	44.24	37.20	31.57	27.47	27.64
CS2+ESS	57.94	54.24	46.28	39.68	37.85	37.45
CS3+ESS	57.94	55.16	46.52	39.64	37.77	37.45

The results for mismatched training for Highway subset are summarized in Table 5.18. The results for the *Baseline* AMs reached 44.55%, 61.96% and 64.97%, which was actually the highest measured *WER* for CS2 and CS3 channels. However, the results for the CS0 were in fact slightly better than for the Country subset. The final MPE trained models achieved the error rates of 27.31%, 39.76% and 40.60%, which corresponded to 38.7%, 35.8%, 37.5% *WERR*. The comparison of these results with the City subset showed that the additional 0.8 dB SNR caused 5.1% performance drop for the CS0 channel, additional 1.8 dB SNR caused the 8.2% drop for the CS2 and the additional 2.3 dB SNR caused the 7.5% drop for the CS3 channel. Simply speaking, the levels of noise present in the car riding a highway degraded the recording much more than in other subsets. This increased mismatch was clearly degrading the ASR performance but it also showed that the AM adaptation was the best source of improvement.

The results for the matched Highway subset are summarized in Table 5.19. The results displayed the trend of very similar results between the CS2 and the CS3 channel. Also, the CS0 channel performed at very similar error rate with the Country subset. One interesting thing, however, was the comparison of the relative improvement with matched training for SAT across subsets. The analyses have documented a decrease in its capability to improve the AM quality as the degradation got stronger. The *WERR* for CS3 channel reached 22.3%, 18.1% and 13.1% for the City, Country and Highway environments respectively. This observation would suggest that the fMLLR adaptation was losing its capability to create a generalized AM. A very similar trend could also be observed for the discriminative training when the highest *WERR* were also observed for the City environment on average, then the Country and finally the Highway. Finally, the contribution of the ESS was proved only for the CS2 channel. This conclusion was consistent with my previous findings from the distant microphone recognition in the Quiet and Public environments as notable improvements were observed for the middle distance CS2 microphone.

5.2.3 Summary

The results for the final MPE models for all car environments are illustrated in Figure 5.10 and the most important conclusions can be summarized as follows.

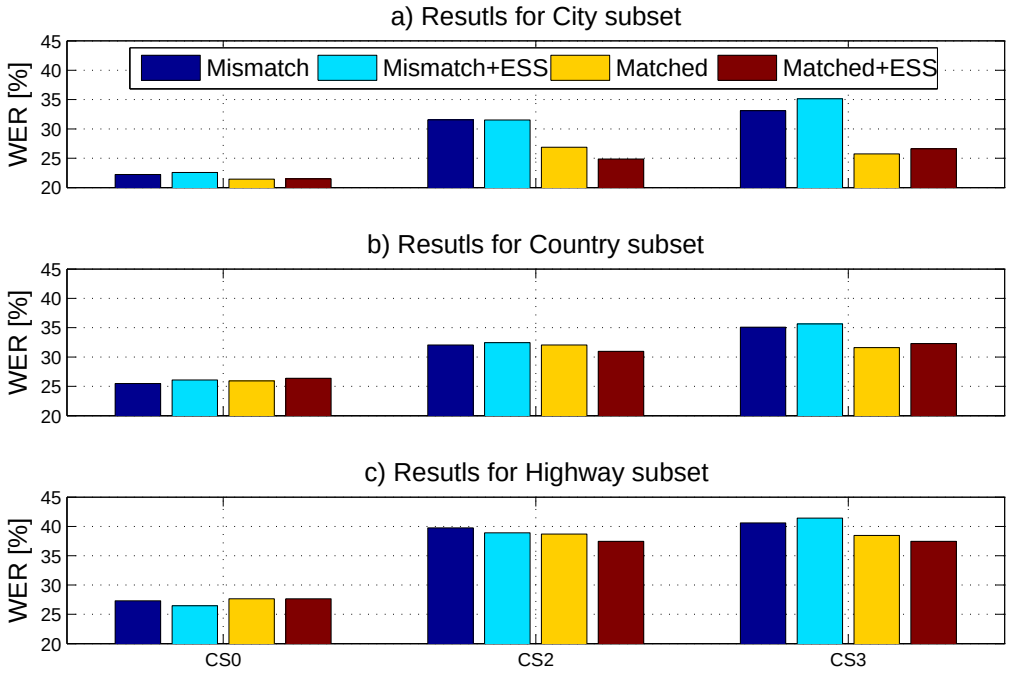

Figure 5.10: Summary for MPE trained AMs for all car environments

- The usage of general purpose AM trained on quiet environment recordings achieved worse results than the car environment trained AMs. This difference was more pronounced for CS2 and CS3 recordings. However, there was very little difference for the close-talk microphone. This observation supported the conclusion that close-talk microphones with a good directional characteristics were suitable even for noisy environments.

- Out of the two studied DT schemes, the MPE yielded slightly better results on average. This conclusion held true for matched and mismatched training, which demonstrated that MPE had better generalization capabilities on unseen acoustic conditions than the bMMI criteria. However, its generalization capabilities decreased with the increasing training-evaluation mismatch.

- The highest *WERR* were achieved with the application of SAT, but it also displayed a tendency of a decreasing *WERR* as the degradation got stronger in Country and Highway subsets. The relative improvements of SAT in mismatched training stayed roughly the same for all subsets.

- The application of ESS was proved to have a consistent and significant improvements for the CS2 channel and the Country subsets. The lightly degraded City subset did not contribute from its application at all while the Highway subset was degraded too much and the improvements were much less on average.

CHAPTER 6

COMPRESSED SPEECH RECOGNITION

MP3 compression exploits the deficits of human auditory system which is capable of distinguishing among individual sinusoidal components in a complex harmonic signal by performing a sort of Fourier analysis with a limited spectral and temporal resolution. In practice, these two physiological limitations result in situations when one sound (maskee) is rendered inaudible in the presence of another sound (masker). This situation is commonly called the auditory masking [76] and the psychoacoustics distinguishes between two different types of masking, each caused by its own mechanism.

- **Simultaneous** - The masker is present for the whole time the maskee is present. The phenomena is dependent on the relative difference between the masker's and maskee's frequency. If the frequency difference is too small, the sounds reside in the same critical band and can't be distinguished from each other.
- **Nonsimultaneous** - The masker occurs before or after the maskee, denoting the situations as forward and backward masking respectively. The forward masking is caused by the fact that the auditory system's sensitivity is reduced directly after the termination of a sound. The generally accepted explanation is that either the neural response is suppressed after the stimulation or the neurons stay adapted to the first sound and cannot immediately tune to a different sound. Very little is known about the backward masking, but it has been shown that people can learn to suppress it.

Using this knowledge, MP3 omits the perceptually "irrelevant" information from the audio, basically the sounds which can't be heard anyway, to save the data. While the designers of the algorithm placed strong requirements on preserving the high-fidelity of the output audio for subjective listening tests, the nature of the coding is still lossy

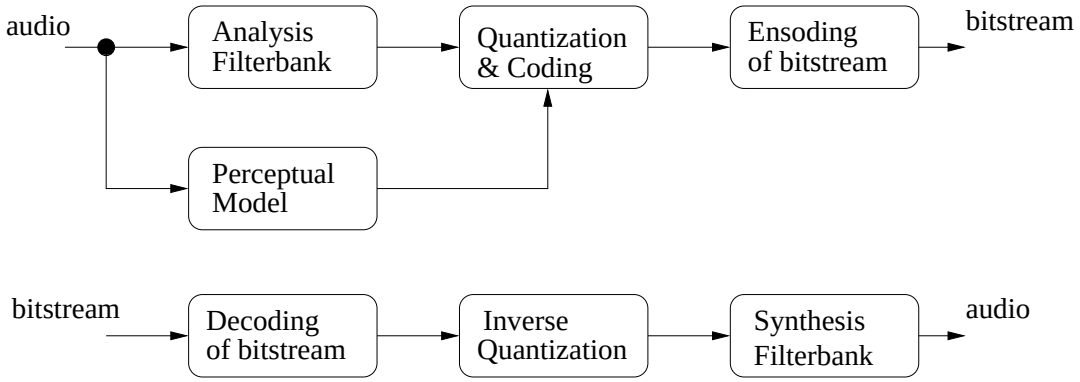

Figure 6.1: Block diagram of MP3 coder/decoder

and introduces multiple distortions [77, 78]. The general effects of PAC schemes have been studied in the literature from both the objective and subjective points of view. Several official materials and guidelines have been published on subjective listening tests, see [79, 80, 81]. These materials specify numerous conditions for testing, evaluation and reporting procedures, define the artifacts to listen for, sort them according to their severity and type or simply provide results for carried out tests. This topic was not explored further as the purpose of subjective tests is to study the perceived quality by a listener, a task which plays only marginal role for ASR and is not necessarily correlated with the actual recognition accuracy. However, the conclusions reached within these works could be extended to the field of ASR as well as the application of auditory masking functions and the quantization resulted in the distortion of the signal's spectra. Some of the artifacts were only perceptually identifiable while others directly influenced the quality of extracted features and could be easily identified using the spectral analysis. They have been shown to also influence the estimations of basic speech characteristics such as pitch and formants frequencies [82]. The following text studied two main distortions:

- bandlimiting due to the application of a low-pass filter,
- unnaturally deep spectral valleys (also called gaps), which are frequency bins with a very low energy.

Figure 6.2 plots the logarithmic spectrum in a 32 ms frame from a 16 kbps coded signal to illustrate the effects of both distortions. The bandlimiting occurred for f greater 5600 Hz and the flat areas with the central frequencies at 1400 and 3300 Hz were the spectral valleys. The width of the first valley was about 250 Hz while the second valley was more than 1000 Hz wide. The next sections are devoted to the analysis of the above mentioned artifacts in great detail as well as their effects on the ASR blocks of the front-end processing and the AM training. The focus was to utilize objective methods for analysis and to present conclusions which could be directly related to the particular blocks of a recognition system and the output accuracy. The design of the proposed compensation technique was derived from the conclusions reached in this section.

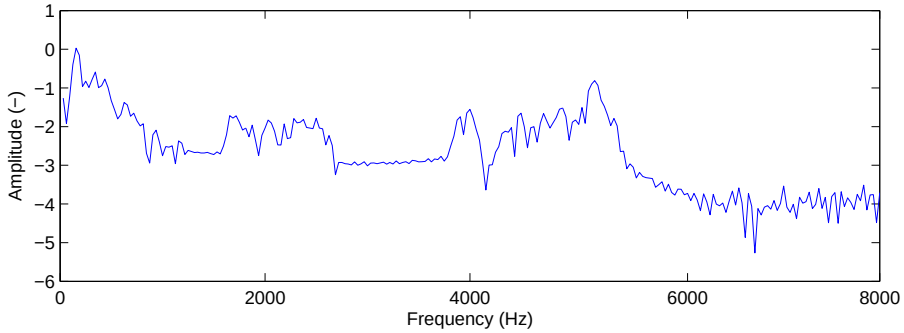

Figure 6.2: Logarithmic spectrum of a frame distorted by MP3 coding.

6.1 Related Works

All previous studies on practical usability of ASR for MP3 recordings have commonly concluded that the GMM-HMM systems can work with little difficulties if sufficiently high bitrate is used. The authors agreed that the bitrate of 32 kbps represents a threshold after which the accuracy starts to drop rapidly [83, 84, 85, 86]. Several solutions to the discussed problems have been proposed and experimented with, starting with limiting the training signals bandwidth, using PLP features for bitrate-specific AMs, adding a controlled amount of noise [87] or using DNN-HMM architecture [88].

The contribution of PLP features for the ASR system built for compressed signals has been reported in [86], where the authors trained AMs specifically for each bitrate from the pre-compressed database. The main advantage of PLP over MFCC features was the fact that the error rate of the system did not rise as significantly for lower bitrates. This work reported about 40% absolute difference in *WER* between MFCC and PLP features in a simple digit recognition task for 8 kbps bitrate.

The first attempt to account for the problem of bandwidth limitation which avoided the process compressing the whole training set was presented in [83]. The authors proposed and practically tested a feature extraction scheme where the training data was low-pass filtered on cut-off frequencies which corresponded to each bitrate. The purpose was to train AMs that were better matched against the testing conditions and the results showed an absolute decrease in *WER* of about 1-2%.

The main issue with spectral valleys is that only a part of training data for each speech unit is likely to be affected by it and even then not always the same bins. Authors in [87] employed a pre-processing scheme where signals were dithered by a controlled amount of noise to "fill in" these holes. The amount of added noise was estimated manually and then interpolated using a logistic regression from the observed trends. The results showed that the application of this technique could bring significant error reduction, about 45% absolutely for 16 kbps and MFCC features. In general, better results were obtained for lower bitrates, while the results for higher bitrates were only slightly compromised due to the introduction of the noise.

A more recent study in [88] compared the DNN-HMM and DNN-HMM architectures for MP3 recognition. The DNN-HMM without any feature-level compensation outperformed the GMM-HMM system by more than 25% for 16 kbps coded speech. The histogram equalization method for MFCC features was found to be particularly effective for the HMM-GMM system. Also, the application of AM adaptation in the form of CMLLR served very well, when the absolute *WER* dropped to values comparable with a DNN-HMM system. These findings are of particular interests for several reasons. First of all, the DNN acoustic models were able to outperform the GMM models for MP3 recognition. Second, the application of proper pre-processing methods at the level of feature extraction was found to increase the accuracy for GMM-HMM systems, but brought mixed results for DNN-HMM. Third, AM adaptation was found to perform very well as a MP3 compensation technique, while also had the added benefit of not being MP3 specific.

6.2 Bitrate detection

The cited works pointed out that the key issue in matched training for real-life ASR is the precise bitrate detection. A simple SVM classifier based on energy values from narrow frequency bands [89] was shown to achieve 97% accuracy in detecting bitrates greater than 128 kbps. This work also showed that transcoding a lower bitrate into a higher bitrate does not effect the detection accuracy. The problem of double compression was studied in [90], which analyzed both the up-transcoding and down-transcoding scenarios for bitrates in the range from 192 kbps to 64 kbps. The results showed significant differences in detection accuracy, depending on whether the signal was down- or up-transcoded. The classifier achieved 100% accuracy in the case of up-transcoding 64 kbps→192 kbps but only 61.8% accuracy for down-transcoding 192 kbps→64 kbps. The work of [91] followed up on using different encoders in each step and found a small difference in detection accuracy. It is interesting to note that these results qualitatively corresponded with the findings in [87], but the results for using different encoders were not nearly as significant. It is also important to note that the use of different encoders is not that rare as it may seem. It often occurs in broadcast archives where some speech segments were compressed for telephone transmission first and later re-compressed for archivation.

6.3 Effects of MP3 on the ASR

Previous chapter on distant and car speech recognition has established that recognition in adverse conditions puts an increased demand on ASR design as more robust pre-processing and AM refinement algorithms are required in order to compensate the introduced distortions. While some types of signal degradation can be hardly avoided, there are distortions that have been introduced unintentionally and this fact proves to be especially true for speech compressed by a lossy compression. The following section analyses the effects of MP3 compression on the quality of the speech wave in time and spectral domains, the extracted features and finally on the trained AM.

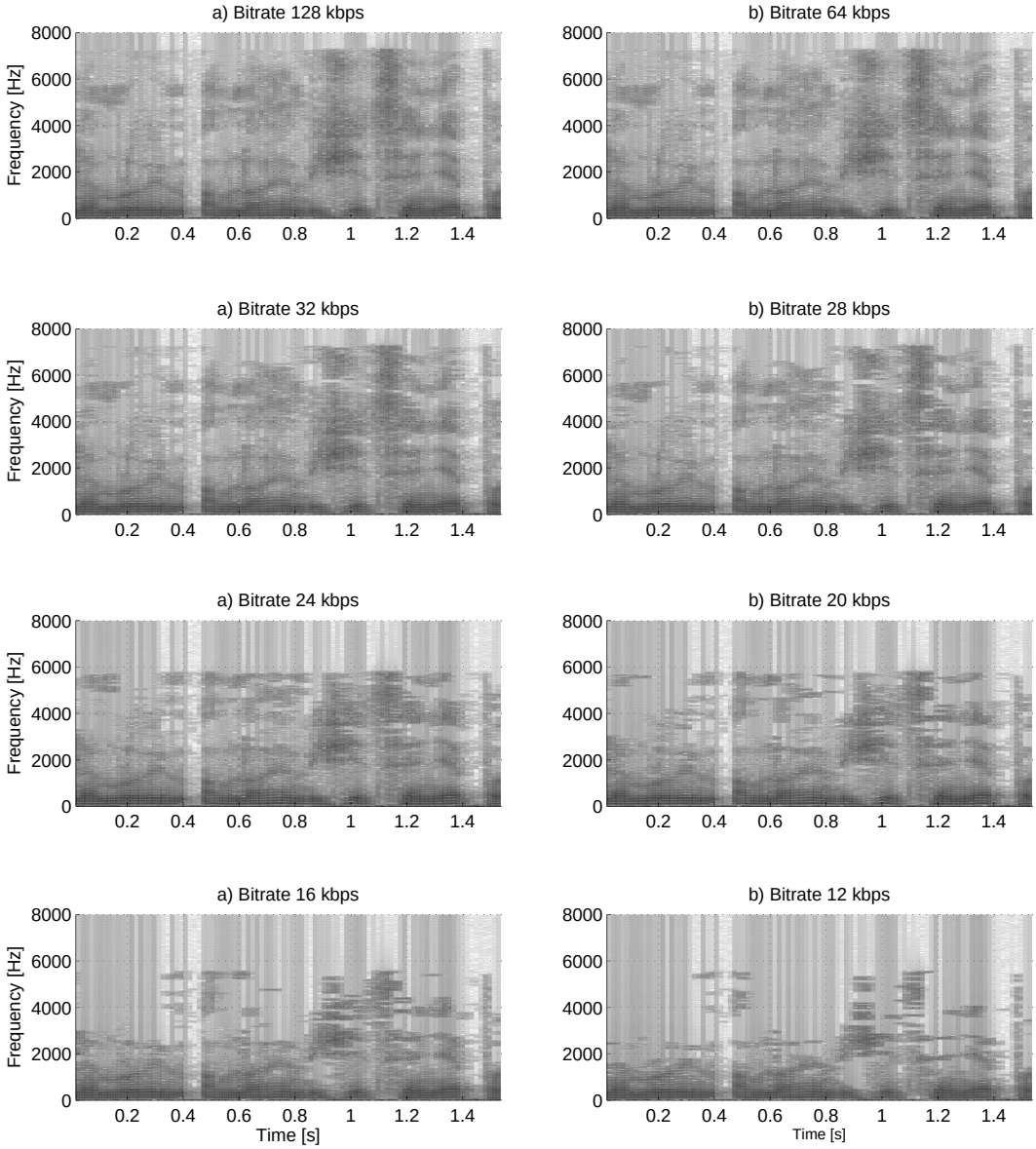

Figure 6.3: PSD estimation for the same signal compressed with various bitrates

6.3.1 Effects on Speech Wave in Time and Spectral Domain

The set of pictures in Figure 6.3 illustrate the power spectral density (PSD) estimated from the compressed speech. The plotted spectrograms contain a full sentence of read speech in Czech, sampled at 16 kHz sampling rate and coded in 16 bit precision. The PSDs were computed from the same utterance for both the uncompressed and MP3 speech using Welch methods for 32 ms frames with 50% overlap and 8 frame averaging. The figures once again documented the degrading effects introduced by MP3 compression and also demonstrated that these effects were becoming more severe with a decreasing bitrate.

Bandlimiting

Table 6.1: Summary of the LP cut-off frequencies as reported by LAME

bitrate [bps]	(Inf.-28k⟩	(28k-20k⟩	(20k-12k⟩
f_{cut} [Hz]	no LP filt.	5750-5950	5500-5700

The MP3 format actively narrows the spectral bandwidth as the bitrate decreases in order to improve the subjective quality after the compression. Table 6.1 summarizes the cut-off frequencies as they are reported by LAME coder. It is interesting to compare these values to the spectrograms displayed in Figure 6.3 and it is worth noticing that the bandlimiting occurred even though the block of LP filtering was disabled by encoder for bitrates greater than 28 kbps. The lowest cut-off frequency of about 5600 Hz was observed for the 16/12 kbps bitrates, and a slight bandlitmiting at about 7200 Hz was observed even for bitrates greater than 28 kbps. The severity of the effect increased with decreasing bitrate and it should therefore be, at first glance, of concern only for rates lower than 28 kbps, if at all. The generally accepted consensus on the position of formant frequencies for Czech postulates that none of the generally estimated ones (F1–F3) occur at frequencies higher than 5600 Hz [74]. Therefore, their estimation should be robust against lossy compression. However, this may not be the truth. Son [82] analyzed the precision of f0 and formant estimation for the Vorbis and MP3 coders at 40 kbps, 80 kbps, 192 kbps rates for Dutch. The author demonstrated the formant estimation error was less than 3% for bitrates higher than 80 kbps, but concluded that the estimation for 40 kbps rate was unsuitable due to a markedly larger error. The author also concluded that f0 computation was largely unaffected by the compression.

Table 6.2: Error of f0 estimation for various bitrates

bitrate [kbps]	160	64	32	28	24	20	16
f_{cut} [Hz]	7200	7200	7200	7200	5800	5800	5600
Δf [Hz]	0.04	0.05	0.06	0.1	0.1	0.1	0.1

In order to verify these results reported in [82], I did a precision analysis of f0 and formants estimation for Czech. The obtained results are presented in Table 6.2 for f0 and in Figure 6.4 for formants. The measurement was done with Praat using the cross-correlation method. The reported values represent an average difference in f0 between PCM quality and compressed signals. There are several conclusion which could be drawn from this experiment.

- The results proved that f0 estimation was generally very robust against bandlimiting regardless of the bitrate. The absolute error was within the 0.1 Hz range, which could be considered as statistically insignificant. It is important to note that using cross-correlation method of f0 estimation was influenced by the measurement error which occurred due to the fixed windowing. However, it could be assumed the introduced error was random and thus had zero mean for a large set measurements.

- The error of formant estimation was much more significant. My results showed only a slight error for F1 (less than 30 Hz), which corresponded to about 7% relatively. On the other hand, the estimation of F2-F3 was much more erroneous as the absolute difference reached 370 Hz and 720 Hz respectively, which meant nearly the same 22% relatively. It can be also said that the relative error for F2/F3 increased by a factor of 3 when compared to F1.

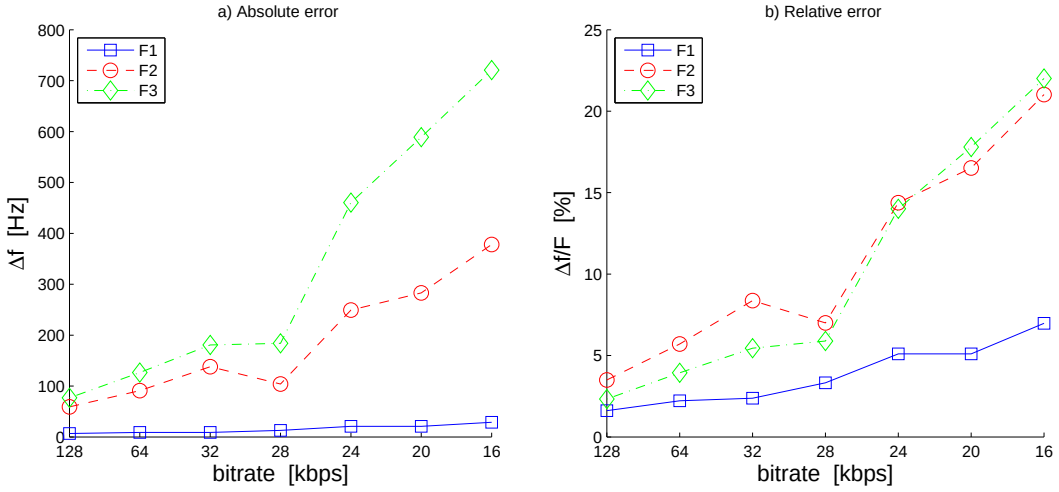

Figure 6.4: Development of the formant estimation error, Δf [Hz] for various bitrates

It is important to note however, that these results were obtained using automatic measurements for speech segments cut-off from the signal by an energy-based voice activity detector with an adaptive dynamic threshold. The output formant listing from Praat was further post-processed by applying the moving average filter to smooth the formant contours. The uncompressed quality signal was always used as reference and assumed to give correct results. Although this might not have held true for all cases, the upward trend in error was apparent nonetheless.

