Project no: FP6 034362

ACORNS
Acquistion of Communication and RecogNition Skills

Instrument: STREP

Thematic Priority: Information Society Technologies

D1.3: Final modules for features derived from auditory model and a self-learning algorithm

Due date of deliverable: 2009-11-26

Actual submission date: 2009-11-26

Start date of project: 2006-07-01 Duration: 36 Months

Organisation name of lead contractor for this deliverable: KTH

Revision: 1.4

Project co-funded by the European Commission within the Sixth Framework Programme 2002-2006		
Dissemination Level		
PU	Public	**X**
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Table of Contents

Abstract

This report describes the progress within the ACORNS project towards developing and improving features based on knowledge of the human auditory system, and designing phone-class specific features for a self-learning language acquisition algorithm. Our work led to an off-line optimization algorithm that helps to design features based on the ability of the features to describe the components of speech that are most clearly perceived. Experimental results confirm effectiveness of this generic strategy. Also a self-learning algorithm is presented that uses a bottom-up approach to automatically discover, acquire and recognize words of a language.

Chapter 1

Overview

This report is the description component of deliverable D1.3 of the ACORNS project. As listed in the Annex, the deliverable was aimed to consist of "Final modules for features derived with sensitivity analysis method criterion, with quantitative evaluation". With the objective of designing efficient and computationally inexpensive features without explicit feedback from the learning algorithms or speech recognizers, this report describes mainly two parts: (a) Deriving sophisticated auditory model based algorithms for designing improved features through off-line optimization based approach, and (b) Un-supervised learning of time-frequency patches as a noise-robust representation of speech. Part (a) of the deliverable is associated with Task 1 ("distortion-based approach") and Part (b) is associated with Task 2 ("phone-class specific features").

Task 1 deals with developing efficient and computationally inexpensive features that can be subsequently used by ACORNS algorithms for agent learning. Consistent with the ACRONS philosophy, the features are developed without feedback on word classification. The hypothesis is that auditory model based design leads to better features that are sufficiently generic to be used by any pattern recognition/learning method. As ACORNS developed new recognition methods, it provides a natural platform to test the usefulness of new features. Task 1 consists of two sub-tasks. The first sub-task deals with developing an algorithm to select an efficient subset of features from a larger set of existing features. The feature selection is performed based on the ability of the subset to describe the audible components of the signal. The success of the first sub-task forms the basis of further investigation to design improved feature sets in the second sub-task. The second sub-task deals with designing an efficient and computationally inexpensive feature set using advanced auditory models (both static spectral and dynamic spectro-temporal auditory models). Based on the existence of a set of adjustable parameters in the feature definition, this work has led to a general framework for optimizing the adjustable parameters such that the feature set emulates the behaviour of the human auditory system. To design an improved feature set, we address both the issues of static and dynamic auditory perception. The static perception is related to spectral masking as observed in a particular speech frame and the dynamic auditory perception is related to the behaviour of human auditory response across speech frames in the sense of spectro-temporal masking. The work in the second sub-task has led to the design of new features that were shown to provide better speech recognition performance at any environmental condition and subsequently used by ACORNS learning algorithms.

On the other hand, Task 2 deals with a self-learning algorithm that uses a bottom-up approach to automatically discover, acquire and recognize the words of a language by evaluating phone-class specific features. In Task 2, a special type of phone-class specific spectro-temporal feature is used such that long-term temporal behaviour of speech can be handled. The use of spectro-temporal infor-

mation in Task 2 naturally provides a testing platform where the features developed from Task 1 can be used for further investigation.

In this report, the usefullness of auditory model based features is shown through the standard HMM based automatic speech recognition (ASR) experimental setup for phone and word recognition tasks. The performance of new features for ACORNS learning algorithms is reported in deliverable D5.3.

This report part of the deliverable consists of four chapters, each describing one of the forementioned topics. Chapter 3 is based on a paper published in Interspeech'09. Chapter 2 and chapter 4 are based on the papers that are currently under review. The work in chapter 5 is based on a paper published in Speech Communication.