The immediate concern this effect raised was the loss of content carried by frequencies grater than f_{cut}, which provide the major source of information for signals with rich high-frequency (HF) components such as unvoiced consonants. Thus, it was reasonable to assume that the partial error rate for unvoiced speech units would increase more rapidly than for voiced units, which would in turn steeply increase the overall error rate. On the other hand, the voiced speech units (vowels and voiced consonants) have a strong harmonic structure at low frequencies which should make them more robust against this type of distortion.

The bandlimitation problem is common for all PACs and have been known since the introduction of MP3 in the nineties. Therefore, the audio coding research has been focused on improving the subjective quality of coded signals by reconstructing the missing bands. The algorithms belonging to this group are generally called *Artificial Bandwidth Extension* or *Spectral Band Replication (SBR)*, see Liu [92], Hsu[93], Diet [94] or Arora [95]. The core idea is the assumption that higher frequency bands are in fact redundant and the information they carry can be inferred from the lower bands. The common principle is to

extrapolate the spectral envelope of the filtered bands from the lower bands and to shape its contour by a frequency dependent function. In fact, SBR [94] is the core of Advanced Audio Coding (AAC) and is considered to be the main reason why this new coder sounds better than its predecessor MP3. However, there are no surveys on its performance in ASR systems, to the best of my knowledge.

Spectral Valleys

The application of a psychoacoustic model creates artifacts which are easily identifiable in a spectrogram as almost zero energy areas at low and middle frequencies. These artifacts are referred to as "spectral valleys (SV)" [77] or "spectral holes" [87] in the literature. Following the results presented in Table 6.2, the analysis on their nature was done only for bitrates of 28 kbps and lower as the frequency error started to rise rapidly after passing this threshold.

Figure 6.5 plots the spectrum of a speech segment compressed by low bitrates. The first valley spans progressively larger frequency bands, always beginning at 2500 Hz and continuing up to 3700 Hz for 28 kbps, 3800 Hz for 24 kbps, 3800 Hz for 20 kbps and finally 3900 Hz for 16 kbps. The following peak represents the 3^{rd} formant, that is located right after the valley at the 4000 Hz. The second SV once again begins at 5600 Hz for all bitrates and gradually increases its width until it spans the whole frequency range up to 8 kHz for the 16 kbps. It can be reasoned that the noticeable difference in formant estimation accuracy was primarily caused by this phenomenon.

The most common formant estimation method relies on a linear prediction coding (LPC) of the speech, which uses the AR model to characterize the vocal tract as a concatenation of resonance filters, as has already been explained in detail in Chapter 2. However, the compression destroys this all-pole characteristics as the zero-energy areas often occur right next to the formant peaks. Figure 6.5 illustrates the shift of the formant peaks in a segment of speech estimated by the LPC model of the 12^{th} order. The noticeable characteristics of the spectral envelope for MP3 impaired speech is that the curve follows the valleys which precede the formants. As a result, the AR model incorrectly estimates the position of the peaks on the frequency axis.

It can be also noticed that the LPC algorithm completely failed to detect the 4^{th} formant for the lowest bitrate, simply because the coder zeroed all frequency bins higher than 4600 Hz, even if the position of this formant was clearly visible in the raw signal spectrum. These findings further support the previously presented experimental analysis. Another important conclusion is that the order of the AR model greatly influenced the estimation. The lower order models, e.g. (8^{th}) order which is generally used for PLP features, are more prone to completely ignoring the higher formants if there is a valley preceding it. This observation lead me to conclusion that PLP features for MP3 recognition should be computed with a higher order LPC analysis.

This phenomenon also tends to reach into much lower frequencies than is the f_{cut} as higher frequency bands have already been cut-off by the low-pass filter. Unlike bandlim-

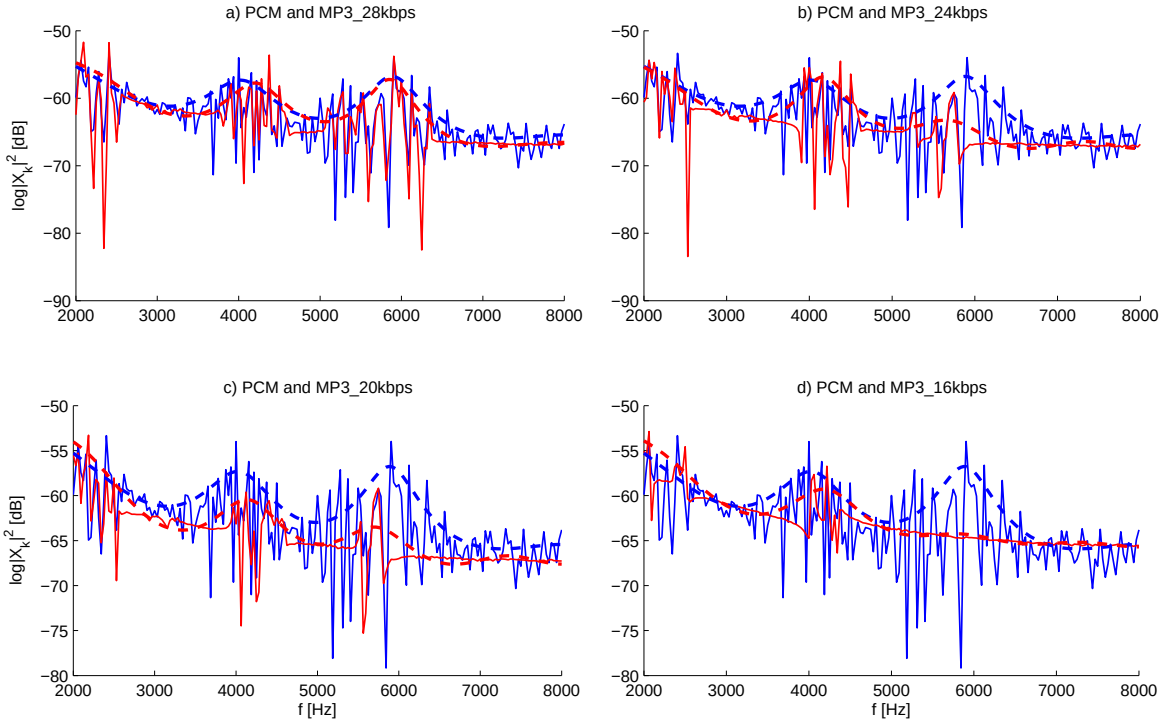

Figure 6.5: The illustration of the negative effect of compression on LPC spectrum and the estimation of formant peaks for compressed and PCM speech. Full line is the signal itself and dashed line is the LPC spectrum.

iting, the exact nature of spectral valleys is also highly context dependent, which makes it statistically impossible to predict the affected frequency bins. It often influences only some parts of speech (usually starts and ends of continuously uttered phrases) and some phonemes. It makes spectral features, and those that build on them (e.g. cepstral ones), less reliable. Its detrimental effect on formants estimation was the primary reason the results in 6.4 did not include the estimation of F4 as the error was simply too high. On regular occasion, the algorithm could not even detect the presence of F4 while no such problem occurred for uncompressed speech.

6.3.2 Effects on Cepstral-based Features

The standard acoustic features for GMM-HMM/DNN-HMM systems are derived from the short-time estimation of power spectra, either in the form of MFCCs, PLPs or Mel-frequency energies, and then concatenated into feature vectors. For further statistical processing in a GMM model, the basic vectors are mainly low-dimensional, decorrelated, and they ignore the time context outside the extraction frame. It can be assumed the estimation of speech features is also influenced by the degradation described in the previous sections, which in turn creates a problem of innate degradation to the system. The correlation between the degradation in power spectra and the degradation in spectra-derived

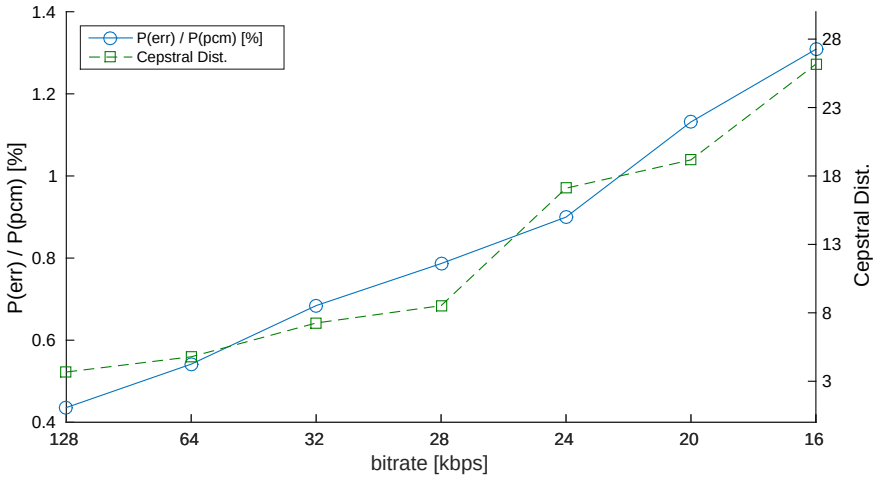

Figure 6.6: Relative power ratio [%] and cepstral distance for static parameters for decreasing bitrates [kbps]

features can be illustrated on the trend of cepstral distance (CD), as it is shown in Figure 6.6. The CD was computed as the standard Euclidean distance in multidimensional space and can be interpreted as a measure of similarity: the more two signals resemble each other, the closer is their position in space and vice versa. The relative power ratio of the error function increases with decreasing bitrate, which can be subsequently extrapolated to the cepstral domain as the increase in distance between the reference uncompressed and compressed signals.

The problem of frequency-band narrowing influences the ASR mainly due to the loss of information carried by higher frequencies and due to the mismatched filter-bank for feature extraction. If we consider the standard feature extraction scheme for short-time spectral features, we can conclude that the distortion will effect always the same higher cepstral coefficients regardless of the neighboring context. The filter-bank range is usually set to the upper limits of available spectra, i.e. $f_s/2$, in order to extract as much information out of the signal as possible. But in the case of compressed speech, the upper frequency limits are zeroed, and the energy extracted by the filter-bank in these bands is equal to zero as well. As a result, the representations of the acoustic units in the feature space are shifted and the resulting AMs trained on these features are mismatched against the standard AMs.

The spectral valleys problem is closely tied to the commonly used feature vectors which contain the temporal information from the preceding and following frames. Authors in [87] theoretically analyzed the effects of spectral valleys on extracted features and showed that the MP3 coding displaced the positions of features in the cepstral domain and significantly increased their variances. The latter has a large impact on the dynamic and acceleration coefficients. Since their computation in a frame is dependent on the values from neighboring frames, the areas of low energy act as a sudden step change and can dramatically increase the resulting values. The main difference between badnlimitation and spectral valleys is that while the former affects mainly higher order cepstral coefficients, the later

can affect much lower coefficients. Another important fact to realize is that spectral valleys are more or less random in nature and they create a serious problem as we proceed further into ASR architecture.

Another problem that occurs when a general purpose ASR is used to decode compressed speech is due to the mismatch between training and testing conditions. Although it might seem tempting to solve this problem by training AMs on compressed signals, there are several drawbacks to this strategy. Each bitrate is assigned its own low-pass cut-off frequency, and thus this strategy would require training bitrate-specific (matched) AMs. Moreover, a bitrate signal detector would have to precede the standard ASR scheme and we have shown that it is not an easy task in certain situations. We also have to remember that there is no strict specification for MP3 coder only the suggestions on its parts and any interested party is free to use their own implementation. This creates a situation when there are many encoders available on the market; i.e. LAME [64], mp3Pro [96] or iTunes; with perceptually different audio quality on the output [97]. Thus, we can safely assume that a signal coded at the same bitrate but with different coders will differ. Authors in [87] have shown that ASR accuracy achieved with two same (low) bitrates but different encoders may differ by more than 40% absolutely. This would suggest that matched AMs would have to be not only bitrate-specific, but coder-specific as well.

When we take into account the degree of trouble involved in accurately detecting real compression rates in certain setups and a differing audio quality for different encoders, we have to reach the conclusion that matched training for each bitrate is not a preferable solution for an every-day system. Nonetheless, I still investigated this option in the experimental part of this thesis in order to compare its performance against the unmatched conditions to see whether the potential improvement was worth the additional computational and design effort.

A multitude of other artifacts (pre-echo, tonal spike, noise amplification, etc.) are introduced in conjunction with the ones already mentioned, but since most of them are easily identifiable only through listening tests, they will not be explored further. A more thorough overview can be viewed either in [79] or in [77], which also includes the list of applicable compensation methods.

6.4 Basic Front-End Optimization for Digit Task

This section presents the initial results of MP3 recognition with a baseline ASR system. The analysis is focused on the contribution of various feature extraction setups and the application of feature normalization for PLP-based systems. The first series of experiments were focused on determining the influence of frame length and frame shift for compressed data and to compare the results with non-compressed data. For this purpose, individual AMs were trained for each setup and their quality was evaluated in the task of isolated digit recognition.

The analysis was done for three compression rates: 160 kbps, 32 kbps and 16 kbps.

The PLP features were computed using CtuCopy for different frame lengths and shifts. The purpose was to determine the effect of frame length and shift for the compressed data in a LVCSR task. The training set consisted of 51 hours of speech and the feature extraction used 13 PLP plus dynamic and acceleration coefficients. Normalization was applied for each speaker separately. The signal modifications can occasionally result in a sequence of zeros in the time domain which, if not treated properly, can cause the extraction algorithm to fail. Since the standard procedure to avoid infinite values in logarithmic spectra is to add small amounts of uniformly distributed noise, all signals were dithered with a uniformly distributed random values from $< -1, 1 >$ interval.

The AM was trained without any advanced refinement technique, with identical state-tying conditions for each bitrate, and the final models contained about 60k Gaussians. The digit recognition task was performed on 15 minutes of speech and used a simple unlimited loop zero-gram grammar. The whole system was constructed using the HTK toolkit and *HDecode* decoder.

6.4.1 Results for Reference System in Digit Task

Table 6.3: Results for RAW data with different window lengths/shifts [ms] and normalization schemes: a)Non-Normalized; b)CMN; c)CMVN

A	*Shift*					
Length	8	10	12.5	13	15	16
16	6.33	-	-	-	-	-
25	4.31	2.29	2.42	-	-	-
30	4.71	2.29	-	-	2.69	-
32	5.52	2.69	-	2.56	-	2.56

B	*Shift*					
Length	8	10	12.5	13	15	16
16	2.02	-	-	-	-	-
25	2.15	1.88	1.88	-	-	-
30	1.48	1.62	-	-	2.29	-
32	1.75	1.88	-	2.02	-	1.75

C	*Shift*					
Length	8	10	12.5	13	15	16
16	5.38	-	-	-	-	-
25	4.98	3.23	1.35	-	-	-
30	3.9	3.5	-	-	1.62	-
32	2.69	1.75	-	2.15	-	2.02

The presented results from non-compressed data proved the overall advantage of using CMN and CMVN techniques. The recognition results for a CMN technique were fairly similar across the setups. The commonly used values of 25 ms window length and 10 ms shift and 32 ms length and 16 ms shift achieved good results in general, but not the overall best. The average relative improvement of CMN for all parametrizations was 38%, while the four highest values of improvement belonged to parametrizations with the shortest shift (8 ms). These findings could be attributed to the fact that these window shifts achieved initially higher *WERs*. The average relative improvement of CMVN was 11%. The best overall error rate of 98.65% was achieved for 25/12.5 ms setup with CMVN normalization.

6.4.2 Results for Compressed Speech in Digit Task

The next analysis was focused on data compressed with different bitrates when the methodology remained the same. The 160 kbps compression rate is generally considered to be high enough to not distort the speech wave in any significant way. The 32 kbps and 16 kbps bitrates should distort the signal more severely, with 32 kbps being the breaking point after which the *WER* starts to drop rapidly. The purpose of this analysis was to find the optimal window setup and to investigate the contribution of normalization schemes for these three bitrates. The results are organized into the tables as in the previous section.

Table 6.4: Results for 160 kbps data with different window lengths/shifts [ms] and normalization schemes: a)Non-Normalized; b)CMN; c)CMVN

A	*Shift*					
Length	8	10	12.5	13	15	16
16	5.92	-	-	-	-	-
25	7.13	5.11	3.1	-	-	-
30	7.67	5.65	-	-	2.29	-
32	8.08	4.04	-	2.56	-	1.62

B	*Shift*					
Length	8	10	12.5	13	15	16
16	3.23	-	-	-	-	-
25	3.1	2.96	1.75	-	-	-
30	2.69	2.56	-	-	2.15	-
32	3.1	2.29	-	1.75	-	1.08

C	*Shift*					
Length	8	10	12.5	13	15	16
16	2.15	-	-	-	-	-
25	2.42	1.75	1.08	-	-	-
30	2.69	2.29	-	-	1.48	-
32	2.96	2.69	-	1.21	-	0.94

Table 6.5: Results for 32 kbps data with different window lengths/shifts [ms] and normalization schemes: a)Non-Normalized; b)CMN; c)CMVN

A	*Shift*					
Length	8	10	12.5	13	15	16
16	6.19	-	-	-	-	-
25	4.71	2.02	2.02	-	-	-
30	5.38	2.29	-	-	1.62	-
32	6.33	3.23	-	1.62	-	1.62

B	*Shift*					
Length	8	10	12.5	13	15	16
16	2.29	-	-	-	-	-
25	1.35	1.35	1.62	-	-	-
30	1.35	1.62	-	-	1.35	-
32	1.88	1.75	-	1.35	-	1.35

C	*Shift*					
Length	8	10	12.5	13	15	16
16	2.56	-	-	-	-	-
25	2.15	1.62	1.21	-	-	-
30	1.75	1.08	-	-	1.35	-
32	2.02	1.21	-	1.48	-	0.94

The window length and step proved to play a large role in the overall quality of the trained AM, especially if the data were not normalized. The shorter window lengths coupled with shorter shifts achieved worse results in general. The trend of a decreasing *WER* for increasing shift was apparent for all setups and normalization schemes. Especially notable values highlighting this fact were 8% absolute increase for 32/16 ms vs. 32/8 ms setup for 160 kbps or over 10% for 25/12.5 ms vs. 25/8 ms setups for 16 kbps rate. The

160 kbps and 32 kbps rates benefited more from the increased lengths and shifts, when the overall highest recognition results were achieved for longest frame-lengths and shifts (32/16 ms) and CMVN normalization, in both cases 0.94%.

On the other hand, the 16 kbps achieved the lowest error of 0.67% for a rather non-standard 25/12.5 ms setup, while the more common 32/16 ms setup achieved the error rates of 1.35% and 1.48% for CMN and CMVN normalization. Another interesting thing to note was that the results for 16 kbps bitrate followed the same trend as the one observed for uncompressed speech, where the window shift played a major role in the overall error rate. Certain setups, 16/8 ms in particular, proved to yield unacceptable *WER* for such an easy task such as digit recognition.

Table 6.6: Results for 16 kbps data with different window lengths/shifts [ms] and normalization schemes: a)Non-Normalized; b)CMN; c)CMVN

A	*Shift*					
Length	8	10	12.5	13	15	16
16	18.84	-	-	-	-	-
25	12.79	10.5	2.42	-	-	-
30	7.54	7.13	-	-	1.21	-
32	8.48	6.86	-	8.48	-	1.88

B	*Shift*					
Length	8	10	12.5	13	15	16
16	2.83	-	-	-	-	-
25	1.48	1.21	0.81	-	-	-
30	1.75	1.21	-	-	1.08	-
32	2.15	1.35	-	0.81	-	1.35

C	*Shift*					
Length	8	10	12.5	13	15	16
16	2.56	-	-	-	-	-
25	2.56	1.08	0.67	-	-	-
30	2.83	2.42	-	-	1.21	-
32	2.29	1.48	-	0.81	-	1.48

Table 6.7 summarizes the potential gain of increasing the values of frame-length or shift for a fixed value of shift or length. As the table indicates, once a certain window shift was chosen, the overall effect of window-length played only marginal role and the systems achieved roughly the same results. On the other hand, the decrease in *WER* for fixed window-length and increasing window-shift was noticeably higher. Thus, the conclusion from these experiments was that the window overlap had much greater impact on the resulting *WER* of the system.

Table 6.7: Average decrease in *WER* for fixed length/shift and increasing shifts/lengths

Fixed Shift [ms]		*Fixed length [ms/]*		
8	10	25	30	32
0.34	0.15	1.27	1.1	0.95

The *WERRs* for all parametrizations are shown in figures below, where the trend already observed from previous experiment was observed again. The improvement decreased as the window-shift was getting longer, which could be attributed to the fact that longer window-shifts achieved considerably lower base *WER* and thus there was less room for potential improvement. The experiments also showed that the major portion of improvement was due to the CMN and that the subsequent extension to CMVN decreased the error

only slightly. In both cases, the relative error reduction reached for certain parametrization nearly 80%, while the average reduction for CMN was 57% and for CMVN 60%. The second observation is the fact that the application of normalization evened out the performance among different bitrates for such a simple task as digit recognition.

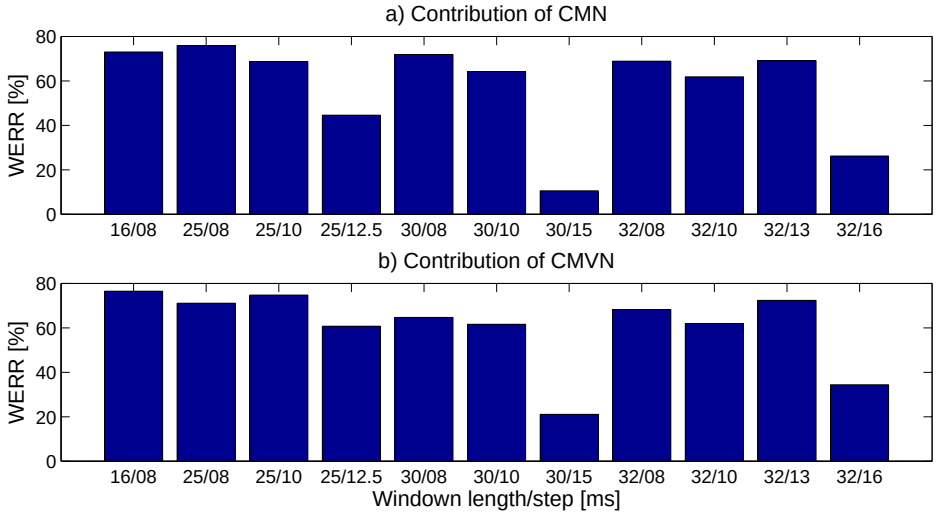

Figure 6.7: *WERR* for CMN and CMVN techniques in digit recognition

6.4.3 Initial Results for LVCSR task

As the LVCSR task was the primary application of the MP3 speech recognition, the overall best performing setup determined in previous analyses (25/10 ms with CMVN) was evaluated in a LVCSR task. The evaluation set consisted of 1 hour of speech and the decoder used a bigram LM and 340k vocabulary with 4% OOV. The obtained results are summarized in Table 6.8.

The RAW models achieved 28.56% *WER* and the absolute *WER* difference reached 6.61%, 7.84% and 13.32% for 160 kbps, 32 kbps and 16 kbps bitrates respectively. On the other hand, the previous loop-digit recognition tests showed lower *WER* for non-compressed data than for compressed, a fact that was not observed in the LVCSR test. This situation occurred most likely because the loop-digit task tested only a part of trained triphones from the whole AM, while the LVCSR set contained phonetically richer content and the AM was tested as a whole. Although non-compressed AM achieved overall worse results, the absolute *WER* difference was only about 0.5%. Given the size of the loop-digit testing dataset, the 0.5% difference meant about 4-5 correctly recognized words.

Table 6.8: Results in LVCSR task for the best parametrization setup as determined from digit recognition. The digit recognition task is repeated for comparison.

	RAW	160 kbps	32 kbps	16 kbps
Digit	1.35	1.08	1.24	0.94
LVCSR	28.56	35.17	36.4	41.88

6.4.4 Summary

The most important conclusions of these analyses can be summarized as follows.

- The commonly used frame setups of 32/16 ms and 25/10 ms achieved generally good results for higher compression rates of 160 kbps and 32 kbps. The best error rates of 0.94% was achieved for 32/16 setup for both the 160 kbps and 32 kbps rates. The lower compression rate of 16 kbps required lower segmentation setup of 25/12.5 ms, but with a proper setting achieved results even marginally better (0.67%) than higher compression rates for a simple digit recognition.
- The contribution of normalization schemes was crucial for system refinement and yielded up to 80% relative improvement. Although very slightly, the average contribution of CMVN outperformed the CMN and should therefore be used regardless of segmentation setup.
- The performance of AMs in the LVCSR task showed clear advantage for uncompressed speech with 28.56% *WER*, while the 160 kbps and 32 kbps rates performed at 35.17% and 36.4% WER respectively. The worst overall results of 41.88% were achieved for 16 kbps.

6.5 Basic Front-End Optimization for LVCSR

The analyses and results of particular experiments presented in this section follow on the findings from the previous section and extend on them. It investigates the contribution of methods working at the level of front-end processing (CMVN) and acoustic modelling (MAP and CMLLR adaptations) which are generally used for distorted speech recognition. It also presents a more thorough analysis on frame level optimization for a LVCSR task using the same protocol as in the previous section. The compression rates were chosen according to the previously presented results with the intention to map the performance more closely for lower bitrates (160 kbps, 32 kbps, 24 kbps and 16 kbps) where the error rate started to rise rapidly. The system was constructed using the same setup as in the previous section. The recognition task consisted of 1 hour set of signals containing only full sentences. Both the supervised and unsupervised speaker adaptation was performed using the CMLLR and MAP adaptation techniques. The MAP adaptation constant τ and the number of regression classes for CMLLR were set according to the previous empirical analyses. The decoder used the bigram LM with 340k vocabulary. The decoding was done with *HDecode* decoder.

6.5.1 Results for Matched conditions

The experiments in this section were performed for matched conditions when the features were extracted with the basic setup and the AMs were speaker independent and in

match with the testing conditions. The results are summarized in Table 6.9. The analysis of the feature extraction setup for compressed speech showed that the optimal values of the frame-shift were around the values of 10/12.5 ms with a slight variation and that the longer shifts were suboptimal. These results were marginally different from the results for the uncompressed speech were the statistically important differences were achieved for slightly longer shifts of 12.5/13 ms. The optimal values for frame lengths were found to be around 30/32 ms for both the compressed and uncompressed speech. The frequently utilized setup of 25/10 ms proved to work well for all bitrates, while the other standard setup of 32/16 ms showed the worst results. The absolute difference between 160 kbps and 16 kbps AMs was relatively small, at about 7%.

Table 6.9: Results for matched training for different window lengths/shifts [ms]

		Shift [ms]				
	Length [ms]	10	12.5	13	15	16
No comp. (SI)	*25*	35.17	31.37	-	-	-
	30	39.2	28.87	29.43	32.47	-
	32	39.47	**28.02**	29.39	-	36.21
160 kbps	*25*	28.81	29.58	-	-	-
	30	29.95	**29.12**	29.83	32.03	-
	32	40.28	29.72	29.56	-	34.86
32 kbps	*25*	32.47	32.36	-	-	-
	30	**29.08**	30.28	30.89	33.69	-
	32	29.14	30.08	32.24	-	36.54
24 kbps	*25*	37.08	32.38	-	-	-
	30	36.98	34.57	31.95	35.83	-
	32	40.62	**31.47**	32.22	-	36.77
16 kbps	*25*	36.58	36.4	-	-	-
	30	**35.11**	35.69	37.1	39.26	-
	32	35.17	39.33	37.21	-	41.88

Supervised Adaptation

The deployment of ASR system for MP3 recognition is often intended for pre-recorded speech, where the identity of the speaker in known and the AM adaptation can be performed in the supervised fashion. The goal of the following analysis was to investigate in limit case of *WER* the ASR system can potentially achieve. The CMLLR adaptation was performed in a two-step fashion. The global transformation acted as the parent for the class specific one. The number of regression classes for supervised CMLLR adaptation was set up to 12 and the MAP adaptation constant τ was set to 10. The results for CMLLR adaptation are summarized in Table 6.10 and were expected to follow the trend already observed for the SI system, where certain segmentation setups would achieve considerably better results then the other. The average *WERR* for the CMLLR was slightly over 35%. The absolute difference between 160 kbps and 16 kbps rate was lowered down to 3%. The results for MAP adaptation, summarized in Table 6.11, improved the recognition even further and lowered the absolute error rate just under 10%, which meant about 65% *WERR*. The difference between particular bitrates was generally around 1%. These results showed that the application of an AM adaptation has evened out the differences

between particular segmentation setups and bitrate speeds. This results indicated that extracted feature still contained enough information about the content and that the observed ASR performance drop occurred mostly due to the training-evaluation mismatch. However, it is still important to remember that the AM adaptation was employed in a supervised fashion.

Table 6.10: Results for supervised CMLLR with matched training and different window lengths/shifts [ms]

		Shift [ms]				
	Length [ms]	10	12.5	13	15	16
No comp.	*25*	20.12	19.33	-	-	-
	30	19.04	**18.64**	19	21.06	-
	32	19.06	18.87	19.7	-	21.91
160 kbps	*25*	19.58	20.37	-	-	-
	30	18.98	19.21	19.91	22.37	-
	32	**18.71**	19.58	19.95	-	24.07
32 kbps	*25*	18.44	19.73	-	-	-
	30	**18.73**	19.04	19.75	21.04	-
	32	21.04	18.89	18.6	-	22.99
24 kbps	*25*	20.52	21.16	-	-	-
	30	**19.43**	21.04	20.81	23.26	-
	32	21.37	19.66	20.62	-	24.53
16 kbps	*25*	22.91	23.09	-	-	-
	30	**20.66**	22.07	22.64	25.42	-
	32	22.47	23.32	23.03	-	27.69

Table 6.11: Results for supervised MAP, matched training and different window lengths/shifts [ms]

		Shift [ms]				
	Length [ms]	10	12.5	13	15	16
No comp.	*25*	10.85	9.89	-	-	-
	30	11.39	9.98	9.77	10.33	-
	32	11	**9.3**	9.87	-	11.58
160 kbps	*25*	10.54	10.68	-	-	-
	30	**9.23**	9.69	9.44	10.6	-
	32	9.46	9.44	9.87	-	12.91
32 kbps	*25*	9.69	9.96	-	-	-
	30	10.21	9.54	9.79	11.14	-
	32	13.06	**9.08**	9.21	-	12.7
24 kbps	*25*	10.23	9.58	-	-	-
	30	9.81	12.51	9.4	13.43	-
	32	11.21	**9.19**	9.42	-	11.68
16 kbps	*25*	11.16	10.56	-	-	-
	30	10.62	**9.42**	10.85	11.64	-
	32	10.1	11.2	10.41	-	12.37

Unsupervised Adaptation

The analysis in this section was focused on using the unsupervised adaptation from recognized transcription and the results were compared against the supervised adaptation from the previous section. The adaptation was performed only with the CMLLR technique. The optimization of the number of regression classes did not have such a significant impact on the error rate as it did for supervised CMLLR, although a few differences have been observed. The application of the global transformation worked well in general, but the optimal number ranged from 2 to 4. As a results, the number of regression classes was lowered to 4 to compensate for the transcription errors. The application of MAP was found to actually degrade the performance instead of improving it. This problem occurred mostly due to the number of errors in prior transcription used for subsequent adaptation. The results are summarized in Table 6.12. The overall difference between the supervised and unsupervised adaptation for the selected segmentation was always above 5% absolutely. The adaptation achieved progressively worse results with a decreasing bitrate, which could be partially attributed to the increasing input error in the obtained transcription.

Table 6.12: Results for unsupervised CMLLR adaptation, matched training and different window lengths/shifts [ms]

		Shift [ms]				
	Length [ms]	10	12.5	13	15	16
160 kbps	*25*	25.59	26.27	-	-	-
	30	26	**25.17**	25.77	28.1	-
	32	35.96	25.73	25.55	-	30.99
32 kbps	*25*	27.33	27.83	-	-	-
	30	25.5	26.61	27.48	29.54	-
	32	**25.48**	26.86	27.33	-	32.88
24 kbps	*25*	30.53	27.94	-	-	-
	30	30.45	31.34	28.52	31.66	-
	32	35.77	**28.14**	28.52	-	33.24
16 kbps	*25*	32.51	32.38	-	-	-
	30	36.4	32.11	33.05	35.59	-
	32	**31.26**	34.59	33.69	-	37.44

6.5.2 Results for Mismatched conditions

The previous sections have established that building an AM for each specific bitrate is both a time and resources consuming process, that provides only limited usability in a real-life situations. As a result, we would often want to have a generic AM that would perform equally well on RAW and compressed speech regardless of used bitrate. The results achieved from the previous experiments suggested that the AMs would mutually interchangeable if a proper AM adaptation was used. The focus of the following analysis was to evaluate the performance of an uncompressed AM on the compressed data using the best achieved setups. The previous analysis showed that the proper frame setup can vary for compressed and uncompressed speech. Therefore, the first step was to determine

which uncompressed AM would perform the best if used on compressed speech. From all the possible options, four were selected:

- 25/10 ms - most frequent in ASR, good results for RAW and compressed speech,
- 30/10 ms - most consistent for compressed speech but very poor for RAW speech,
- 32/12.5 ms - best results for uncompressed and well for compressed speech,
- 32/16ms - frequently used setup in ASR although the results were suboptimal.

Table 6.13: Results with mismatched AMs for selected segmentaion setups

	160 kbps	32 kbps	24 kbps	16 kbps
25/10 [ms]	30.37	32.76	34.55	41.36
30/10 [ms]	38.08	41.07	42.51	45.62
32/12.5 [ms]	27.89	30.18	32.26	39.91
32/16 [ms]	27.35	30.38	32.45	38.86

Table 6.14: Results for supervised CMLLR and MAP in mismatched conditions

	CMLLR				MAP			
	160 kbps	32 kbps	24 kbps	16 kbps	160 kbps	32 kbps	24 kbps	16 kbps
25/10	20.45	21.21	21.69	21.91	10.37	10.55	10.38	10.89
30/10	21.32	21.67	22.01	21.2	10.99	10.21	11.29	10.72
32/12.5	19.98	20.71	20.19	20.65	9.21	9.71	9.7	9.79
32/16	19.77	20.45	20.22	20.7	9.56	9.73	9.7	9.99

The results for the baseline system are summarized in Table 6.13. Interestingly, the 32/16 ms setup performed the best in this mismatched scenario while the 32/12.5 ms setup performed as the second best. The worst results were achieved for the 30/10 ms. Also, the results repeated the trends of of a rapid rise in *WER* for the bitrates higher than 24 kbps. The absolute difference in *WER* between the RAW and 16 kbps compressed speech reached 12.49%. These results confirmed that without any AM adaptation, the uncompressed and AM could be used on the compressed data only in a limited number of cases and always resulted in significantly worse performance.

Table 6.14 summarizes the contribution of supervised MAP and CMLLR adaptation technique in the mismatched training. The application of AM adaptation techniques has improved the recognition significantly and the obtained results obtained comparable to the bitrate specific AMs. The recognition results after the MAP adaptation were even marginally better and proved that the AM trained on the uncompressed data could be used for the compressed signals in case of subsequent supervised adaptation, which is the case for any current off-line dictation systems. The comparison of different segmentation setups demonstrated that longer window lengths were preferable while the results for shift were not as clear. The best overall results with MAP adaptation were achieved for the 32/12.5 ms segmentation and the 32/16 ms proved to be only slightly worse. However, the situation was actually reversed for the CMLLR adaptation. Based on these findings I have chosen to use the 32/16 ms segmentation for all my further analyses and the reasoning is as follows. The following analyses make use fMLLR adaptation for the SAT and speaker adaptation during the decoding step. Also, the 32/16 ms segmentation is more standard for ASR systems.

Additional Dithering

The cited works on the MP3 recognition showed that the PLP features demonstrated a much slower decline in recognition rate as the bitrate decreased than the MFCCs. Also, the addition of a small amount of noise has been found to greatly improve the performance for MFCC features and the theoretical explanation of this phenomenon is given in [98]. The uniform dithering technique was applied at the feature extraction level and could be considered similar to other normalization techniques. However, it is important to realize that I always used a uniform dithering during feature extraction with R set to 1 in order to avoid zeros in spectrum. The purpose of the following analysis was to extend these results for the employed PLP features as well as to find the optimal value of R.

The results for additionally dithered features are summarized in Table 6.15. The absolute *WER* difference between 160 kbps and 16 kbps was about 11%. The application of additional dithering lowered the error rates by about 3% absolutely, although the optimal R values were not distributed as expected. My initial assumption was that the optimal dithering value should have increased as the bitrate decreased. This hypothesis was proved to be only partially true as the optimal R for 24 kbps was found to be 2 and not in the range from 16 to 32 as the other results would indicate. However, the absolute difference in *WER* for the best value $R = 2$ and $R = 16$ was only 1.5%. It could be concluded that uniform dithering was able to bring addition improvement for PLP feature and basic AMs.

Table 6.15: Results with mismatched AMs and increasing dithering value R

	Dithering value R					
	2	4	8	16	32	64
160 kbps	28.68	28.71	**23.62**	29.00	38.87	43.28
32 kbps	30.64	30.70	30.80	**26.59**	31.03	43.05
24 kbps	**29.10**	29.21	29.63	30.62	36.58	41.18
16 kbps	36.53	37.22	37.40	41.57	**35.20**	48.56

6.5.3 Summary

Table 6.16 summarizes the most important results with mismatched conditions while the conclusions of this part can be summarized as follows.

Table 6.16: Summary of results for basic AM in LVCSR with mismatched AM

Rate	160 kbps	32 kbps	24 kbps	16 kbps
Base-line	26.35	29.38	31.45	38.86
Dithering	23.62	26.59	29.10	35.20
superv. CMLLR	19.77	20.45	20.22	20.7
superv. MAP	9.56	9.73	9.7	9.99

- The optimal values for segmentation values for matched condition included longer frame-lengths and shorter frame-shifts. The proper values were found to be around 30/12.5 ms.

- PLP features were naturally more robust against the degradation and the quality of the acoustic models trained on compressed data deteriorated slowly with decreasing bitrate for them. The difference between 160 kbps and 16 kbps AMs was only 7%. The application of CMVN proved to improve the performance significantly, especially for shorter windows and shifts and lower compression rates. This finding could be attributed to the fact that CMVN presented a simple solution to the problem of increased dynamic and acceleration coefficients.

- The supervised adaptation proved to work very well. The results for unsupervised adaptation were not as conclusive. The application of MAP proved to be impossible due to the very high number of errors in the initial transcription. The CMLLR proved to be much more robust but its setup had to be changed slightly as well when the number of regression classes was lowered down to 4 from the initial 12 as was the case for the supervised CMLLR.

- The optimal segmentation values for mismatched condition included longer frame-lengths and longer frame-shifts. The proper values were found to be around 30/16 ms. All further analyses made use of this setup as they were mainly concerned with mismatched conditions and made use of fMLLR adaptation.

- The application of AM adaptation was found to be crucial in the case of mismatched training-evaluation. Although the bitrate specific AMs performed better than the uncompressed AM on the compressed speech, the application of an AM adaptation lowered the *WERs* to about 20% for CMLLR and 10% for MAP, regardless of the bitrate. These results demonstrated that uncompressed AM could be used for compressed speech recognition in the case of supervised adaptation and with a proper front-end processing methods.

- The application of uniform dithering was found to be beneficial even for PLP features as it resulted in about 3% absolute error rate reduction.

6.6 Advanced Front-end and AM optimization

The analysis presented in this section continued with the set trend from the previous sections on evaluating the whole AM creation chain in terms of its robustness against MP3 compression in mismatched conditions. This time, however, the goal was to evaluate the performance with current state-of-the-art acoustic modelling techniques and one specific front-end compensation method for a GMM-HMM based system. Specifically, the section concentrated on speaker adaptive training, AM adaptation, discriminative training and uniform dithering as prominent means of compensating in the task of phoneme and LVCSR recognition.

The reason to examine both the phoneme and LVCSR tasks was as follows. The phoneme recognition task allowed a much greater insight into the errors made by the decoder and hinted at the possible shortcomings of the used setup. It also enabled to

examine the assumptions made earlier in the thesis about the effects of particular distortions (bandlimiting and spectral valleys) on particular phonetic units. In short, these results were used mainly to get further knowledge of MP3 compression and its effects on ASR. On the other hand, MP3 speech recognition was generally intended for applications such as off-line transcription of recorded speech or indexing of audio archives and thus the results from the LVCSR task were necessary for reference.

The signals came from the Czech SPEECON and TEMIC databases and the compression rates were selected with the intention of evaluating the performance of the system for bitrates of 128 kbps, 64 kbps, 32 kbps, 28 kbps, 24 kbps, 20 kbps, 16 kbps and 12 kbps. The PLP and MFCC features were computed using the CtuCopy with the 32/16 ms segmentation. The CMN technique was applied in a speaker specific fashion and on static features only. The frame-splicing step used 5 neighboring frames. In the first stage of the experiments, the signals were dithered with uniformly distributed random values from the $< -1, 1 >$ range. The effect of additional dithering for the test subset was studied in the later stages, when the dithering value $< -R, R >$ was gradually increased until the error rate stopped dropping.

The AMs were trained on uncompressed speech using with an overall length of 72 hours. The phoneme test subset contained 45 speakers and 8.5 hours of speech of varying content and the recognition was performed using a bigram phoneme model. The LVCSR task was evaluated on 2 hours of speech and a bigram LM with 340k vocabulary. The results were evaluated by PER and PERR criteria. In addition, the recognized transcription was remapped into three phonetic classes: voiced consonants, unvoiced consonants and vowels, and the phone error rate contribution ($PERC_{cl}$) for a particular phonetic class was computed.

6.6.1 Results for Phoneme Recognition

The initial study of the behaviour of PLP and MFCC features for MP3 speech recognition was performed with all previously-discussed AM refinement techniques, but without any non-standard modifications to the feature extraction process. This analysis served as the benchmark for the subsequent modification in the form of additional dithering and its potential contribution.

Table 6.17 presents results for the PLP-based system. Significant differences in absolute *PER* were observed between compressed and uncompressed data for initial baseline AMs but implementation of each subsequent modelling technique decreased the *PER*. The studied bitrates were selected to have a linear trend, but the achieved results identified the 24 kbps bitrate as a breakpoint after which the error started to rise exponentially. This conclusion held true for all levels of AM development.

Another point of interest was the reduction of error as a function of the employed modelling technique. The fMLLR adaptation achieved the highest *PERR* for compressed speech in general, and its gain rose with decreasing bitrate. On the other hand, the gain

Table 6.17: *PER* for PLP and progressively refined AM

	Baseline	LDA	SAT	SGMM	bMMI	MPE
RAW	17.9	16.1	12.3	10.2	7.2	7.2
128 kbps	18.8	17.1	13.5	10.8	7.9	7.8
64 kbps	18.9	17.1	13.6	11.0	8.2	8
32 kbps	19.3	17.3	13.2	11.0	8.2	8.1
28 kbps	19.6	17.7	13.5	11.3	8.6	8.2
24 kbps	20.7	18.7	14.1	11.9	9.3	8.6
20 kbps	23.9	21.1	15.5	12.9	10.5	10.1
16 kbps	36.2	30.4	18.6	15.5	13.3	13.1
12 kbps	62.5	52.9	26.8	22.2	20.2	19.8

of discriminative training was the highest for RAW data and decreased with decreasing bitrate. This finding was consistent with the theoretical premise that discriminative training fits the AM on the training set, but not necessarily on the testing set. In the case of my experiment, the employed DTs optimized the AM for uncompressed signals and as the bitrate decreased, so did the match between the training and evaluation signals. It should be noted, however, that the overall *PERRs* computed between baseline and final bMMI models were still in the $\langle 59\%,67\% \rangle$ range. Another interesting thing was that the MPE trained achieved marginally better results than bMMI models.

Table 6.18: *PER* for MFCC and progressively refined AM

	Baseline	LDA	SAT	SGMM	bMMI	MPE
RAW	17.8	15.9	12.3	10.3	7.3	7.2
128 kbps	18.9	17.2	13.7	10.8	8.1	8.2
64 kbps	19.3	17.5	13.7	11	8.2	8.2
32 kbps	20.9	17.9	13.6	11.3	8.7	8.3
28 kbps	24.6	19.5	14.3	11.9	9.4	9.2
24 kbps	33.9	25.9	16.4	13.5	11.4	11.1
20 kbps	47.1	31.9	19.0	15.6	13.6	13.3
16 kbps	62.2	49.4	25.8	20.4	18.7	18.6
12 kbps	73.0	68.7	46.0	32.6	30.2	29.7

The same set of experiments for an MFCC-based system is summarized in Table 6.18. The system behaved similarly and displayed the same trends as far as the contribution of specific acoustic modelling techniques went. The *PER* displayed a tendency to rise rapidly after passing the 24 kbps breakpoint and AM adaptation proved to be crucial as it contributed the most to the overall *PERR*. The major difference was the overall increase of error rate for compressed signals, which was much higher for MFCCs than for PLPs. This finding lead to the conclusion that the MFCC features were not suitable for low bitrate MP3 speech recognition.

A more detailed study of the nature of error confirmed the theoretical assumptions about the compression distortions and their effect on particular phonemes. Figure 6.8 documents a decrease in *PERC* for voiced phonemes at the expense of unvoiced phones.

While the reference *PERC* distribution for 128 kbps was dominated by the vowels, the *PERC* for unvoiced consonants steadily increased up to 34.2% for 12 kbps. In fact, all three studied classes contributed to the overall *PER* approximately equally for the lower bitrates. This observation was particularly interesting if we looked at the absolute number of phonemes for each phonetic class. The whole evaluation set contained 116 501 phonemes altogether, 47 389 vowels (22.5% of the whole), 42 939 voiced consonants (40.6%) and 26 173 unvoiced consonants (36.9%). In fact, these numbers very closely matched the distribution of *PERC* for 128 kbps bitrate. This observation proved that a "normal" ASR system produces equally distributed phonetic errors across the studied classes. Later experiments with partial contribution of each distortion showed that this negative effect occurred due to the combined presence of both the bandwidth limitation and spectral valleys.