Chapter 2, chapter 3 and 4 are concerned with Task 1. Conceptually, in Task 1, we should develop a feature that is most sensitive to the human auditory system in the sense of perceptual distortion. However, there exists no theory which conclusively describes that non-perceivable features of speech signal do not contain any useful information for speech recognition in the central auditory system. Also, in reality, a technical pattern recognizer/learning algorithm works differently than our central auditory system/brain does, and it is questionable whether the technical recognizer can use the auditory motivated features efficiently. Even though some usefulness of auditory motivated features were earlier shown through isolated sound/word recognition tasks (which can be considered as primitive experiments today), no concerted effort has been carried out to conclusively show their usefulness for speech recognition and learning tasks as the human being does. With these caveats in place, Task 1 deals with developing systematic approach to design auditory motivated features which are shown to provide improved performance for contemporary ASR tasks and ACORNS learning algorithms.

Chapter 2 describes our algorithm to find a good subset of features for recognition from a larger set, using only knowledge of the human auditory system as a measure. The underlying assumption of our work is that the human auditory system is effective at extracting relevant information from the speech signal. We use a dynamic spectro-temporal auditory model to perform a sensitivity analysis on speech, based on a distortion measure. The method eliminates the dependency of the feature set to the speech recognition system used, and results in a generic set of good features. We evaluated the selected feature subsets on a real speech recognizer. The results confirm that knowledge of the human auditory system forms a good basis for selecting a subset of features from a larger set for the purpose of speech recognition.

While the feature selection method of Chapter 2 can be used within ACORNS to limit the number of features, this is not a major concern for the project. Rather, the method should be seen as a first step towards a method that uses auditory-knowledge to improve existing features and define new features. Buoyed by the success of our approach, Chapter 3 and chapter 4 deal with the issue of designing better features than existing features. In Chapter 3, we use a static spectral auditory model to perform a sensitivity analysis on speech, based on a distortion measure. For a specific feature design example, we generalize the evaluation procedure of existing MFCC features through incorporating some adjustable parameters and then optimize the parameters using the perturbation based algorithm. We evaluated the new features on a real speech recognizer (such as HTK) and observed considerable improvement over existing MFCC features. The results confirm that knowledge of the human auditory system forms a good basis for designing and optimizing a set of features that is efficient for speech recognition. The new features reflect the innate perception of a baby (or human in general), and are not based on the feedback from some automatic speech recognition system.

For further improvement, chapter 4 explores the use of a dynamic auditory model to design a dynamic feature set. The use of a dynamic spectro-temporal auditory model leads to the incorporation

of human auditory behaviour across speech frames. This dynamic feature set is found to assist the features developed in Chapter 3 and we show that a significant improvement in automatic speech recognition (ASR) performance is obtained for any environmental condition, clean as well as noisy.

For Task 2, chapter 5 presents a self-learning algorithm using a bottom-up approach to automatically discover, acquire and recognize the words of a language. The magnitude spectrum of a special form of conventional short time Fourier transform (STFT) is used as the input to the algorithm. First, an un-supervised technique using non-negative matrix factorization (NMF) discovers phone-sized time-frequency patches into which speech can be decomposed. These patches can be considered to be speech *features*. The input matrix for the NMF is constructed for static and dynamic speech features using a spectral representation of both short and long acoustic events. By describing speech in terms of the discovered time-frequency patches, patch activations are obtained which express to what extent each patch is present across time. We then show that speaker-independent patterns appear to recur in these patch activations and how they can be discovered by applying a second NMF-based algorithm on the co-occurrence counts of activation events. By providing information about the word identity to the learning algorithm, the retrieved patterns can be associated with meaningful objects of the language. In case of a small vocabulary task, the system is able to learn patterns corresponding to words and subsequently detects the presence of these words in speech utterances. Without the prior requirement of expert knowledge about the speech as is the case in conventional automatic speech recognition, we illustrate that the learning algorithm achieves a promising accuracy and noise robustness.