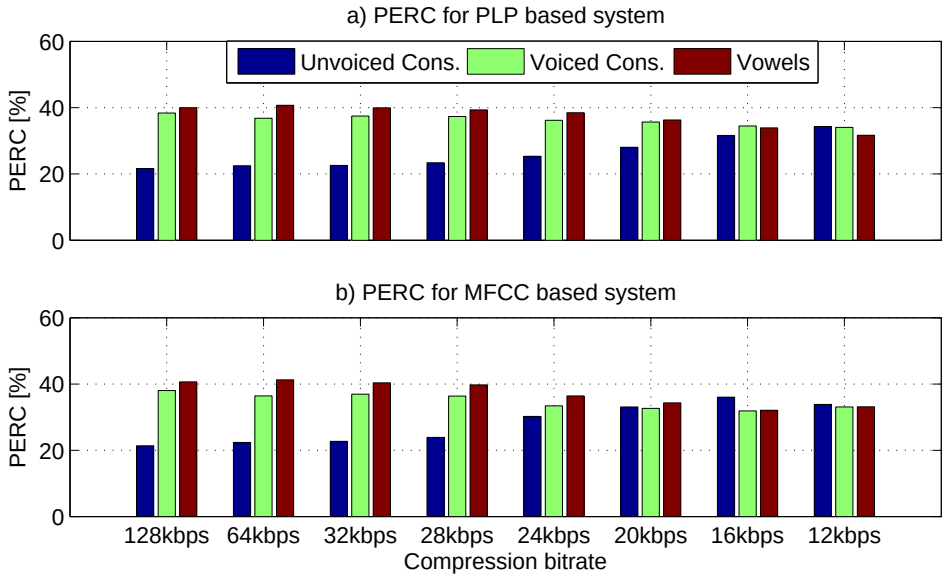

Figure 6.8: PERC for PLP and MFCC features. The realative contribution to the total *PER* is mapped for three phonetic groups: Unvoiced Consonants, Voiced Consonants and Vowels.

Additional Dithering

All the previous analyses in this section used features without any further compensation. Since the previous analyses showed the detrimental effect on higher frequency bands, it was important to investigate the additional dithering method. The dithering value R was gradually increased by a factor of 2. Table 6.19 and Table 6.20 present the best *PER* together with the optimal R.

Additional dithering for PLP features, summarized in Table 6.19, yielded consistent improvement for the lowest 12 kbps rate and some improvement for the 16 kbps rate. Its application was particularly useful for baseline and LDA models, but the reduction for more advanced AMs was only marginal and higher bitrates were mainly unaffected by the method. In cases when the dithering value was too high, the additional noise degraded

Table 6.19: *PER* for dithered PLP with diff. dithering value R

	Baseline	LDA	SAT	SGMM	bMMI	MPE
128 kbps	2/18.7	2/17.0	4/13.2	4/10.7	4/7.9	2/7.9
64 kbps	2/18.8	2/17.1	2/13.2	4/10.8	4/8.0	2/8.0
32 kbps	4/19.1	2/17.3	2/13.2	2/11.0	2/8.2	4/8.0
28 kbps	4/19.4	4/17.7	4/13.5	4/11.2	4/8.6	2/8.4
24 kbps	2/20.8	2/18.8	4/14.2	2/11.8	2/9.3	4/9.0
20 kbps	4/23.5	2/21.0	4/15.6	2/12.9	2/10.5	4/10.1
16 kbps	16/32.0	8/27.9	8/18.5	4/15.4	2/13.4	4/13.4
12 kbps	32/44.7	32/39.4	16/24.9	8/21.2	4/19.8	6/19.4

Table 6.20: *PER* for dithered MFCC with diff. dithering value R

	Baseline	LDA	SAT	SGMM	bMMI	MPE
128 kbps	2/18.9	2/17.1	4/13.1	4/10.7	4/8.0	2/8.1
64 kbps	2/18.8	2/17.3	4/13.2	4/10.9	4/8.2	2/8.2
32 kbps	8/20.1	4/17.8	4/13.4	4/11.1	2/8.6	4/8.2
28 kbps	8/21.6	4/18.6	8/13.9	8/11.5	8/9.1	6/8.7
24 kbps	16/30.1	2/26.2	8/15.8	8/13.1	8/10.9	8/10.7
20 kbps	16/34.7	8/30.0	8/17.7	8/14.8	8/12.8	8/12.6
16 kbps	32/41.2	32/35.9	16/21.3	8/17.9	8/16.4	8/16.1
12 kbps	64/49.9	64/45.5	16/29.0	16/24.2	16/23.0	12/22.4

the features further, which resulted in worse *PER* than for the undithered system. The process of estimating the R value included several iterations of feature extraction and decoding and thus consumed a lot of time and resources. When all of these factors were taken into consideration, I came to the conclusion that the usage of additional dithering couldn't be advised for PLP features.

The next main point of interest was to investigate whether the dithered MFCCs can match the PLPs. These experiments showed more convincing results as a positive *PERR* was obtained for all bitrates and levels of AM refinement, as summarized in Table 6.20. The generally observed trend was that the lower bitrates gained more from the additional dithering than the higher bitrates. It should be noted however, that MFCCs still did not manage to outperform the PLPs. The error rates were somewhere between the original MFCCs and PLPs.

Since it was confirmed that additional dithering can improve the recognition with MFCCs, the last analysis was focused on phonetic composition of the error. The initial hypothesis was that the addition of relatively weak noise would compensate the spectra of unvoiced phones, but the results showed that the error reduction was spread evenly among phonetic classes or slightly towards the voiced phones. Figure 6.9 compares *PERC* for dithered and undithered MFCCs at 16 kbps and 12 kbps as these were the only bitrates which displayed a statistically relevant improvement. The values for the 12 kbps test sets with and without dithering were basically the same, which indicated a uniform contribution. The error for dithered 12 kbps set was dominated by the unvoiced phonemes, but

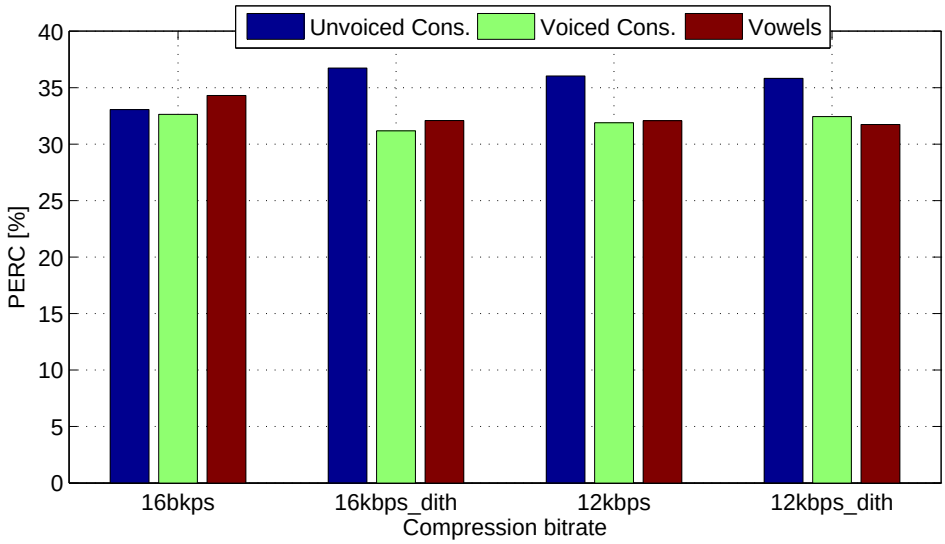

Figure 6.9: PERC for MFCC and dithered MFCC features. The contribution is evaluated for three phonetic groups: Unvoiced Consonants, Voiced Consonants and Vowels.

the *PERCs* for original MFCCs were dominated by the vowels, which means that voiced phones gained more than the unvoiced phonemes. Based on these results, it could be said that the key principle of this method lied more in enhancing the features suitability for statistical modelling and less in actual reconstruction of their spectral characteristics.

6.6.2 Results for LVCSR

The primary application of an MP3 recognizer is for the off-line transcription of compressed speech and thus the following analyses will be focused on the LVCSR task. Tables 6.21 and Table 6.23 compare the LVCSR results for PLP/MFCC systems and Tables6.22 and 6.24 present to results for dithered MFCC and PLP based systems for the LVCSR task. The observed trends of error rates corresponded with the conclusions observed in the phoneme experiments. The error started to rise exponentially after passing the 24 kbps threshold, and the AM adaptation was the main source of error reduction for lower bitrates. The contribution of DT had a decreasing tendency. The advanced acoustic modelling techniques displayed a trend of increasing relative gains as the bitrates decreased, a result which was in contrast to phoneme recognition. This development occurred most likely due to the usage of word level LM in combination with progressively better AMs. The PLP features achieved better results than MFCC on average, and marginally better than the dithered MFCC (dMFCC) features.

Another interesting thing to look at was the comparison of two DT methods: bMMI and MPE. Figure 6.10 illustratively summarizes error rates for bMMI and MPE trained AMs in both recognition tasks. The figures demonstrated the advantage of MPE training criteria, which was especially pronounced at low bitrates. As a result, all further experiments were conducted using this training method. Another interesting thing to note was that

Table 6.21: *WER* for PLP system for progressively refined AM

	Baseline	LDA	SAT	SGMM	bMMI	MPE
RAW	23.74	21.8	18.19	16.14	14.25	14.01
128 kbps	24.50	22.55	19.45	16.20	14.43	14.18
64 kbps	24.53	22.52	19.45	16.50	14.50	14.33
32 kbps	24.50	22.45	19.40	16.87	14.55	14.45
28 kbps	25.57	23.67	19.59	17.21	14.54	14.64
24 kbps	25.98	23.75	19.84	17.67	15.21	15.02
20 kbps	28.19	25.01	20.67	18.08	16.15	16.02
16 kbps	38.79	31.76	23.56	20.11	18.57	18.15
12 kbps	68.57	47.20	33.19	28.87	25.23	25.20

Table 6.22: *WER* for dithered PLP system for progressively refined AM

	Baseline	LDA	SAT	SGMM	bMMI	MPE
128 kbps	24.84	23.56	18.52	14.88	13.99	14.24
64k kbps	24.60	23.28	18.22	14.87	14.00	14.27
32k kbps	24.86	23.12	18.48	14.94	14.24	14.5
28k kbps	25.38	23.32	18.85	15.36	14.48	14.64
24k kbps	26.17	24.82	18.91	15.98	14.88	15.07
20k kbps	28.45	26.31	20.10	16.66	16.24	15.88
16k kbps	37.67	32.51	22.91	18.57	18.19	18.27
12k kbps	61.38	54.93	31.66	26.66	24.57	24.04

Table 6.23: *WER* for MFCC system for progressively refined AM

	Baseline	LDA	SAT	SGMM	bMMI	MPE
RAW	23.72	21.7	18.44	16	14.22	14.28
128 kbps	25.07	22.45	19.06	16.22	14.72	14.56
64 kbps	25.17	22.51	19.09	16.41	14.79	14.68
32 kbps	25.13	22.58	19.11	16.52	14.92	14.77
28 kbps	26.67	23.99	19.92	16.95	15.12	15.11
24 kbps	31.75	24.43	20.7	17.89	15.82	15.54
20 kbps	38.46	29.67	22.51	19.79	17.57	17.21
16 kbps	62.45	44.71	28.11	24.46	21.48	21.09
12 kbps	91.43	71.48	44.82	36.76	31.54	31.45

the absolute *WER* difference between MFCCs and PLPs increased with the decreasing bitrate. In another words, MPE proved to be a more robust DT method than bMMI. This finding corresponded with cited works which also demonstrated the superiority of MPE training over bMMI training for distorted speech. And finally, Figure 6.11 illustratively summarizes the contribution of using *Uniform Dithering* for the DT models. It can be noticed that statistically important improvements were reached only for MFCC features and lower bitrates.

Table 6.24: *WER* for dithered MFCC system for progressively refined AM

	Baseline	LDA	SAT	SGMM	bMMI	MPE
128 kbps	24.85	22.55	18.75	16.32	14.25	14.18
64 kbps	26.03	22.67	18.89	16.64	14.38	14.31
32 kbps	25.05	22.85	19.01	17	14.78	14.69
28 kbps	26.01	23.89	19.61	17.64	15.06	15.18
24 kbps	31.77	25.01	20.32	18.23	15.5	15.34
20 kbps	36.69	28.11	21.71	19.7	16.84	16.5
16 kbps	48.83	34.56	25.17	22.42	19.47	19.13
12 kbps	70.03	41.4	34.75	30.1	26.41	26.5

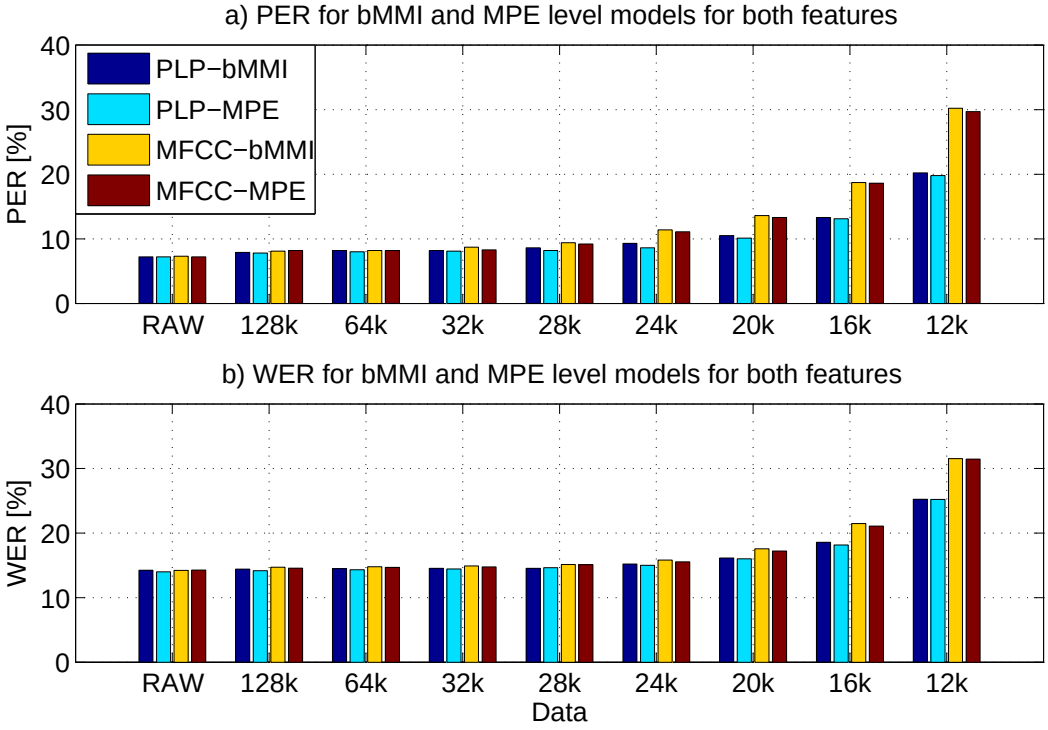

Figure 6.10: Error rates for the final DT models

6.6.3 Summary

The most important findings from these series of analyses can be summarized as follows.

- The evaluation runs documented that the usage of PLP features and application of AM adaptation and DT could significantly reduce *WER* of the system. The bMMI trained AMs performed at 14.24% on the reference test set, but the *WER* increased to 18.57% for 16 kbps and 25.23% for 12 kbps rates. In comparison, the MFCC system performed at 14.22%, 21.48% and 31.54% *WER*.
- MPE criteria function yielded slightly better results than bMMI which made it a preferred DT method for all further experiments. Also, it was concluded that MPE

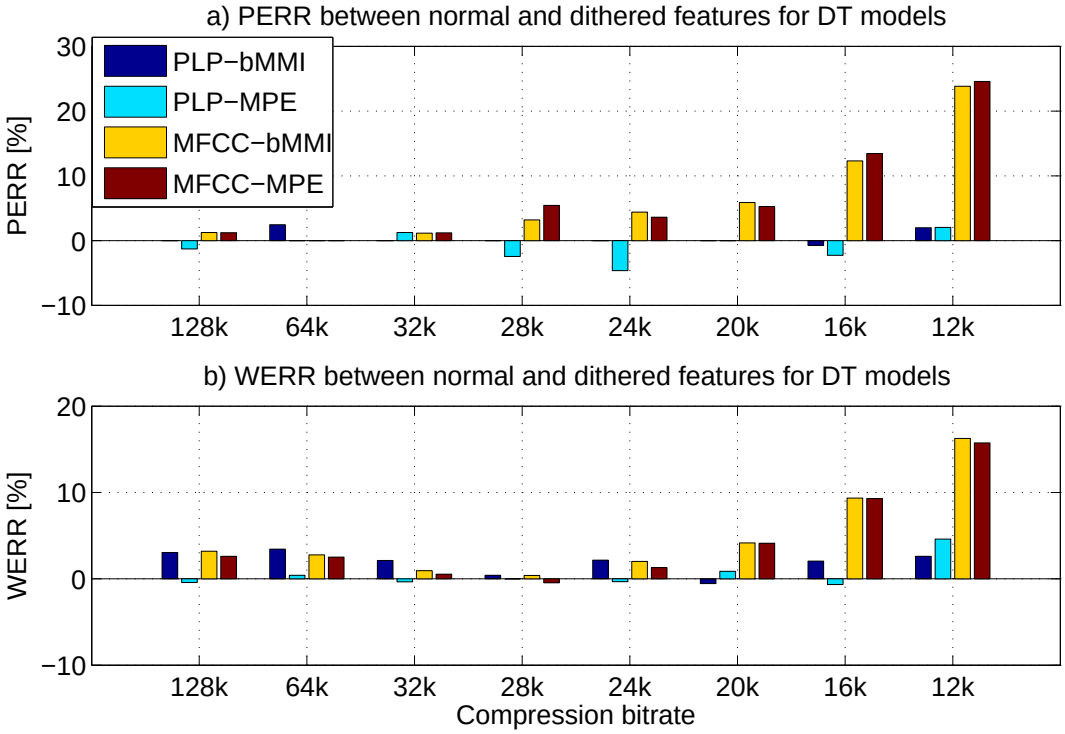

Figure 6.11: The absolute contribution of dithered features for the final DT models

was more robust against MP3 distortions.

- Adapting the AMs to the specific speaker and bitrate yielded the highest mean improvement out of all the analyzed modelling techniques, and proved to be essential for recognition of compressed speech. The gain of discriminative training diminished with decreasing bitrate.

- The phoneme-level recognition confirmed the theoretical hypothesis that the MP3 compression affected the unvoiced phonemes more significantly than the voiced phonemes. The contribution of the unvoiced phonemes to the total phone error rose from 21.6% for the reference test set to 34.2% for the 12 kbps set.

- While the observed results justified the usage of additional dithering for the MFCC features, the error rates for dithered MFCCs were still slightly higher than those for PLPs. However, the main problem of this approach was the need to manually tune the dithering value to achieve the best results. The results of detailed phoneme accuracy showed that the technique was not able to compensate the introduced distortions and that the overall distribution of PER remained the same as for undithered MFCCs.

- It can be assumed that the advantage of PLPs originated from their design, which emulates the behavior of the human hearing system much more closely. The Bark filter bank attenuates the higher frequencies, and the cube root transforms the intensity into perceived loudness. This particular knowledge is similarly, although in much greater detail, exploited in the psychoacoustic model of the MP3 encoder.

CHAPTER 7

SPECTRALLY SELECTIVE DITHERING

This chapter introduces the proposed compensation technique called **Spectrally Selective Dithering** (SSD). The algorithm works at the front-end processing level before any subsequent feature extraction and was designed to selectively compensate the effects of spectral valleys. It was based on the principle of detecting the corrupted bands in the frequency domain and adding a controlled amount of noise. Its contribution was evaluated in both GMM-HMM and DNN-HMM systems for the Czech. The algorithm was also evaluated for German and English languages to demonstrate that the discussed problems and proposed solutions were applicable to other foreign languages. The results demonstrated that SSD could bring further improvements over the analysed state-of-the-art acoustic modelling setup from the previous chapter. The following analysis studied the limits of middle and far distance microphone recognition for MP3 speech. The chapter is concluded with the comparison of the proposed technique with the spectral bandwidth replication algorithm designed for advanced audio coding, which is the successor of MP3.

7.1 Modelling of MP3 Distortions

Previous analyses have determined that lossy compression influences unvoiced phonemes more strongly than voiced ones. In order to determine which signal degradation, bandwidth limitation or spectral valleys is more detrimental to the overall performance, the initial experiments were focused on estimating their partial contributions separately. The uncompressed signals were filtered by a FIR low-pass filter at corresponding cut-off frequencies, which were set for each bitrate according to values reported by LAME. The coefficients for linear-phase LP FIR filters were estimated using the window method of a sufficiently high order (p=50) to ensure a steep attenuation above the cut-off frequency.

On the other hand, the effects of spectral valleys was much harder to simulate without introducing other compression artifacts at the same time. Another thing to consider was whether the spectral valleys should reach into the higher frequency bands or not. The approach used in this experiment allowed spectral valleys to distort only the lower parts of the bandwidths, since I assumed that the upper parts would be wiped out by the LP filter anyway. As a result, the signals were preprocessed as follows. The signals were at first compressed/decompressed and then the upper bands were artificially added by coping the higher bands from the uncompressed signals. This approach approximated the situation when only the lower bands were compressed, while the upper bands retained all their original information. The experiments were performed for Czech language only, using both PLPs and MFCCs and GMM-HMM architecture using the common ASR framework.

The described approach allowed me to quantify each degradations as if it affected only the selected spectral parts. It should be noted however, that other compression artifacts (pre-echo, birdie, etc.) might have also degraded the lower bands and thus affected the results obtained for spectral valleys. Table 7.1 summarizes the obtained results in the LVCSR task using the best performing system from previous chapter and provides an overview of the used cut-off frequencies. It is important to realize that multiple bitarates share the same f_{cutt} and thus the table contains less values for LP filtering. The obtained results demonstrate that bandlimiting had only marginal effect on ASR as the absolute *WER* difference between the 128 kbps and 12 kbps was within 1% for both features. Second, it also shows that spectral valleys degraded the speech more significantly on average, but their contribution to the overall degradation was also only marginal as well. The sole exception was 12 kbps bitrate and MFCC features. The experimental protocol for the following analysis was changed a little in order to validate the generally accepted conclusions about the perceived quality of compressed signals.

Table 7.1: Partial contributions of LP filtering and spectral valleys in a LVCSR task, *WER*

	BitRate [bps]	128k	32k	28k	24k	20k	16k	12k
	f_{cut}	7200 Hz			5800 Hz		5600 Hz	
PLP	SV	14.16	14.28	14.45	14.22	14.43	14.96	16.68
	Low-pass	14.22			14.34		14.54	
MFCC	SV	14.35	14.91	14.89	14.94	15.56	16.18	19.7
	Low-pass	14.15			14.45		14.86	

Lets examine a different approach of estimating the contribution of spectral valleys. The signals were at first compressed at the corresponding bitrates, but the cut-off frequency was uniformly set at 8 kHz. There were two main reasons for this decision. First, it is more natural for a user to simply compress audio and not worry about artificial bandwidth extension/reconstruction. Second, I wanted to find out if by not limiting the bandwidth of MP3 speech, I could improve the system's performance. The primary argument for this approach was that even if high-frequency information was heavily distorted, it was still present and could therefore contribute to the overall performance. However, this procedure is generally considered suboptimal and is advised against by the music community.

The common sense advises that the available bandwidth needs to be limited, otherwise it would introduce other unwanted artifacts. The argument against full-bandwidth MP3 was thus based on subjective listening tests and as such might not hold true for ASR. Figure 7.1 shows the obtained error rates for full-band and limited MP3 speech. While the MFCC features contributed from removing the heavily distorted HF bands for lower bitrates, the situation was reversed for the PLP ones. The process did not affect signals with higher bitrates in any significant way. However, full-band MFCCs still did not outperform PLPs in any setup, which leads to conclusion that LP filtering is beneficial for MP3 ASR as well.

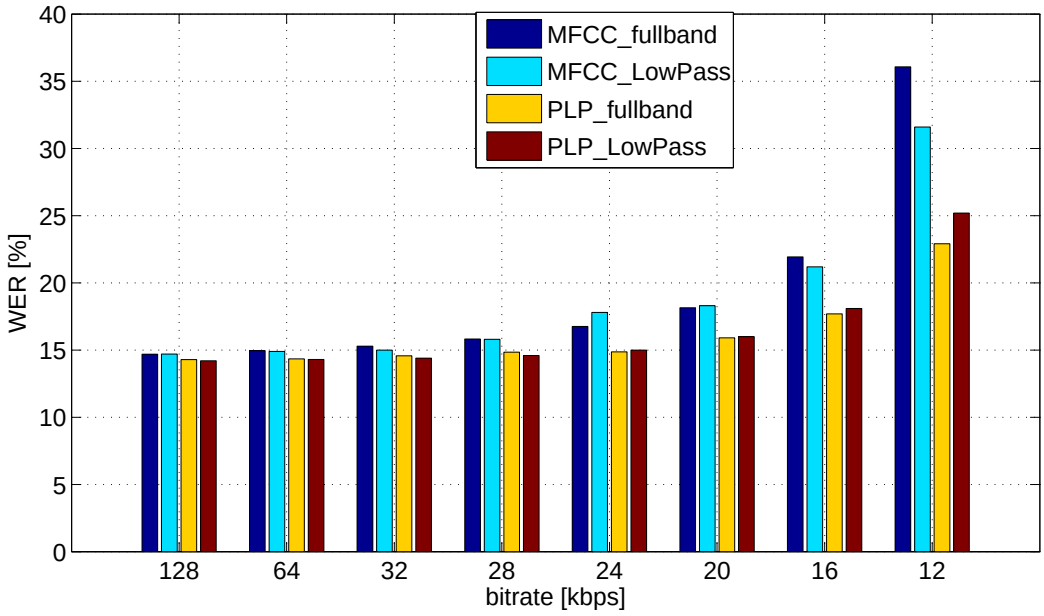

Figure 7.1: Comparison of fullband and standard (LP-filtered) MP3

While the f_{cutt} for the 12 kbps bitrate was around 5.6 kHz, the following experiments went even further to demonstrate that current ASR systems are reasonably resistant to bandwidth limitations. The used signals were recorded with a 16 kHz sampling frequency, which meant that $f_{max} = 8$ kHz, but many real-life ASR applications run successfully over a telephone channel, which limits the usable bandwidth down to 3.4 kHz. Therefore, I decided to stop the experiments at the 4 kHz threshold. The achieved results are summarized in Tab 7.2. The MFCC and PLP coefficients showed very similar results for the full-band signals. The error rate curves followed the same trend which approximated an exponential function with a breakpoint around 6 kHz. The only difference was in the rate of increase, which was steeper for the MFCC features. It could be concluded that the substantial increase in *WER* for MP3 speech was not caused by low-pass filtering alone.

Table 7.2: *WER* for low-pass filtered speech

f_{cut}	full	7kHz	6kHz	5kHz	4kHz
MFCC	14.6	14.7	15.1	16.5	19.5
PLP	14.3	14.3	14.3	15.3	17.5

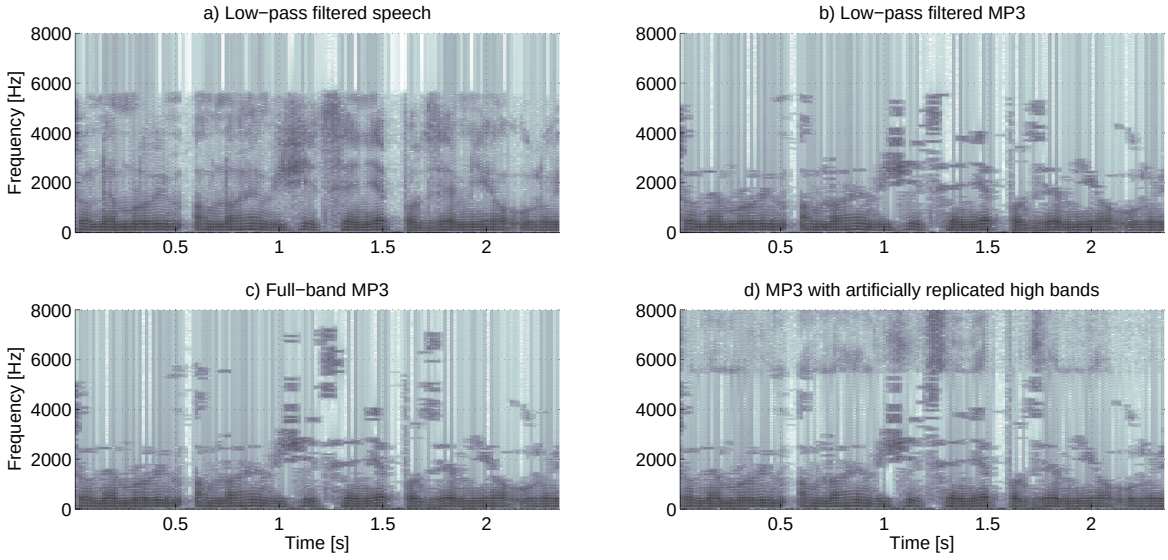

Figure 7.2: Illustrative spectrograms for all the studied cases of degrading the speech signal

To illustrate the effects of all analysed setups; LP filtered speech, normal MP3, full-band MP3, and artificially replicated high frequency; Figure 7.2 plots sample spectrograms of the same signal for each of these scenarios. Based on these series of experiments, I have concluded that although the overall poor ASR performance on MP3 speech was caused by a non-linear combination of both studied artifacts, the contribution of spectral valleys was far more detrimental. As a result, I have decided to focus on compensating this distortion as a primary way of improving the recognition results for MP3 speech.

7.2 Description of SSD

Following the nature of the distortion described in the previous section, along with already achieved results in this task, led to conclusion that the reconstruction of missing low and middle frequency components was the key to robust MP3 speech recognition. Numerous works on this topic have been published in the field of audio coding, i.e. [93, 95], but to my knowledge, no of these algorithms have been tried in the field on ASR. Following the research presented in these works, I decided to design an algorithm which could be easily incorporated into the current parametrization schemes for MFCC and PLP features. In order to combine all of these aspects, I decided to modify the uniform dithering technique to dither only the selected frequency bands with automatically estimated amount of noise.

Figure 7.3 presents the full block scheme of the designed SSD algorithm. The principal idea was to use to the LPC model to decompose the signal into the spectral envelope and the residual signal (exc), detect the zero-energy bands in the residual signal and patch them to get the compensated signal. To accomplish this goal, I had to designed two separate blocks: the zero-bands detector and gain estimation/compensation block.

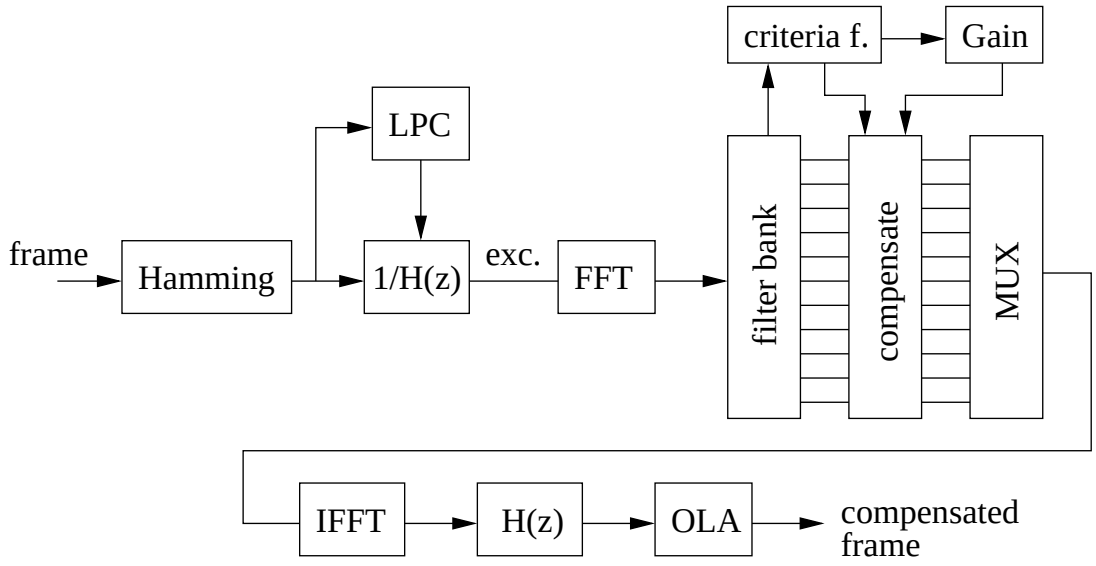

Figure 7.3: Block diagram of the SSD compensation technique

The **Zero-energy Band Detector** block was composed of LPC coefficient estimation block, an analysis filter with the frequency response 1/H(z), an FFT computation block, a frequency filter bank with linear frequency axis and the criteria function estimation. The **Gain Estimation and Reconstruction** block was composed of the gain estimation, frequency domain compensation block, frequency domain multiplexor and the IFFT block, a synthesis filter with the frequency response H(z) and finally the time-domain reconstruction using the OLA method. Each of these block is described in greater detail in the following text.

7.2.1 Zero-band detection

The analysis of coded speech showed that the discussed distortions effect both the spectral envelope and the residual signal at the same time. Although these changes were detectable in both spectral domains, I chose to base my detection algorithm on the residual signal. The residual signal for AR process is a decorrelated signal with flat spectral characteristics with a zero mean value. However, the actual values in the distorted bands approximated a flat line whose trend could be approximated by a linear function with gradient $\rightarrow 0$. It allowed me to employ a criteria function based on the smoothness of the spectral curve and classify distorted bands using a fixed threshold. In the first step, the input frame was weighted by the Hamming window and then inverse filtered with a simple LPC filter with the frequency response $1/H(z)$ in order to remove the vocal tract characteristics and to obtain the excitation signal. The order of the LPC filter was set to 10 and its frequency characteristics had the form of:

$$H(z) = \frac{\sqrt{E_p}}{1 + \sum_{k=1}^{p} a_k z^{-k}}, \tag{7.1}$$

where E_p was the power of the prediction error. The frame was then converted to log-magnitude spectral domain where it was windowed once more to analyse each frequency bands separately. The threshold was then applied to classify each band as either good or corrupted. Let's assume the frequency band b, which contains spectral components f_1 through f_2, was extracted from the spectra of the excitation in a frame. The criteria $crit(b)$ used by the detector was computed as:

$$crit(b) = \sqrt{\sum_{f=f_1}^{f_2} (exc(f) - exc(f\text{-}1))^2}. \tag{7.2}$$

The decision function was then defined as:

$$mask(b) = \begin{cases} 0, & crit(b) \geq Thr, \\ 1, & crit(b) < Thr, \end{cases} \tag{7.3}$$

where *Thr* was a fixed threshold which was set as a constant for all bitrates. The threshold value was estimated empirically and it was the only manually optimized variable in the algorithm. The number of spectral components in a band was crucial for robust estimation. If the band was too narrow, the detector returned too many false alarms. On the other hand, if the band was too wide the location of zero-bands became inaccurate. The experiments showed that a band containing 4 spectral bins gave a reasonably precise estimation for 16 KHz sampled signals. Four spectral bins in a band equalled the frequency resolution of 125 Hz. Another possibility was to use an overlapping frequency windowing with a longer frame and shorted shift. However, I did not use this option.

7.2.2 Gain estimation and compensation

In the next step, the masking function was used to estimate the gain of added noise and to patch the corrupted bands. The gain was estimated as a simple average from the undistorted bands. This average was used to patch the excitation signal. The added noise had uniform distribution with zero mean value and unit variance. We can then express the compensated excitation *exc(b)* as follows:

$$exc(b) = \begin{cases} exc(b) + G * noise, & mask(b) = 1, \\ exc(b), & mask(b) = 0. \end{cases} \tag{7.4}$$

The excitation signal was then put together in the multiplexer block and the compensated frame was obtained through forward LPC filtering with the same coefficients and the impulse response $H(z)$. The whole compensated signal was reconstructed using the OLA method and a new set of features was extracted from the compensated signals.

7.2.3 Analysis of SSD blocks

Figure 8.1 illustrates the estimated masks (ones or zeroes) from the same signal as the spectrograms in Figure 7.2. It can be noted that the selected criteria function was not only able to detect the suppressed bands accurately but did so only in the speech frames. Both starting and ending parts of the recording which contained silence were correctly classified as "good". These attributes enhanced the selectivity of the algorithm as only the speech frames were subject to subsequent compensation.

The lack of verifiable labels disallowed me to use the standard detection measures (true positive, false positive, false positive, false negative) to evaluate the detection accuracy and thus I had to rely on comparing the achieved values among bitrates. The evaluation of each SSD blocks (Zero-band detector and Gain estimation) is summarized in Table 7.3. The presented statistics were computed on Czech test dataset. The measure I used to evaluate the Zero-band detector was based on counting the number of corrupted bands to the total number of bands in a signal. In another words, I worked with *time*×*frequency* patches in the estimated mask across the whole signal. If we consider the used segmentation setup for feature extraction and the frequency resolution of the Zero-band detector, then one band was defined as a patch of $16 \times 125\ ms \times Hz$ dimension. I chose this measure in order to account for different lengths of signals. The statistical values displayed an overall upward trend in the average number of corrupted bands which was in accordance with the initial assumption. Figure 8.3 also plots the histograms of the relative number of corrupted bands in a signal.

Table 7.3: Percentage of corrupted bands and gain G in the evaluation corpora as estimated by SSD

	128k	64k	32k	28k	24k	20k	16k	12k
Zero-bands	4.37	4.43	6.17	7.88	13.09	15.76	19.72	25.09
$(\mu \pm \sigma)$ [%]	±1.18	±1.23	±1.84	±2.44	±3.93	±4.63	±5.78	±7.05
Gain	18.44±	18.40	17.96	17.74	17.49	17.21	16.95	16.69
$\mu \pm \sigma$ [-]	±1.4	±1.38	±1.44	±1.52	±1.82	±1.93	±2.13	±2.51

On the other hand, the gain values were estimated only from positively detected bands. The estimated gains displayed an opposite trend as the average values were decreasing rather than increasing. There are several possible explanation for this. First, the detector misclassified the pauses in between the words as zero-bands. Second, the detector failed to detect all occurring corrupted bands. Third, previous experiments have demonstrated that MP3 effects unvoiced consonants much more than voiced phonemes and thus more unvoiced speech segments were included. In all cases, the actual gain would be estimated from a series of comparatively smaller values (pause vs. speech/ unvoiced vs. voiced) or even a series of zeroes (zero-band) which would explain the downward trend. It is also important to realize that these values were computed in log-magnitude spectral domain and were not comparable to the dithering values R for uniform dithering. Figure 8.2 plots the histograms of estimated gains on a per frame basis. We can also notice that the standard deviation is increasing in both cases.

7.3 Evaluation of SSD performance

The performance of proposed SSD algorithm was evaluated and compared with the method based on uniform dithering. Both dithering techniques were implemented either at the level of the feature extraction tool (uniform dithering within CtuCopy tool) or separately in the MATLAB environment (Spectrally Selective Dithering). The algorithms were evaluated for both GMM-HMM and DNN-GMM architectures. The experiments were performed for Czech, English and German in order to demonstrate that the compression degraded different languages in the same manner.

The Czech GMM-HMM system used the previously described common setup. Signals for the English experiments were from the WSJ database [99]. I used the full 81-hour train-si284 set plus the eval92 set as my test set. The data for German were taken from the GLOBALPHONE database [100] and the training set contained 14.9 hours of speech. The German results are presented on 1.5 hours of data from the eval set. The signals from all databases were recorded in acoustically clear conditions with a 16 kHz sampling frequency and 16 bit precision. This consistency in the sampling frequency and the bit-depth across the languages allowed me to use the same number bins in a band for SSD.

The ditherings were applied at the feature extraction level before any other normalization and transformation. The optimal value for uniform dithering was determined by manually increasing the power of added noise until a minimal error rate was achieved. The English AM contained 39 starting phone and the German AM was built for 41 monophones. As for the discriminative training, I used only the MPE criteria as it showed slightly better overall results in the previous analyses. The DNN-HMM hybrid system was built upon 40 dimensional baseline features which were later speaker-adapted by fMLLR. The transformation matrices were estimated during the SAT stage of GMM-HMM training. The DNN topology consisted of an input layer with 440 units (for the 40-dimensional fMLLR features with the context of 5 frames with mean and variance normalization), followed by 6 hidden layers with 2048 neurons per layer and the sigmoid activation function. The process of building the DNN-HMM system began with the initialization of hidden layers that employed Restricted Boltzmann Machines and then added the output layer. The process continued with frame cross-entropy training and ended with sMBR sequence-discriminative training. More detailed information can be found in Kaldi recipe s5 [62].

For the English experiments, I used the trigram LM available in WSJ corpora [99]. The German trigram LM was created using the Rapid Language Adaptation Toolkit [101]. The complete information about the test sets and the LMs used is summarized in Table 7.4.

7.3.1 Results for Czech

The comparison analysis of matched and mismatched conditions is summarized in Table 7.5, and there are several interesting things to see there. First of all, the advantage of using matched training was clear only for bitrates lower than 24 kbps. for both PLPs and

Table 7.4: Summary of used setups for different languages

	AM			LM		
lang.	train	test	phn.	voc.	n-gram	OOV
CZ	72h	2h	44	340k	2	1.2
ENG	81h	0.7h	39	125k	3	1.8
GER	14.9h	1.5h	41	38k	3	2.2

MFCCs. This observation is somewhat contrary to the my initial expectations. While the absolute *WER* over mismatched conditions was within 1%, and had a decreasing tendency, the higher bitrates still suffered from the process. This trend might have occurred due to slightly more varying acoustic environment of used data. Second, the PLPs once again outperformed the MFCCs and the absolute *WER* difference had an increasing tendency, starting at 0.5% for 128 kbps and ending a 6.4% for 12 kbps.

Table 7.5: *WER* in matched & mismatched training for Czech, GMM system

	MFCC		PLP	
	match	mismatch	match	mismatch
RAW	14.6	-	14.2	-
128 kbps	15.7	14.7	14.5	14.2
64 kbps	15.8	14.9	14.8	14.3
32 kbps	16.5	15.0	15.0	14.4
28 kbps	16.7	15.8	15.0	14.6
24 kbps	17.1	17.8	15.0	15.0
20 kbps	17.1	18.3	15.0	16.0
16 kbps	19.7	21.2	16.8	18.1
12 kbps	22.9	31.6	18.9	25.2

The previous section has summarized the results with a gradually improved AM using the discussed training chain. Thus, in this section I present only the results with the final, discriminative AM and evaluated only the option of dithering signals with either UD or SSD. However, I also added the results for DT model without the fMLLR to demonstrate its effect on the error rate. The acronyms for particular setups are listed below.

- GMM-SI - speaker-independent
- GMM-SD - speaker-dependent
- GMM-UD - speaker-dependent + UD compensation
- GMM-SSD - speaker-dependent + SSD compensation

The contribution of both discussed front-end compensation techniques is summarized in Table 7.6. The results obtained with uniform dithering were ambiguous as I observed an actual increase in *WER* for all rates aside from 20 kbps and 12 kbps. This behaviour, along with the need to set the dithering value R properly, was the reason why the technique is used rarely, if at all. However, it is important to realize that these conclusions were

valid for the final MPE trained AMs as opposed to my previously discussed results from Chapter 6 where I observed statistically significant improvements for weakly refined AMs.

Table 7.6: Comparison of UD and SSD for Czech GMM system

	WER [%]				*WERR [%]*
	GMM-SI	GMM-SD	GMM-UD	GMM-SSD	GMM-SSD
128 kbps	18.20	14.29	14.24	**14.18**	0.7
64 kbps	18.43	14.35	14.27	**14.22**	0.9
32 kbps	18.70	14.45	14.5	**14.25**	1.4
28 kbps	19.21	14.64	14.64	**14.52**	0.8
24 kbps	19.87	15.02	15.07	**14.97**	0.3
20 kbps	21.27	16.02	15.88	**15.72**	1.9
16 kbps	25.19	18.15	18.27	**17.83**	1.8
12 kbps	34.73	25.20	24.04	**23.35**	7.3

The evaluations demonstrated that the proposed SSD algorithm was able to accurately detect low-energy areas and estimate the amount of noise needed to compensate the distortion. The proposed SSD technique displayed an absolute *WER* reduction over baseline system ranging from 0.05% for 24 kbps to 1.85% for 12 kbps. The second important observation was that the SSD never increased the error rate. An absolute margin of 0.69% between uniform dithering and SSD was the highest for the lowest bitrate. Although the presented improvements were relatively small, the nature of distortion have lead me to the conclusion that potentially even greater error reductions could be achieved by compensating SV with more advanced techniques.

Neural net based AMs have displayed a much greater robustness against adverse environmental conditions without any pre-processing than their GMM predecessors. This feature has naturally raised a question whether the DNN-HMM system could still benefit from any feature-level modification technique. To answer this question, I used the same experimental protocol and similar acronyms as before, re-summarized for clarity in the points bellow. The recognition results are summarized in Tab 7.7. Once again, SSD was able to improve the recognition results. This conclusion proved to be true for all bitrates aside from 20 kbps and also the average *WERR* was much higher than for the GMM-HMM system. The UD once again failed to deliver a consistent improvement and for most cases it even worsened the results. The highest absolute improvement of 1.87% was achieved for 12 kbps. However, it is also important to notice that the DNN-HMM system achieved worse results than the GMM-HMM system in general. The most likely reason why this situation occurred was due to the fact that DNN systems are known to be data-hungry. Failing to provide sufficient amount of data resulted in a system with higher error rates than a comparable GMM system. Simply put, I lacked data to properly train the system.

- DNN-SI - speaker-independent
- DNN-SD - speaker-dependent
- DNN-UD - speaker-dependent + UD compensation
- DNN-SSD - speaker-dependent + SSD compensation

Table 7.7: Comparison of UD and SSD for Czech DNN system

	WER [%]				*WERR [%]*
	DNN-UD	DNN-SD	DNN-UD	DNN-SSD	DNN-SSD
RAW	18.98	**17.24**	17.61	17.56	-1.9
128 kbps	19.5	17.77	17.87	**16.94**	4.6
64k bps	19.95	17.8	18.03	**17.00**	4.5
32 kbps	20.28	17.88	18.03	**17.07**	4.5
28 kbps	20.89	18.05	18.26	**17.19**	4.8
24 kbps	21.64	18.35	18.46	**17.35**	5.4
20 kbps	22.47	**19.06**	19.01	19.32	-1.4
16 kbps	26.69	21.46	21.25	**20.71**	3.6
12 kbps	37.76	26.78	26.43	**24.91**	7

Figure 7.4 plots the results of GMM and DNN systems for all setups. The graphs illustrate the advantages of using SSD compensated features for all studied bitrates and architectures. The figures also illustrate the advantage of SSD over the simple UD and show that speaker adaptation provided the major portion of improvement out of all studied acoustic modelling techniques.