The NMF work in chapter 5 uses a carefully selected time-frequency representation as input. Patches are identified and in a second stage patch activations are obtained that express to what extent each patch is present across time. The NMF-based approach does not consider the use of auditory motivated spectro-temporal features. The work in chapter 4 provides a framework to design improved spectro-temporal features and hence, it may be worthy to investigate in future the issue of using auditory motivated spectro-temporal features for an NMF-based word learning algorithm.

We can draw some general conclusions from this deliverable. We first consider the auditory periphery based feature selection/definition work. Our results show that the knowledge implicit in the human auditory periphery can be used to define or improve features for speech recognition without any knowledge or the meaning of the signal. In practice this means that a sophisticated auditory model can be used to find a feature set that improves speech recognition performance over standard methods for a large range of environmental conditions. The success of our perceptual-distance preserving measure in optimizing features suggests that the auditory system provides as output a signal representation that is 'efficient' for speech recognition. In a more general sense, our work reinforces the common scientific perception that improved features can be designed through the use of auditory knowledge and the resulting features are sufficiently generic to be used by any pattern recognizer/learning algorithm. Secondly, we consider the feature-*discovery* work. We conclude that, given a carefully chosen time-frequency input representation, it is possible to discover automatically *i)* elementary time-frequency structures (high-level features, or patches) and *ii)* words that each consist of a contiguous set of the elementary time frequency structures. While the result was shown for a small vocabulary, it is consistent with the notion that humans do not need to be and indeed are not pre-programmed with a set of elementary time-frequency structures for the purpose of speech recognition.

The software for the two components forms part of the software that is available to the partners of the ACORNS project. Some of the software will be made publicly available.

Chapter 2

Selecting static and dynamic features using auditory model

C. Koniaris, S. Chatterjee and W.B. Kleijn (KTH)

2.1 Introduction

In speech recognition, the goal of acoustic-feature selection is to reduce input dimensionality while retaining most of the information relevant for accurate classification. Although a large feature cardinality results in principle in a better recognition accuracy, in practice it is not possible to train or use such an automatic speech recognition (ASR) system. This phenomenon is called the *curse of dimensionality* [1].

The selection of appropriate features is not a trivial task. Generally a large pool of possible features is used as a starting point. For a good selection algorithm, the performance of the ASR system increases when the cardinality of the feature set is reduced from that initial pool, e.g., [2]. This increase in performance results from improved generalization as the number of system parameters is reduced. Many approaches to feature selection have been developed, e.g., [2, 3, 4]. Generally these methods require annotated speech data and feedback on classification performance from an ASR system. This means they are optimized for a particular ASR system. The methods require ASR systems to be trained for the full set first. They do not use knowledge implicit in the human auditory system, which has become available with the development of sophisticated models of the auditory periphery, e.g., [5, 6].

In our previous work [7], we presented a new acoustic-feature-selection method that was based on human perception only, called *auditory model-based feature selection* (AMFS) for speech recognition. Inspired by the observation that the best recognition performance can be achieved by a normal-hearing individual, we use knowledge of the human auditory periphery to select robust feature subsets. The selection of acoustic features is performed by finding the subset that maximizes the similarity of the Euclidian geometry of the feature set and the human auditory representation of the signal. In [7], we considered only static features - for our speech recognition application we applied it to mel-frequency cepstrum coefficients (MFCCs) [8]. Accordingly, a static psycho-acoustic masking model [5] was considered. The results showed a significant performance increase compared to the linear discriminant analysis (LDA) [2] method and compared to an average of randomly selected feature subsets in various noisy conditions. The performance increase was sustained when first and second-

order derivatives of the selected subsets were included as input for the ASR system.