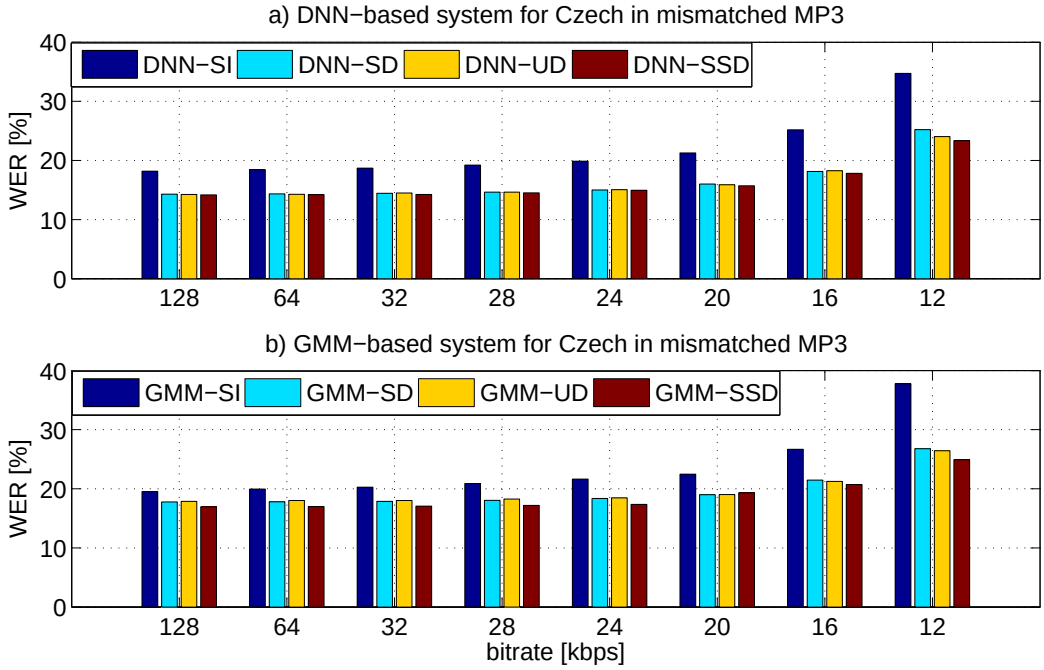

Figure 7.4: Comparison of GMM and DNN systems for Czech

7.3.2 Results for English & German

The results of matched and mismatched recognition for English and German languages are summarized in Table 7.8, and there are several interesting things to see there. The

English and German displayed the expected trend, where all the matched conditions outperformed the mismatched ones for virtually all bitrates. For low bitrates especially, the difference was significant, and the improvement of using matched AMs reached as high as 49% relatively. The second notable trend was a relatively smaller degradation of PLPs over MFCCs and a much slower increase in *WER* as a function of bitrate for both features. Based on these result, it could be concluded that training bitrate-specific AMs was potentially a viable option for these languages, since the overall improvements in error rates were significant.

Table 7.8: *WER* in matched & mismatched training for English & German GMM system

	ENG				GER			
	MFCC		PLP		MFCC		PLP	
	match	mismat.	match	mismat.	match	mismat.	match	mismat.
RAW	8.5	-	8.3	-	18.7	-	18.8	-
128 kbps	8.7	10.0	8.7	9.4	18.9	18.8	18.8	19.1
64 kbps	8.8	10.4	8.5	9.3	19.1	19.1	18.8	19.1
32 kbps	8.7	10.9	8.7	9.8	19.2	19.4	19.1	19.5
28 kbps	8.7	11.6	8.7	10.1	19.9	19.9	19.0	19.9
24 kbps	9.3	12.3	8.9	11.0	20.2	23.0	19.3	20.5
20 kbps	9.7	13.3	9.1	11.7	21.0	26.2	20.1	21.7
16 kbps	11.1	16.4	9.9	13.8	22.7	33.1	22.8	25.4
12 kbps	11.5	22.6	10.4	17.1	26.6	46.0	23.3	42.6

The full results for mismatched system with the application of SSD are summarized in Table 7.10 and Table 7.9. Tables use the same acronyms as for the Czech. The initial error rates for RAW speech differed for each language. The large amount of training data for English, along with its relatively simple grammatical structure, were the primary reasons why it achieved the best *WER* of 8.3%. The results for German were significantly worse, with *WER* of 18.8%. The results for German were more likely influenced by the smaller amount of training data. Despite of this, the overall contribution of the dithering techniques was clearly proven for all the studied languages and bitrates.

Table 7.9: Comparison of UD and SSD for German GMM system

	WER [%]				*WERR [%]*
	GMM-SI	GMM-SD	GMM-UD	GMM-SSD	GMM-SSD
RAW	21.85	18.86	18.81	**18.79**	0.3
128 kbps	22.13	19.10	19.13	**18.85**	1.3
64 kbps	22.33	19.12	19.32	**19.01**	0.6
32 kbps	22.68	19.56	19.64	**19.21**	1.8
28 kbps	23.54	19.94	19.95	**19.57**	1.9
24 kbps	24.48	20.53	20.69	**20.05**	2.3
20 kbps	26.89	21.72	21.78	**21.08**	3
16 kbps	33.61	25.45	25.66	**24.83**	2.5
12 kbps	51.02	42.68	**34.87**	35.86	16

Aside from a single case (German and 12 kbps), the SSD algorithm outperformed the UD technique and DT models by a slight margin for all studied languages and bitrates. The relative average improvement was highly language-specific, 12.5% for English, and 1.73% for German. On the other hand, the relative improvement of SSD over SA features was more consistent. It reached at maximum 16% for German and 15.3% for English. However, it is also interesting to note that the English contributed from SSD much more on average than German did. Based on these results, it can be concluded that SSD compensated features ertr more similar to non-compressed features and were a preferable solution for MP3 recognition.

Table 7.10: Comparison of UD and SSD for English GMM system

	WER [%]				*WERR [%]*
	GMM-SI	GMM-SD	GMM-UD	GMM-SSD	GMM-SSD
RAW	10.26	**8.37**	8.40	8.51	-1.7
128 kbps	11.60	9.49	9.51	**8.32**	12.3
64 kbps	11.82	9.37	9.65	**8.35**	10.9
32 kbps	11.95	9.89	9.67	**8.39**	15.1
28 kbps	12.74	10.14	9.98	**8.67**	14.5
24 kbps	13.47	11.02	11.21	**9.54**	13.4
20 kbps	14.23	11.70	11.70	**10.04**	14.1
16 kbps	17.65	13.82	13.47	**11.72**	15.3
12 kbps	25.16	17.12	16.39	**15.47**	9.6

Since the previous analyses with Czech DNNs have proved that 72 hours of data was not sufficient to properly train the net, I excluded German from the next analysis and focused solely on English. The comparison between matched and mismatched conditions is shown in Table 7.11. The qualitative trends previously observed in GMM systems held true for DNN as well. The matched training significantly outperformed the mismatched training, starting at 0% for 128 kbps and ending at a relative 60.5% for 12 kbps. Another comparison could be drawn against the matched GMM, where the overall difference between DNN and GMM systems was up to 2% and the neural nets proved to outperform the GMM architecture. However, the absolute *WER* difference decreased with decreasing bitrate (only 0.1% for 12 kbps), which led me to the conclusion that the constraints of current MP3 recognition are dependent more on bitrate and less on the choice of ASR architecture. This observation proved true for the mismatched system as well where the speech compression process nearly quadrupled the error rate from 6.8% to 26.1%. In comparison, the error rate for GMM dropped from 8.3% to 17.1%. In other words, the clean-conditioned DNN outperformed the GMM, but the compressed GMM outperformed the DNN. However, even this observation could be easily explained it we realize that GMM is a generative model while DNN is a discriminative model.

Table 7.11: *WER* in matched & mismatched training for English, DNN architecture

	RAW	128k	64k	32k	28k	24k	20k	16k	12k
match	6.8	7	7	7.2	7.23	7.6	7.96	8.2	10.3
mismatch	-	7	7.1	7.4	7.4	8.5	9.2	11.9	26.11

The final experiment involved the LVCSR task with the DNN system in mismatched conditions and with compensated features. The fMLLR adaptation provided the major portion of the improvement, by up to a relative 42%. The application of a feature compensation technique, in the form of either UD or SSD, yielded mixed results. The former often worsened the actual recognition score and the latter brought only marginal improvements. The *WERR* for SSD reached 2.9% on average, and the only notable improvement was observed for 12 kbps, at 10.27%. For comparison, the relative improvement of SDD for the GMM system and English was 13%. Figure 7.6 plots the results of GMM and DNN systems for all setups. The graphs illustrate the advantages of using neural nets for all bitrates aside from the very lowest.

Table 7.12: Comparison of UD and SSD for English DNN architecture

	WER [%]				*WERR [%]*
bitrate	DNN-UD	DNN-SD	DNN-UD	DNN-SSD	DNN-SSD
RAW	8.33	**6.82**	7.11	6.93	-1.6
128 kbps	9.06	7.07	7.28	**6.84**	3.3
64 kbps	9.57	7.16	7.30	**6.91**	3.5
32 kbps	9.75	7.44	7.58	**7.25**	2.6
28 kbps	9.94	7.46	7.78	**7.37**	1.2
24 kbps	12.81	8.56	9.60	**8.49**	0.8
20 kbps	14.42	9.29	10.31	**9.06**	2.5
16 kbps	20.70	**11.94**	13.22	12.00	-0.5
12 kbps	38.61	26.17	**23.36**	23.50	10.2

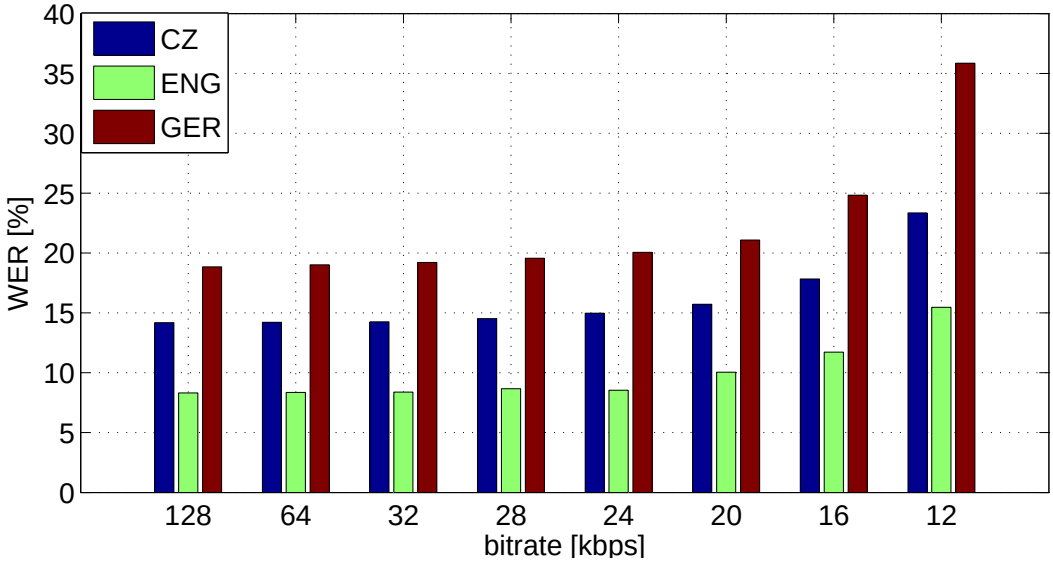

Figure 7.5: Results for SSD-compensated features in GMM systems and all languages.

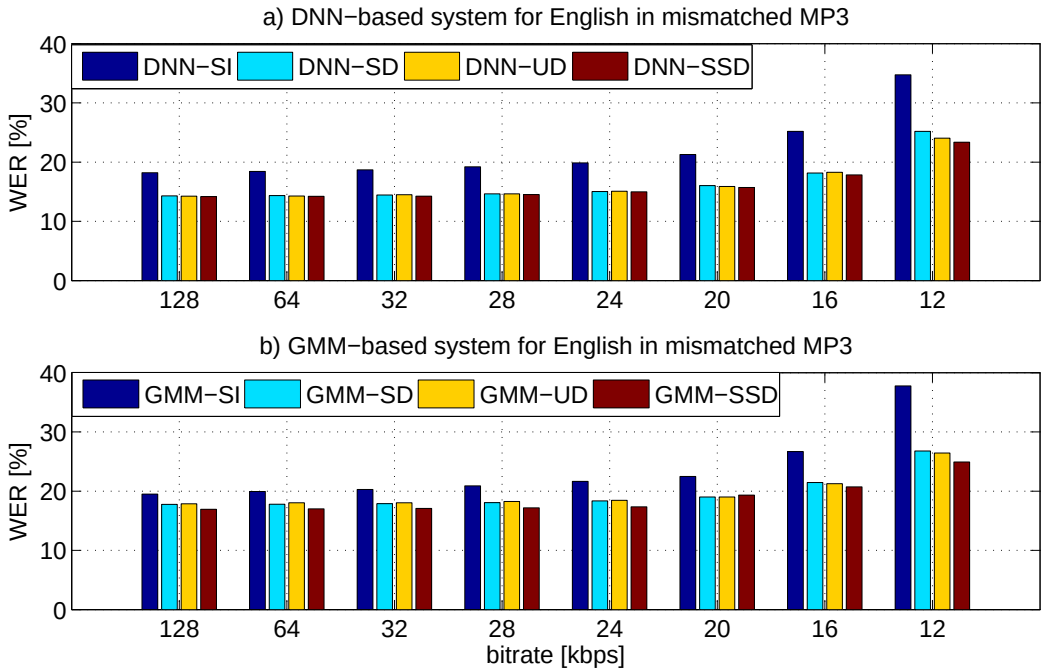

Figure 7.6: Comparison of GMM and DNN systems for English

7.3.3 Summary

Figure 7.5 plots the results for SSD-compensated features. Although the baseline *WERs* differed greatly, all languages displayed the same general trend. The breakpoint for the exponential increase occurred around 24 kbps. Based on these results, it can be concluded that MP3 compression was largely language-independent. The main conclusions from these analyses can be summarized as follows.

- The MP3 compression introduced two main artifacts that degraded speech: low-pass filtering and spectral valleys. The initial analysis evaluated their impacts separately and showed that spectral valleys influenced ASR performance on a much greater scale. Thus, the proposed algorithm focused on reducing the depth of these valleys.

- The results proved that the SSD technique could consistently outperform both adapted features and uniform dithering at the same time. The selective dithering method was particularly effective for GMM-based systems, where significant reduction of *WER* values was observed for all languages and bitrates. The only exception was the 12 kbps German test set where the uniform dithering outperformed SSD, albeit only by a small margin. Aside from this case, the results achieved with the UD were inconclusive.

- The results with matched and mismatched training showed that using matched AMs could bring considerable *WERR*. However, the main problem was the reliable detection, mainly in cases where different or even multiple encoders were used.

- A complete set of tests with DNN systems was performed for Czech and English only as I had the most amount of data for them. The neural nets outperformed the GMMs in general, but proved to be more sensitive to the quality of the input data in the mismatched scenario and hence less successful in case of very low bitrates. However, the results might have been influenced by the available amount of training data. It is possible that with more training data, the DNNs would be more robust even in these extreme situations. The contribution of SSD was proved only for Czech while English displayed mixed results.

7.4 SSD for Distant Microphone MP3

The performed analyses with MP3 so far have been focused on a high quality input audio recorded with a close-talk microphone. However, MP3 coding is often used to compress signals from conference meetings or lectures which are recorded with a middle-to-far distance microphones. The previous chapters have analysed the performance of discussed AM techniques in similar scenarios and for two different environment (private and public) and this section combines the two scenarios and explores the usability of MP3 for middle-to-far distance microphone recognition. The analyses were performed only for the signals from a clean environment since MP3 is unlikely to be used for car recordings. A closer look revealed that this recognition tasks suffered from all discussed distortions (additive/convolution noise, bandlimiting and spectral valleys). The ASR used the previously described setup when the only difference was the application of ESS in addition to the already described parametrization. Thus, the experiments included results with and without the application of this algorithm. Its potential contribution was also studied in conjunction with SSD. In case both techniques were used at the same time, the signals were compensated with SSD at first and then with ESS.

7.4.1 Results for CS2 channel

The results for CS2 channel are summarized in Table 7.13 for the most important stages of AM development. First, the direct comparison of CS0 and CS2 channels showrf that only the 12 kbps and 16 kbps bitates were severely affected by the choice of a microphone. The absolute *WER* difference between CS0 and CS2 channels for 16/12 kbps was 2.53% and 6.76% respectively. The margins for the higher bitrates were within 2% and decreasing very slightly with a decreasing bitrate. However, even the highest bitrate still showed a performance degradation of 1.7% over the CS0 channel. Finally, MPE outperformed the bMMI for all studied bitrates once again. Also, the total *WERR* between the baseline and MPE level models reached 49.2% for 128 kbps and slowly increased up to 60% for 12 kbps. It has to be noted however, that the major portion of improvement was a result of LDA+MLLT and SAT application. Table 7.14 summarizes the results for SSD compensated signals. The application of the SSD brought a significant improvement (4.5% for MPE) only for the lowest bitrate.

Table 7.13: Results for MP3 for CS2 channel

	128k	64k	32k	28k	24k	20k	16k	12k
Baseline (SI)	31.39	30.34	30.58	30.45	31.72	33.93	45.82	78.97
SAT (SA)	25.82	24.52	24.69	24.83	25.68	26.71	30.23	42.56
MPE (SA)	15.95	15.82	15.76	15.89	16.95	17.86	20.68	31.96

Table 7.14: Results for MP3 compensated with SSD for CS2 channel

	128k	64k	32k	28k	24k	20k	16k	12k
Baseline (SI)	31.26	30.35	30.39	30.35	31.62	33.2	43.53	74.04
SAT (SA)+SSD	25.41	24.34	24.59	24.64	25.55	26.49	29.32	36.87
MPE (SA)+SSD	15.95	15.77	15.81	15.92	16.99	17.75	20.50	27.49

The previous results on far-microphone recognition for public environment confirmed the advantage of using ESS in conjunction with the advanced AM for CS2 microphone. The experimental setup of ESS remained the same as in the previous chapter and the achieved results are summarized in Table 7.15. There are several interesting things to note. The application of ESS did not bring any consistent improvement, regardless of the bitrate. However, the absolute *WER* between standard and ESS compensated features had a tendency to drop with decreasing bitrate. It reached 1.8% for 128 kbps rate but only 0.25% for 12 kbps.

Table 7.15: Results for MP3 compensated with ESS for CS2 channel

	128k	64k	32k	28k	24k	20k	16k	12k
Baseline (SI)	33.19	32.57	32.29	32.81	33.98	36.4	49.07	79.22
SAT (SA)	25.82	24.52	24.69	24.83	25.68	26.71	30.23	42.56
MPE (SA)	16.31	16.18	16.29	16.52	17.39	18.31	20.79	32.58

The last experiments included the application of both SSD and ESS algorithms. Due to the nature of both algorithms, the ESS was applied after the speech wave was compressed and compensated using the SSD. The achieved results are summarized in Table 7.16. It can be noted that applying the SSD+ESS consistently worsened the error rates by about 1%, regardless of the bitrate. However, the results were still better than just for the sole application of ESS. The performed experiments have proved that using the combination of SSD+ESS did not bring any improvements a middle distance microphone.

Table 7.16: Results for MP3 compensated with ESS+SSD for CS2 channel

	128k	64k	32k	28k	24k	20k	16k	12k
Baseline (SI)	33.08	32.27	32.16	32.82	33.72	35.8	46.95	78.75
SAT (SA)+SSD	26.46	25.59	25.95	26.04	26.78	27.95	30.43	37.97
MPE (SA)+SSD	16.41	16.12	16.30	16.63	17.32	18.26	20.64	28.57

7.4.2 Results for CS3 channel

The final set of MP3 experiments involved using the CS3 channel and the results for the standard features from CS3 channel are summarized in Table 7.17. The average increase in absolute *WER* over the CS0 channel was much more significant this time and also apparent for all bitrates. This trend was most prominent for baseline system were even the highest bitrate. The absolute difference in *mbox WER* between the final MPE trained CS0 and CS3 models increased from 22.17% for 128 kbps to 34.32% for 12 kbps. Also, the total *WERR* between the baseline and MPE level models reached 34.4% for 128 kbps but reached practically the same value of 36.8% for 12 kbps. This findings demonstrated that the presence of additive and background noises had a non-linear degradation affect on the MP3 compressed speech.

Table 7.17: Results for MP3 for CS3 channel

	128k	64k	32k	28k	24k	20k	16k	12k
Baseline (SI)	56.37	56.07	56.03	56.29	57.13	59.13	68.58	91.24
SAT (SA)	48.65	48.08	47.91	47.79	49.28	50.12	55.84	64.85
MPE (SA)	36.35	36.15	36.17	36.73	38.26	40.67	46.61	57.67

Table 7.18: Results for MP3 compensated with SSD for CS3 channel

	128k	64k	32k	28k	24k	20k	16k	12k
Baseline (SI)	56.17	55.96	56.11	56.19	57.19	58.62	67.01	87.62
SAT (SA)+SSD	48.47	48.05	47.70	47.83	49.34	49.80	54.91	60.91
MPE (SA)+SSD	36.21	35.91	36.16	36.81	38.18	40.22	45.60	53.54

The results for SSD compensated signals for CS3 channel are summarized in Table7.18. The application of SSD compensation followed the same trend as for CS2 channel when only the lowest bitrate displayed a significant error reduction of about 4%. The 16 kbps rate also displayed a marginal improvement of 1.01% but the remaining bitrates were mostly unaffected. This observation held true for all stages of AM refinement. The analyses for CS3 channel with ESS compensated features are summarized in Table 7.19 and with SSD+ESS features in Table 7.20. The achieved results followed the trend already observed for CS2 when the application of ESS increased the error rates even further. Also, the combination of SSD and ESS proved to yield results which were much more similar to sole application of SSD, although slightly worse. Thus, it could be concluded that the application of ESS was not advised even for far-distance microphone compressed speech speech.

Table 7.19: Results for MP3 compensated with ESS for CS3 channel

	128k	64k	32k	28k	24k	20k	16k	12k
Baseline (SI)	57.16	56.99	56.96	57.07	58.07	59.58	71.52	94.91
SAT (SA)	48.01	47.98	47.87	47.99	42.90	44.54	56.27	67.1
MPE (SA)	36.36	35.97	36.43	36.53	36.67	37.98	47.35	60.15

Table 7.20: Results for MP3 compensated with ESS+SSD for CS3 channel

	128k	64k	32k	28k	24k	20k	16k	12k
Baseline (SI)	56.97	57	56.88	57.12	57.8	60.09	69.69	92.22
SAT (SA)+SSD	48.04	48.03	47.63	48.07	49.02	50.36	55.23	62.74
MPE (SA)+SSD	36.22	36.02	36.33	36.51	38.02	40.64	46.02	54.78

7.4.3 Summary

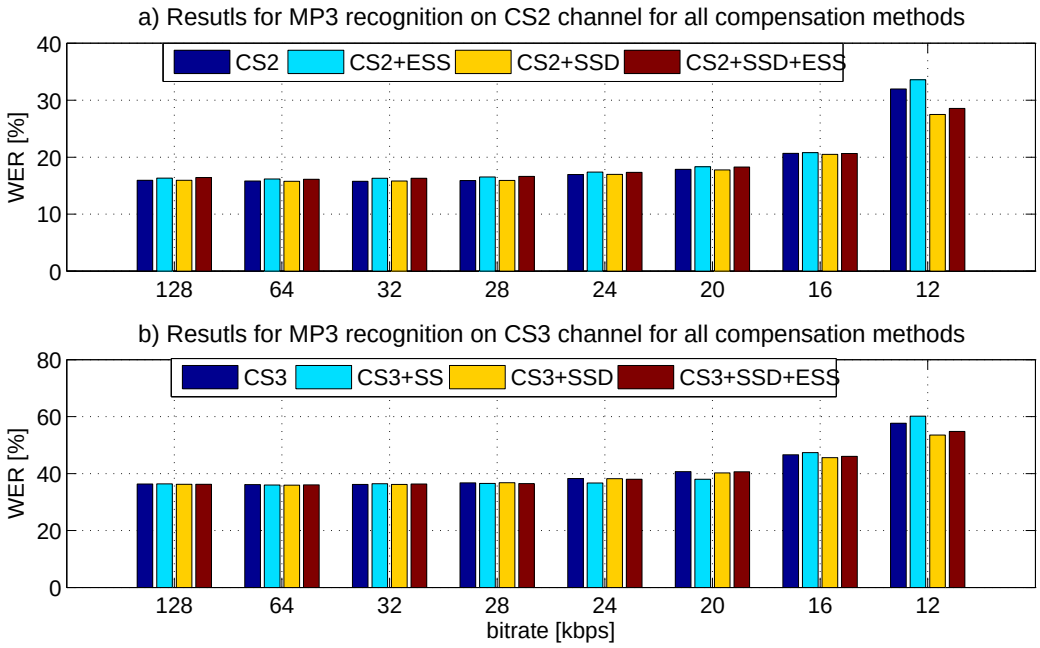

Figure 7.7: Results for best AMs on a) CS2 and b) CS3 channels

The results for the best AMs are illustrated in Figure 7.7. The main conclusions can be summarized as follows.

- The overall error rate followed the trends previously observed for distant microphone recognition, when a significant degradation was observed only for CS3 channel. The results for CS2 channel were very similar to CS0 channel and only the 12 kbps achieved noticeably worse *WER*.

- The relative improvement of the studied AM refinement techniques had an increasing tendency with a decreasing bitrate. Also, it was noticeably higher for weakly distorted CS2 channel than for strongly distorted CS3 channel.

- The application of SSD has improved the error rates only for 16 kbps and 12 kbps bitrates. The sole application of ESS was found to be mostly ineffective in both cases of CS2 and CS3 channels. The combination of ESS and SSD achieved nearly identical results as sole SSD.

7.5 SSD for Advanced Audio Coding

Advanced Audio Coding is the successor to the MP3 that is defined in MPEG-4 part 3 (ISO/IEC 14496-3). It it a popular, often even default, formant for efficient Internet streaming in applications such as YouTube or Internet radios and podcasts. Perceptual tests have demonstrated that AAC achieves greater audio quality in comparison with MP3 for the same bitrate [78]. This improvement was achieved mainly by introducing a special pre-processing algorithm called Spectral Band Replication (SBR) which has been implemented to the standard AAC as the new high-efficiency AAC (HE-AAC) format. It is primary intended for speech applications at low bitrates. The principle idea of SBR is to replicate the LP filtered parts of speech using the middle and low frequency bands and to apply a frequency dependent shaping function to modulate the spectral envelope of replicated bands. HE-AAC v.1 profile has been developed for single channel audio and uses SBR to enhance the perceived quality. On the other hand, HE-AAC v.2 has been developed to improve the compression efficiency of stereo signals and uses SBR in conjunction with Parametric Stereo. The primary goal of this section was to compare the SSD algorithm with SBR even if both algorithms were designed for different purposes, SBR compensates bandlimiting and SSD spectral valleys. The analyses were performed with the same ASR system as the previous analyses with SSD. The FFmpeg tool was used for both compression and decompression. I worked only with HE-AAC v.1 as all my signal were recorded as single channel.

7.5.1 Results for AAC

Table 7.21 summarizes the results for AAC speech without the SBR and Table 7.22 summarizes the results for "high-efficiency" AAC which employed SBR. First interesting thing to note was that the achieved error rates are very similar to MP3 for all bitrates aside from 12 kbps (19.72% vs. 25.2%). In fact, the overall improvement of using AAC over MP3 was within 1% absolute for higher rates and the error rates started to rise after passing the 24 kbps threshold, although more slowly. These observations supported the assumption that the MP3 discussed degradations were also present in AAC speech. This conclusion was somewhat expected as both algorithms made use of a psychacoustic model and thus were expected to introduce the same artifacts. Second interesting thing was that SBR application improved the results only for the baseline and LDA level AMs and lowest bitrates. The final MPE trained AMs suffered from its application by about 5.36% on average. This findings was in direct contradiction with the conclusion made in cited works which were drawn from perceptual evaluations. It has to be noted however that the application of SBR is advised only for bitrates of 96 kbps and lower and thus the performance drop observed for 128 kbps bitrate could be explained by it. However, there is no other explanation for the performance drop for lower bitartes aside from the fact that SBR processed speech was not suitable for ASR. Finally, even the application of SSD worsened the results by 0.05% on average.

Table 7.21: Results for AAC speech and SSD compensation

			Adapted			
	Baseline	LDA	SAT	SGMM	MPE	MPE+SSD
128 kbps	27.58	26.93	21.95	15.66	**14.04**	14.12
64 kbps	27.15	26.98	21.63	15.76	**14.18**	14.25
32 kbps	27.66	27.66	21.76	16.06	14.38	**14.41**
28 kbps	28.38	27.88	21.83	16.44	**14.24**	14.27
24 kbps	28.35	28.15	22.13	16.58	**14.49**	14.56
20 kbps	29.60	28.67	22.21	17.17	15.03	**15.02**
16 kbps	32.81	32.43	25.21	19.47	**17.1**	17.28
12 kbps	38.84	36.33	26.70	22.44	19.72	**19.65**

Table 7.22: Results for high efficiency AAC speech

			Adapted		
Data	Baseline	LDA	SAT	SGMM	MPE
128 kbp	31.46	29.09	24.16	22.46	20.06
64 kbps	31.46	29.09	24.16	22.46	20.06
32 kbps	31.30	29.02	24.02	22.72	20.06
28 kbps	31.30	29.19	24.08	23.00	20.61
24 kbps	31.87	29.40	24.18	23.20	20.51
20 kbps	32.81	30.00	24.39	23.49	20.87
16 kbps	33.83	31.36	25.34	23.86	21.17
12 kbps	38.24	34.11	28.02	26.44	23.80

7.5.2 Summary

The main conclusions regarding the AAC can be summarized as follows.

- The AAC formant achieved very similar results to MP3 aside from the lowest 12 kbps rate, where AAC achieved much better error rates. However, the overall error rate followed the same trend as it started to rise exponentially after passing the 24 kbps threshold. Thus, we can conclude that both MP3 and AAC suffered from very similar problems.
- The application of both compensation methods, SBR and SSD, did not bring any significant improvement. The application of SBR was downright negative for the final MPE trained models, regardless of used bitrate. An actual improvement was observed only for *Baseline* and *LDA* level models for 16 kbps and 12 kbps rates. The application of SSD had practically no effect as the average improvement was only 0.05%.

CHAPTER 8

CONCLUSIONS

The goal of this thesis was to study robustness of ASR systems intended for strongly distorted speech, more precisely distant microphone, car and MP3 compressed speech. The thesis explored methods working at the level of signal preprocessing, feature extraction and AM refinement. The study was conducted in the following way.

- I studied the current state-of-the-art front-end processing and acoustic modelling techniques for robust speech recognition and established two experimental ASR frameworks based on the GMM-HMM and the DNN-HMM architectures. The clean acoustic conditions were used to train a reference AM and to the obtain reference results. These results were later used for comparison with the analysed adverse conditions and for the mismatched recognition.

- I analysed the real-life acoustic conditions of two public environments for distant speech recognition and three noisy car environments. The constructed ASR systems were used to evaluate the performance of the studied techniques with small-vocabulary and LVCSR tasks.

- I analysed the artifacts introduced by MP3 compression using the constructed ASR systems and evaluated the performance of the studied techniques. Also, the optimal setup was determined with small-vocabulary and LVCSR tasks for MP3 recordings.

- A novel compensation method was proposed and its contribution was evaluated with both architectures for Czech, English and German languages. Following experiments have proved that the proposed technique brings significant improvements for both architectures.

The most important conclusions can be summarized as follows. Regarding the distant and car speech recognition, the performed analyses were focused on two public environments and a noisy car environment with various microphone positions. The thesis analysed the combination of feature normalization and ESS techniques and evaluated their contribution for current GMM-HMM based systems. The performed experiments proved that ESS could bring significant contributions for middle and far distance microphones in situation with strong distortions. The highest *WERR* was observed for the AM adaptation and the combination of UBM and SGMM. The MPE criteria proved to have better generalization capabilities in the mismatched training and also proved to be more robust against additive and convolution noises in matched training.

Regarding the MP3, the study was focused on determining the artifacts introduced by the compression and analysing their effects on ASR. It was found that the low-pass filtering had only marginally negative effect while spectral valleys degraded the ASR performance more severely. It was also concluded that the total error rate was caused by a non-linear combination of both artifacts occurring at the same time. The analysis also showed that the spectral valleys could be detected in the spectral and log-spectral domain as a series of near zero values. The PLP features were found to be much more robust than MFCCs against the non-linear distortions introduced by the MP3. The application of AM adaptation was found to be most beneficial if the non-compressed speech AM was used for decoding the compressed speech. The experiments demonstrated that MP3 corrupted unvoiced speech units more severely than the voiced units.

The thesis proposed a novel compensation method called *Spectrally Selective Dithering* whose purpose was to compensate the effect of spectral valleys. The method was based on detecting the corrupted frequency bands in the inverse filtered segments of speech and adding a weighted amount of noise. The criteria function was based on the spectral flatness and the gain of added noise was estimated as a simple average from clean bands. The experiments proved that the proposed techniques could improve the performance for both GMM-HMM and DNN-HMM systems for Czech language. The *WERR* of SSD reached up to 7.3% for the GMM-HMM system and up to 7% for the DNN-HMM system for Czech. The experiments with English and German languages proved that the MP3 compression degraded other languages equally. The proposed SSD algorithm brought up to 15.3% and 10.2% *WERR* for the GMM-HMM and DNN-HMM systems respectively for these languages. Finally, the SSD technique was compared against a perceptually-motivated compensation technique called *Spectral band replication*, that was designed for the AAC encoder. The SSD brought only marginal contribution for AAC speech but the SBR technique was found to severely degrade the ASR performance.

The achieved results demonstrated that there is still a large room for improvements for MP3 and other perceptual coding schemes. The initial study on SBR implemented in the AAC format proved, that just a simple replication of low-pass filtered bands was not a solution for improving the ASR performance for perceptually coded speech. The future research should be aimed at explaining why this degradation occurred as this knowledge could be utilize for improving the performance of SSD even further. A more comprehensive study on newer audio coders and the already developed perception based compensation methods is certainly other interesting area for future research.

BIBLIOGRAPHY

[1] G. Hinton, L. Deng, D. Yu, G. Dahl, A. rahman Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition," *Signal Processing Magazine*, 2012.

[2] Y. Hoshen, R. Weiss, and K. W. Wilson, "Speech acoustic modeling from raw multichannel waveforms," in *International Conference on Acoustics, Speech, and Signal Processing*, (South Brisbane, Australia), 2015.

[3] T. N. Sainath, R. J. Weiss, A. Senior, K. W. Wilson, and O. Vinyals, "Learning the speech front-end with raw waveform CLDNNs," in *Proc. of Interspeech*, (Dresden, Germany), 2015.

[4] D. Palaz, M. Magimai.-Doss, and R. Collobert, "Analysis of CNN-based speech recognition system using raw speech as input," in *Proc. of Interspeech*, (Dresden, Germany), 2015.

[5] J. L. Flanagan, "Note on the design of "terminal-analog" speech synthesizers," *The Journal of the Acoustical Society of America*, vol. 29, pp. 306–310, 1957.

[6] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, pp. 357–366, Aug 1980.

[7] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *The Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738–1752, 1990.

[8] A. R. Mohamed, G. Hinton, and G. Penn, "Understanding how Deep Belief Networks perform acoustic modelling," in *International Conference on Acoustics, Speech, and Signal Processing*, (Kyoto, USA), pp. 4273–4276, March 2012.

[9] J. Li, D. Yu, J. T. Huang, and Y. Gong, "Improving wideband speech recognition using mixed-bandwidth training data in CD-DNN-HMM," in *Spoken Language Technology Workshop (SLT), 2012 IEEE*, (Miami, USA), pp. 131–136, Dec 2012.

[10] S. J. Jacewicz E, Fox RA, "Vowel duration in three American English dialects," *American speech*, vol. 82, no. 4, pp. 367–385, 2007.

[11] N. Zheng, X. Li, H. Cao, T. Lee, and P. C. Ching, "Deriving MFCC parameters from the dynamic spectrum for robust speech recognition," in *Chinese Spoken Language Processing, 2008. ISCSLP '08. 6th International Symposium on*, (Kunming, China), pp. 1–4, Dec 2008.

[12] R. Haeb-Umbach and H. Ney, "Linear discriminant analysis for improved large vocabulary continuous speech recognition," in *International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, (San Francisco, USA), pp. 13–16, Mar 1992.

[13] N. Kumar, *Investigation of Silicon Auditory Models and Generalization of Linear Discriminant Analysis for Improved Speech Recognitiom*. PhD thesis, Johns Hopkins University, 1997.

[14] N. Singh-Miller, *Neighborhood Analysis Methods in Acoustic Modeling for Automatic Speech Recognition*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2010.

[15] D. Kolossa, S. Zeiler, R. Saeidi, and R. F. Astudillo, "Noise-adaptive LDA: A new approach for speech recognition under observation uncertainty," *IEEE Signal Processing Letters*, vol. 20, pp. 1018–1021, Nov 2013.

[16] O. Siohan, "On the robustness of linear discriminant analysis as a preprocessing step for noisy speech recognition," in *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, vol. 1, (Detroit, USA), pp. 125–128 vol.1, May 1995.

[17] H. Abbasian, B. Nasersharif, A. Akbari, M. Rahmani, and M. Moin, "Optimized linear discriminant analysis for extracting robust speech features," in *Communications, Control and Signal Processing. 3rd International Symposium on*, (Malta), pp. 819–824, March 2008.

[18] A. Nadas, "A decision theoretic formulation of a training problem in speech recognition and a comparison of training by unconditional versus conditional maximum likelihood," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 31, pp. 814–817, 1983.

[19] J. Psutka, L. Müller, J. Matoušek, and V. Radová, *Mluvíme s počítačem česky*. Prague: Academia, 2006.

[20] P. S. Gopalakrishnan, D. Kanevsky, A. Nádas, and D. Nahamoo, "An inequality for rational functions with applications to some statistical estimation problems.," *IEEE Transactions on Information Theory*, vol. 37, pp. 107–113, 1991.

[21] R. Cardin, Y. Normandin, and R. de Mori, "High performance connected digit recognition using maximum mutual information estimation," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, (Toronto, Canada), pp. 533–536, 1991.

[22] V. Valtchev, J. J. Odell, P. Woodland, and S. Young, "MMIE training of large vocabulary recognition systems," *Speech Comminication*, vol. 22, pp. 303–314, 1997.

[23] X. He, L. Deng, and W. Chou, "Discriminative learning in sequential pattern recognition," *IEEE Signal Processing Magazine*, vol. 25, no. 5, pp. 14–36, 2008.

[24] H. Jiang, "Discriminative training of HMMs for automatic speech recognition: A survey," *Computer Speech & Language*, vol. 24, no. 4, pp. 589 – 608, 2010.

[25] L. Bahl, P. Brown, P. de Souza, and R. Mercer, "Maximum mutual information estimation of Hidden Markov model parameters for speech recognition," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, vol. 11, (Tokyo, Japan), pp. 49–52, 1986.

[26] A. Nadas, D. Nahamoo, and M. Picheny, "On a model-robust training method for speech recognition," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 36, pp. 1432–1436, 1988.

[27] P. Woodland and D. Povey, "Large scale discriminative training of Hidden Markov models for speech recognition," *Computer Speech & Language*, vol. 16, no. 1, pp. 25–47, 2002.

[28] D. Povey and P. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, (Orlando, USA), pp. I–105–I–108, 2002.

[29] D. Povey, D. Kanevsky, B. Kingsbury, B. Ramabhadran, G. Saon, and K. Visweswariah, "Boosted MMI for model and feature-space discriminative training.," in *International Conference on Acoustics, Speech, and Signal Processing*, (Las Vegas, USA), pp. 4057–4060, IEEE, 2008.

[30] L. D. Dong Yu, *Automatic Speech Recognition: A Deep Learning Approach*. Verlag London: Springer, 2015.

[31] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An overview of Noise-robust Automatic Speech Recognition," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 22, pp. 745–777, Apr. 2014.

[32] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, pp. 1109–1121, Dec 1984.

[33] G. S. Kang and L. J. Fransen, "Quality improvement of LPC-processed noisy speech by using spectral subtraction," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, pp. 939–942, Jun 1989.

[34] G. Doblinger, "Computationally efficient speech enhancement by spectral minima tracking in subbands," in *In Proceedings of 4th European Conference on Speech Technology and Communications*, (Madrin, Spain), pp. 1513–1516, 1995.

[35] C. Kermorvant, "A comparison of noise reduction techniques for robust speech recognition," Tech. Rep. Idiap-RR-10-1999, IDIAP, 1999.

[36] P. Sovka, P. Pollak, and J. Kybic, "Extended spectral subtraction," in *European Signal Processing Conference (EUSIPCO–96)*, (Trieste, Italy), pp. 963–966, 1996.

[37] T. Fux and D. Jouvet, "Evaluation of PNCC and extended spectral subtraction methods for robust speech recognition," in *Signal Processing Conference (EUSIPCO), 2015 23rd European*, (Nice, France), pp. 1416–1420, Aug 2015.

[38] A. Pawar, K. Choudhari, and M. Josh, "Review of single channel speech enhancement methods in spectral domain," *Internation Journal of Applied Engineering, Research (IIAER)*, vol. 7, no. 11, pp. 1961–1966, 2012.

[39] Z. Zajíc, *Adaptace akustického modelu v úloze s malým množstvím adaptačních dat.* PhD thesis, Technická univerzita v Liberci, 2012.

[40] J. Nouza and J. Silovsky, "*Fast Keyword Spotting in Telephone Speech*," *Radioengineering*, vol. Vol. 18, no. 4, pp. 665–670, 2009.

[41] B. Schuller, M. Wöllmer, T. Moosmayr, and G. Rigoll, "Speech recognition in noisy environments using a switching linear dynamic model for feature enhancement," in *Proc. of Interspeech*, (Brisbane, Australia), pp. 1789–1792, 2008.

[42] O. Kalinli, M. Seltzer, J. Droppo, and A. Acero, "Noise adaptive training for robust automatic speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 8, pp. 1889–1901, 2010.

[43] D. P. et al., "The subspace Gaussian mixture model— A structured model for speech recognition," *Computer Speech & Language*, vol. 25, no. 2, pp. 404 – 439, 2011.

[44] C.-H. Lee and J.-L. Gauvain, "Speaker adaptation based on MAP estimation of HMM parameters," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, (Hong Kong), pp. 558–561, IEEE Computer Society, 1993.

[45] J. Gauvain and C.-H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 2, pp. 291–298, 1994.

[46] A. Sankar, A. Sankar, C. hui Lee, and C. hui Lee, "A maximum-likelihood approach to stochastic matching for robust speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 4, pp. 190–202, 1996.

[47] C. Leggetter and P. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density Hidden Markov models," *Computer Speech & Language*, vol. 9, no. 2, pp. 171 – 185, 1995.

[48] M. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech & Language*, vol. 12, pp. 75–98, 1998.

[49] J. Rajnoha and P. Pollák, "ASR systems in noisy environment: Analysis and Solutions for Increasing Noise Robustness," *Radioengineering*, vol. 20, no. 1, pp. 74–84, 2011.

[50] S. Tamura and S. Hayamizu, "Multi-stream acoustic model adaptation for noisy speech recognition," in *Signal Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012 Asia-Pacific*, (Hollywood, USA), pp. 1–4, Dec 2012.

[51] Y. Pan and A. Waibel, "Experiments on distant-talking speech recognition in meeting room using extended MAM," in *International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, (Orlando, USA), pp. IV–4165–IV–4165, May 2002.

[52] X. Cui and A. Alwan, "Noise robust speech recognition using feature compensation based on polynomial regression of utterance SNR," *IEEE Transactions on Speech and Audio Processing*, vol. 13, pp. 1161–1172, Nov 2005.

[53] D. K. Kim and M. J. F. Gales, "Noisy constrained maximum-likelihood linear regression for noise-robust speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, pp. 315–325, Feb 2011.

[54] J. Lu, J. Ming, and R. Woods, "Adapting noisy speech models - Extended uncertainty decoding," in *International Conference on Acoustics, Speech, and Signal Processing*, (Dallas, USA), pp. 4322–4325, March 2010.

[55] A. Ben-Yishai and D. Burshtein, "A discriminative training algorithm for hidden Markov models," *IEEE Transactions on Speech and Audio Processing*, vol. 12, pp. 204–217, May 2004.

[56] J. Droppo and A. Acero, "Maximum mutual information SPLICE transform for seen and unseen conditions," in *Proc. of Interspeech*, (Lisboa, Portugal), International Speech Communication Association, September 2005.

[57] J. Du, P. Liu, F. Soong, J.-L. Zhou, and R.-H. Wang, *Chinese Spoken Language Processing: 5th International Symposium, ISCSLP 2006, Singapore, December 13-16, 2006. Proceedings*, ch. Noisy Speech Recognition Performance of Discriminative HMMs, pp. 358–369. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006.

[58] Y. Tachioka, S. Watanabe, J. L. Roux, and J. R. Hershey, "Discriminative methods for noise robust speech recognition: A CHIME challenge benchmark," in *The 2nd CHiME workshop on Machine Listening in Mutlisource Environments*, (Vancouver, Canada), pp. 1–6, 2013.

[59] A. Spanias, T. Painter, and V. Atti, *Audio Signal Processing and Coding*. Hoboken: A John Wiley & Sons, Inc., 2007.

[60] D. Pan, "A tutorial on MPEG/audio compression," *MultiMedia, IEEE*, vol. 2, no. 2, pp. 60–74, 1995.

[61] S. Young and et al., *The HTK Book, Version 3.4.1*. Cambridge, 2009.

[62] D. e. a. Povey, "The Kaldi Speech Recognition Toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, (Hilton Waikoloa Village, USA), IEEE Signal Processing Society, 2011.

[63] Petr Fousek and Petr Mizera and Petr Pollak, "CtuCopy feature extraction tool.." Available at: http://noel.feld.cvut.cz/speechlab - download section, 2014.

[64] "The LAME Project." Available at: http://lame.sourceforge.net.

[65] "Sox - sound eXchange." Available at: http://sox.sourceforge.net.

[66] "FFmpeg." Available at: https://ffmpeg.org.

[67] "Praat." Available at: http://www.fon.hum.uva.nl/praat.

[68] P. Pollak and J. Cernocky, "Czech SPEECON Adult Database," tech. rep., Czech Technical University in Prague, Nov 2003. http://www.speechdat.org/speecon.

[69] V. Prochazka, P. Pollak, J. Zdansky, and J. Nouza, "Performance of Czech speech recognition with language models created from public resources," *Radioengineering*, vol. 20, pp. 1002–1008, 2011.

[70] "Institute of Czech National Corpus - SYN2006PUB," 2006. http://ucnk.ff.cuni.cz/english/syn2006pub.php.

[71] J. Nouza, D. Nejedlová, J. Zdánský, and J. Kolorenc, "Very large vocabulary speech recognition system for automatic transcription of Czech broadcast programs," in *Proc. of Interspeech*, (Jeju Island, Korea), 2004.

[72] J. Trmal, A. Pražák, Z. Loose, and J. Psutka, "Online TV captioning of Czech parliamentary sessions," in *Text, Speech and Dialogue*, vol. 6231 of *Lecture Notes in Computer Science*, (Springer Berlin / Heidelberg), pp. 416–422, 2010.

[73] P. Ircing, J. Psutka, and V. Radová, "Automatic transcription of audio archives for spoken document retrieval," in *Proceedings of the second IASTED international conference on Computational intelligence*, (Anaheim), pp. 448–452, ACTA Press, 2006.

[74] Z. Palková, *Fonetika a Fonologie češtiny*. Praha: Karolinum, 1994.

[75] R. Schwartz, Y. Chow, S. Roucos, M. Krasner, and J. Makhoul, "Improved hidden Markov modeling of phonemes for continuous speech recognition," in *International Conference on Acoustics, Speech, and Signal Processing*, vol. 9, (San Diego, USA), pp. 21–24, Mar 1984.

[76] M. R. Schroeder, B. S. Atal, and J. L. Hall, "Optimizing digital speech coders by exploiting masking properties of the human ear," *The Journal of the Acoustical Society of America*, vol. 66, no. 6, 1979.

[77] C.-M. Liu, H.-W. Hsu, and W.-C. Lee, "Compression artifacts in perceptual audio coding," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 4, pp. 681–695, 2008.

[78] K. Brandenburg, "MP3 and AAC explained," in *AES 17th International Conference on High Quality Audio Coding*, 1999.

[79] AES, "AES technical committee on coding of audio signals, perceptual audio coders : What to listen for, CR-ROM," 2002.

[80] ITU, "ITU-R recommendation bs. 1116, Methods for subjective assessment of small impairments in audio systems including multichannel sound systems," 1994.

[81] EBU, "EBU project group B/AIM, EBU subjective listening tests on low-bitrate audio codecs„" 2003.

[82] R. H. van Son, "A study of pitch, formant, and spectral estimation errors introduced by three lossy speech compression algorithms," in *Acta Acustica united with Acustica*, vol. 91, pp. 771–778, 2005.

[83] C. Barras, L. Lamel, and J. Gauvain, "Automatic transcription of compressed broadcast audio," in *International Conference on Acoustics, Speech, and Signal Processing*, (Salt Lake City, USA), pp. 265–268, 2001.

[84] L. Besacier, C. Bergamini, D. Vaufreydaz, and E. Castelli, "The effect of speech and audio compression on speech recognition performance," in *Proceedings of 2001 IEEE Fourth Workshop on Multimedia Signal Processing*, (Cannes, France), pp. 301–306, 2001.

[85] P. S. Ng and I. Sanches, "The influence of audio compression on speech recognition systems," in *SPECOM 2004 - Proceedings of Conference Speech and Computer*, (St. Petersburg, Russia), 2004.

[86] P. Pollak and M. Behunek, "Accuracy of MP3 speech recognition under real-word conditions - Experimental study," in *SIGMAP 2011 - Proceedings of the International Conference on Signal Processing and Multimedia Applications*, (Seville, Spain), pp. 5–10, 2011.

[87] J. Nouza, P. Cerva, and J. Silovsky, "Adding controlled amount of noise to improve recognition of compressed and spectrally distorted speech," in *International Conference on Acoustics, Speech, and Signal Processing*, (Vancouver, Canada), pp. 8046–8050, 2013.

[88] L. Seps, J. Malek, P. Cerva, and J. Nouza, "Investigation of deep neural networks for robust recognition of nonlinearly distorted speech," in *Proc. of Interspeech*, (Singapore), pp. 363–367, 2014.

[89] B. D'Alessandro and Y. Q. Shi, "MP3 bit rate quality detection through frequency spectrum analysis," in *Proceedings of the 11th ACM Workshop on Multimedia and Security*, (Buffalo, USA), pp. 57–62, 2009.

[90] M. Qiao, A. Sung, and Q. Liu, "Improved detection of MP3 double compression using content-independent features," in *Signal Processing, Communication and Computing (ICSPCC), 2013 IEEE International Conference on*, (Kunming, China), pp. 1–4, 2013.

[91] R. Yang, Y.-Q. Shi, and J. Huang, "Defeating fake-quality MP3," in *Proceedings of the 11th ACM Workshop on Multimedia and Security*, MMSEC '09, (Princeton, USA), pp. 117–124, ACM, 2009.

[92] C. M. Liu, W. C. Lee, and H. W. Hsu, "High frequency reconstruction by linear extrapolation," in *Proc. of 115th Convention of Audio Engineering Society Convention*, (New York, USA), 2003.

[93] H. W. Hsu, C. M. Liu, and W. C. Lee, "Audio patch method in audio decoders - MP3 and AAC," in *Proc. of 116th Convention of Audio Engineering Society Convention*, (Berlin, Germany), 2004.

[94] M. Dietz and et. al., "Spectral band replication, a novel approach in audio coding," in *Proc. of 120th Convention of Audio Engineering Society Convention*, (Paris, France), 2006.

[95] M. Arora, J. Lee, and S. Park, "High quality blind bandwidth extension of audio for portable player applications," in *Proc. of 120th Convention of Audio Engineering Society Convention*, (Paris, France), 2006.

[96] "Coding technologies. mp3pro."

[97] R. Böhme and A. Westfeld, "Statistical characterisation of MP3 encoders for steganalysis," in *Proceedings of the 2004 Workshop on Multimedia and Security*, MMSEC '04, (Magdeburg, Germany), pp. 25–34, ACM, 2004.

[98] S. G. S. Pettersen, *Robust Speech Recognition in the Presence of Additive Noise*. PhD thesis, Norwegian University of Science and Technology, 2008.

[99] D. B. Paul and J. M. Baker, "The design for the Wall Street Journal-based CSR corpus," in *DARPA Speech and Language Workshop*, Morgan Kaufmann Publishers, 1992.

[100] T. Schultz, N. Vu, and T. Schlippe, "GlobalPhone: A multilingual text & speech database in 20 languages," in *International Conference on Acoustics, Speech, and Signal Processing*, (Vancouver, Canada), pp. 8126–8130, 2013.

[101] T. Schultz, "Rapid language adaptation tools for multilingual speech processing," in *Automatic Speech Recognition Understanding, 2009. ASRU 2009. IEEE Workshop on*, (Merano/Meran, Italy), pp. 51–51, 2009. Available at: http://csl.uni-bremen.de/GlobalPhone/.

PUBLICATIONS

Publications related to the Thesis

Impact Journals

Borský, M. - Pollák, P. - Mizera, P.: Advanced Acoustic Modelling Techniques in MP3 Speech Recognition. EURASIP Journal on Audio Speech and Music Processing. 2015, vol. 2015:20, ISSN 1687-4722. (Authorship Credit 1/3)

Borský, M. - Pollák, P. - Mizera, P. - Nouza, J.: Dithering Techniques in Automatic Recognition of Speech Corrupted by MP3 Compression: Analysis, Solutions and Experiments. Speech Communication *Under review.* (Authorship Credit 1/4)

Peer-reviewed Journals

Pollák, P. - **Borský, M.**: Small and Large Vocabulary Speech Recognition of MP3 Data under Real-Word Conditions: Experimental Study. Communications in Computer and Information Science. 2012, vol. 314, p. 409-419. ISSN 1865-0929. (Authorship Credit 1/2)

ISI Conferences

Borský, M. - Mizera, P. - Pollák, P. : Spectrally Selective Dithering for Distorted Speech Recognition. In Proceeding of Interspeech 2015. (Authorship Credit 1/3)

Borský, M. - Mizera, P. - Pollák, P.: Noise and Channel Normalized Cepstral Features for Far-Speech Recognition. In Speech and Computer. Cham: Springer International Publishing AG, 2013, p. 241-248. ISSN 0302-9743. ISBN 978-3-319-01930-7. (Authorship Credit 1/3)

Borský, M. - Pollák, P.: The optimization of PLP feature extraction for LVCSR recognition of MP3 data. In 19th International Conference on Applied Electronics 2014. Pilsen: University of West Bohemia, 2014, p. 55-58. ISSN 1803-7232. ISBN 978-80-261-0276-2. (Authorship Credit 1/2)

Borský, M. - Pollák, P.: Knowledge-Based and Automated Clustering in MLLR Adaptation of Acoustic Models for LVCSR. In 2012 International Conference on Applied Electronics. Pilsen: University of West Bohemia, 2012, p. 33-36. ISSN 1803-7232. ISBN 978-80-261-0038-6. (Authorship Credit 1/2)

Borský, M. - Pollák, P.: Optimized State-Tying for Triphone-Based HMMs under Training Data Deficiency. In Applied Electronics - 2013 International Conference on Applied Electronics. Pilsen: University of West Bohemia, 2013, p. 45-48. ISSN 1803-7232. ISBN 978-80-261-0166-6. (Authorship Credit 1/2)

Book Chapter

Borský, M. - Pollák, P.: Analysis and automatic recognition of compressed speech. In Tackling the Complexity in Speech. Praha: Filozofická fakulta Univerzity Karlovy v Praze, 2015, p. 205-221. ISBN 978-80-7308-558-2. (Authorship Credit 1/2)

Other Publications

Borský, M. - Pollák, P.: Recognition of Spectrally Distorted Speech after MP3 Compression. In 22nd Czech-German Workshop on Speech Communication. Book of Abstracts. 2014, p. 3-4. (Authorship Credit 1/2)

Borský, M. - Pollák, P.: Various Approaches of Small Vocabulary Speech Recognizer Implementation Using HTK Toolkit. In POSTER 2013 - 17th International Student Conference on Electrical Engineering. Prague: Czech Technical University, 2013, ISBN 978-80-01-05242-6. (Authorship Credit 1/2)

Borský, M.: Experimental Setup of LVCSR System Based on HTK Tools for Evaluation and Development Purposes. In POSTER 2012 - 16th International Student Conference on Electrical Engineering. Praha: Czech Technical University in Prague, 2012, p. 1-4. ISBN 978-80-01-05043-9.

Borský, M.: The recognition of MP3 compressed speech using HMM-based system. In IV. Letní doktorandské dny 2014. Prague: CTU, Faculty of Electrical Engineering, Department of Circuit Theory, 2014, díl 4, s. 77-80. ISBN 978-80-01-05506-9. (in Slovak).

Borský, M.: HMM based acoustic modelling of triphones. In III. LETNÍ DOKTORANDSKÉ DNY 2013. Prague: CTU, Faculty of Electrical Engineering, 2013, díl 3, s. 86-91. ISBN 978-80-01-05251-8. (in Czech).

Borský, M.: The use of MLLR acoustic model adaptation of triphones for continuous speech recognition system based on HTK. In LETNÍ DOKTORANDSKÉ DNY 2012. Prague: CTU, 2012, s. 68-74. ISBN 978-80-01-05050-7. (in Slovak).

Pollák, P. - **Borský, M**. - Mizera, P.: Creation of HMM Acoustic Models for Czech. [Výzkumná zpráva]. Prague: CTU, Faculty of Electrical Engineering, Department of Circuit Theory, 2013. FZ-2013-1. 4 s. (in Czech). (Authorship Credit 1/3)

Pollák, P. - Mizera, P. - **Borský, M.**: Reconstruction of Speech from its Mel-Cepstrum I: Basic Implementation. [Výzkumná zpráva]. Prague: CTU, Faculty of Electrical Engineering, Department of Circuit Theory, 2013. FZ-2013-2. 4 s. (in Czech). (Authorship Credit 1/3)

Pollák, P. - Mizera, P. - **Borský, M.**: Reconstruction of Speech from its Mel-Cepstrum II: Optimized Implementation. [Výzkumná zpráva]. Prague: CTU, Faculty of Electrical Engineering, Department of Circuit Theory, 2014. FZ-2014-1. 4 s. (in Czech). (Authorship Credit 1/3)

Publications unrelated to the Thesis

Borský, M. - Mehta, D.D. - Gudjohnsen, J.P. - Gudnason, J. : Classification of Voice Modality using Electroglottogram Waveforms. In Proceeding of Interspeech 2016. *Accepted for publication.* (Authorship Credit 1/4)

APPENDIX

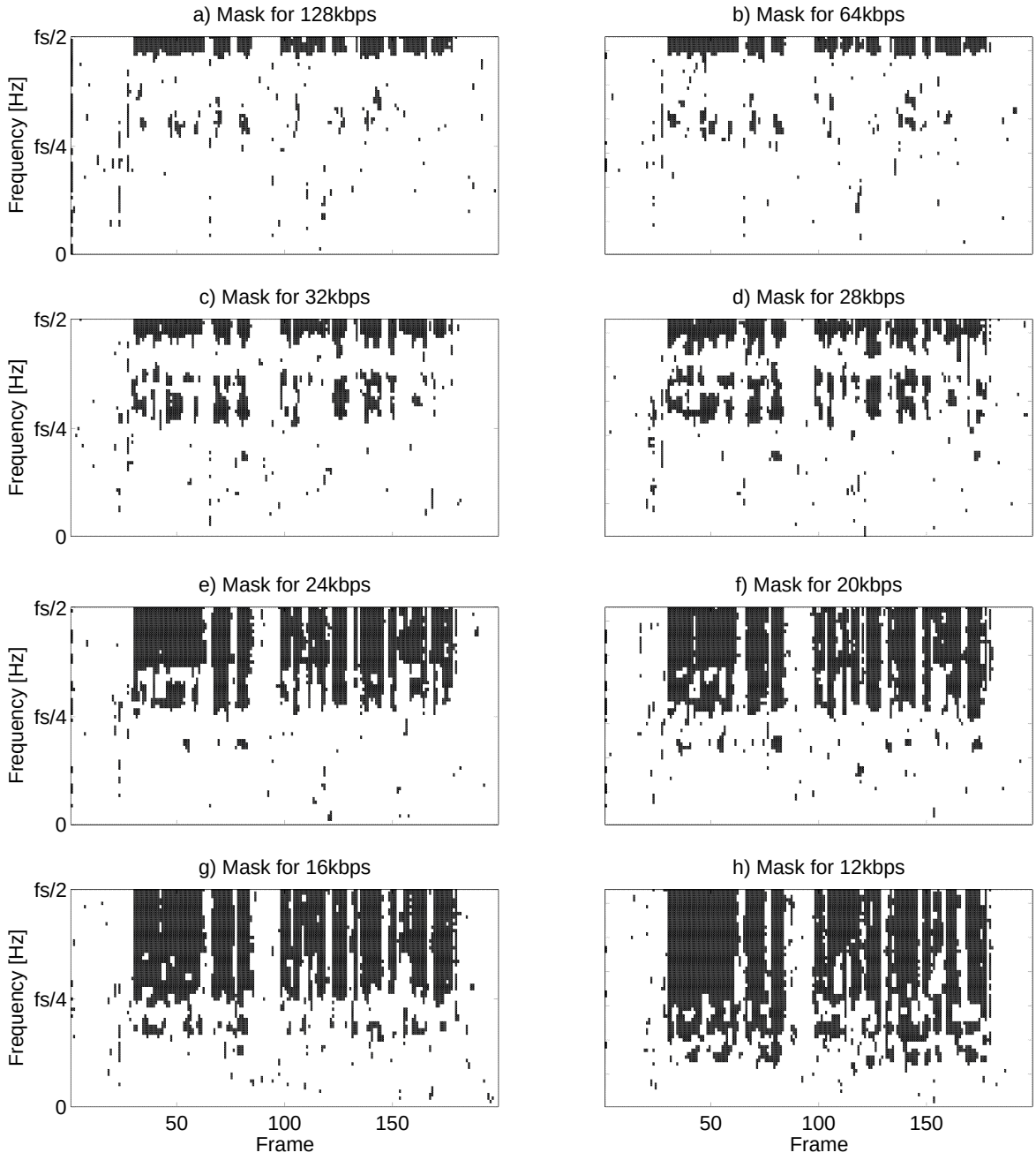

Figure 8.1: Illustrative masks estimated by the Zero-band detector block for a single signal containing a whole sentence.

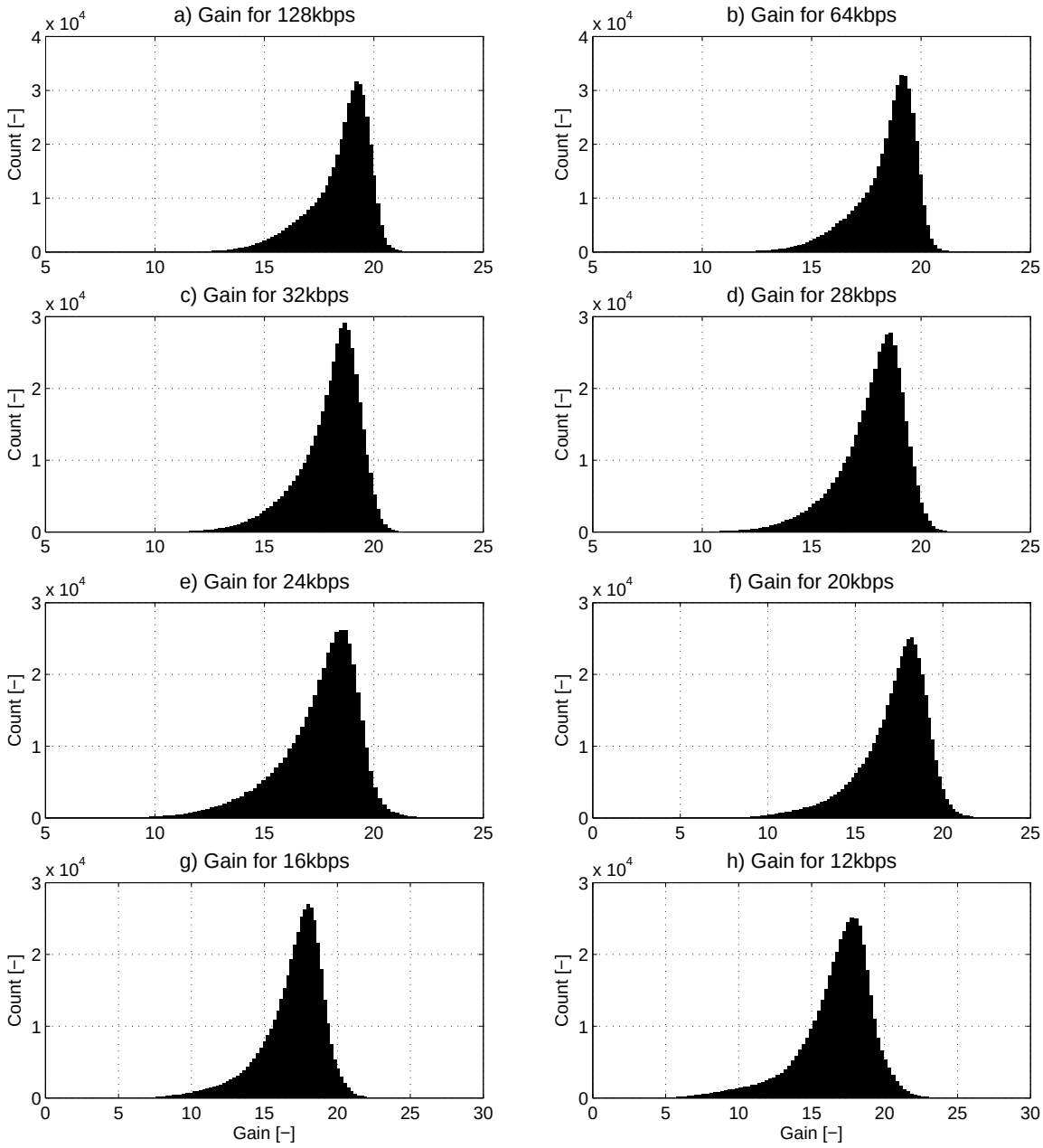

Figure 8.2: Histograms of the estimated Gain by the Gain-estimation block.

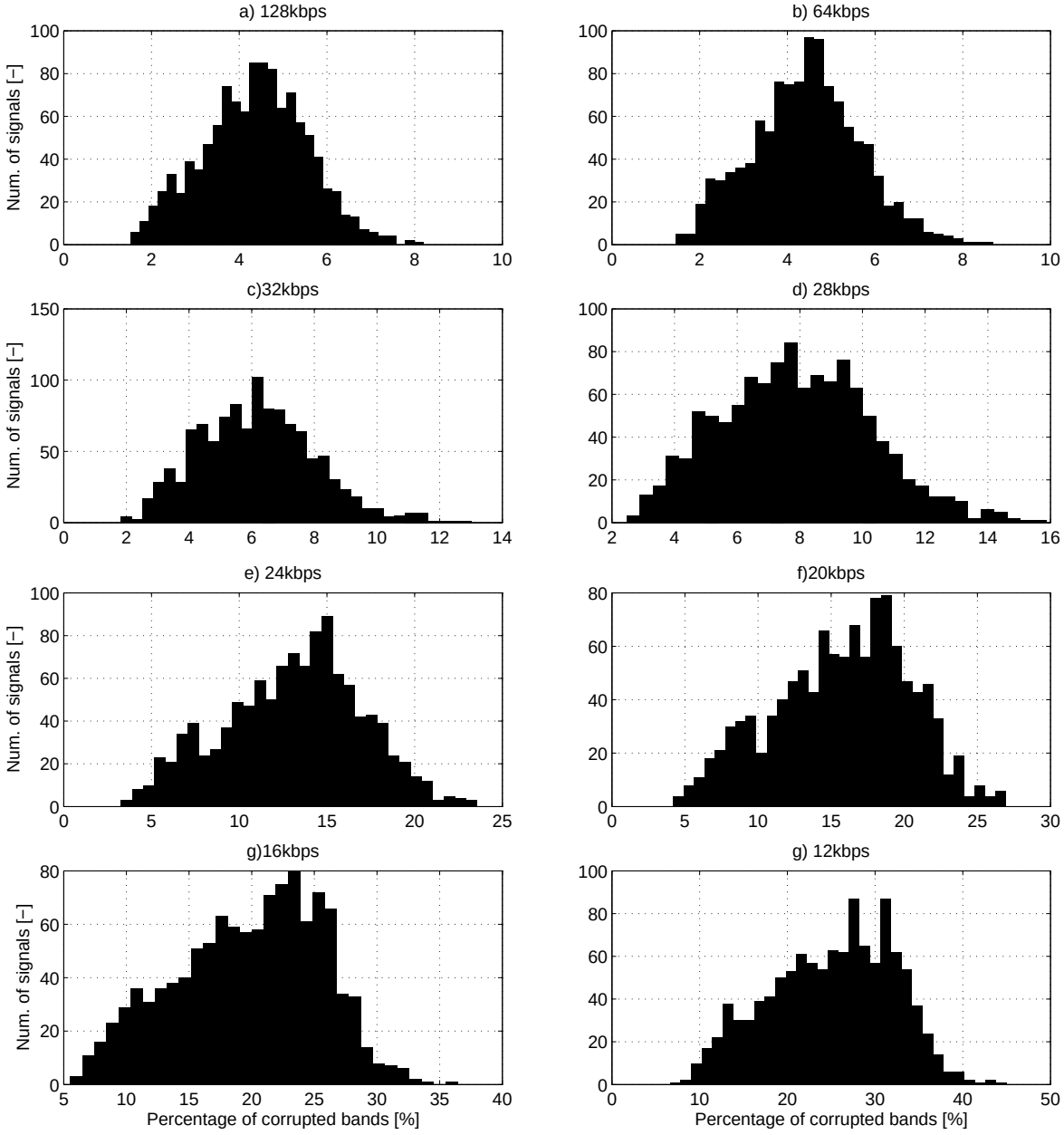

Figure 8.3: Percentage of corrupted bands in a signal.