In this paper, we further develop the method to enable the selection of a subset of features from a set of both static and dynamic features. To make this possible, we consider a more sophisticated auditory model [6] that we refer to as the Dau model. This model accounts for the spectro-temporal processing of sound signals by the human auditory periphery. Let the complete feature set characterize certain (locally) audible components of the speech signal. Then, for a given subset cardinality, the auditory model is used to select the feature subset that best captures the most audible of these signal components.

To validate our approach, we used the MFCCs and their first and second time derivatives. Our results verify that the human auditory model forms an effective basis for selecting robust acoustic features from this set of static and dynamic features.

This paper is organized as follows. Sec. 2.2 discusses a similarity measure for the perceptual and feature domains. Sec. 2.3 presents the auditory model used and describes the selection algorithm. Sec. 2.4 shows experimental results and Sec. 2.5, provides conclusions.

2.2 Maximizing the perceptual relevance of features

The high recognition accuracy of the normal-hearing listeners indicates that the auditory periphery system is efficient in sound-class discrimination. Implicitly, this suggests that the necessary information for sound classification reaches the 'auditory cortex' of the brain where the cognitive processing is performed. As ASR systems require a perceptually relevant set of acoustic features to achieve a high recognition rate, it is natural to find a set that is similar to that used by humans.

Our research is aimed at finding robust feature subsets based on quantitative models of the auditory periphery. The best-case scenario would be that the selected features include all the information relayed by the human auditory periphery. This would lead to the ideal situation in which the perceptual and the selected acoustic-feature domains are isometric. In practice this is not completely possible. In this section we define a measure that has proven to be effective in maximizing similarity between the perceptual domain and the acoustic-feature domain.

2.2.1 A measure of dissimilarity

It is our objective to obtain a feature set for which the geometry of the data (characterized by the distances between sounds) is similar to that of the perceptual domain (the output of the auditory periphery). This is particularly the case for "short" distances, as they determine the performance of classifiers. Let us define an L_2 norm based distance measure (also known as distortion measure) in the perceptual domain as a mapping of two signals: $\Upsilon : \mathbb{R}^N \times \mathbb{R}^N \rightarrow \mathbb{R}^+$, where $\mathbb{R}^+$ are the non-negative reals. Let us also consider the perceptual-domain signals $\mathbf{y}(\mathbf{x}_j)$ and $\mathbf{y}(\hat{\mathbf{x}}_{j,m})$, where $\mathbf{y} : \mathbb{R}^N \rightarrow \mathbb{R}^K$ is a mapping to the (K-dimensional) perceptual domain. Then

$$\Upsilon(\mathbf{x}_j, \hat{\mathbf{x}}_{j,m}) = \|\mathbf{y}(\mathbf{x}_j) - \mathbf{y}(\hat{\mathbf{x}}_{j,m})\|^2, \tag{2.1}$$

where $\mathbf{x}_j \in \mathbb{R}^N$ denotes the N-dimensional speech signal vector characterizing a segment with time index $j \in \mathbb{Z}$ and $\hat{\mathbf{x}}_{j,m}$ is a perturbation of $\mathbf{x}_j$ with perturbation index m.

Similarly, a distance measure for the feature domain can be defined as $\Gamma_i : \mathbb{R}^N \times \mathbb{R}^N \rightarrow \mathbb{R}^+$ for the feature set i. Let $\mathbf{c}_i : \mathbb{R}^N \rightarrow \mathbb{R}^L$ be the mapping from a signal segment $\mathbf{x}_j$ to a set of L features $\mathbf{c}_i(\mathbf{x}_j)$ with set index i. An L^2 norm based measure is then

$$\Gamma_i(\mathbf{x}_j, \hat{\mathbf{x}}_{j,m}) = \|\mathbf{c}_i(\mathbf{x}_j) - \mathbf{c}_i(\hat{\mathbf{x}}_{j,m})\|^2. \tag{2.2}$$

Our geometry objective is equivalent to find the particular set of features i that minimizes a *measure of dissimilarity* of the perceptual-domain distances and the feature-domain distances. We find the set i that minimizes

$$G(i) = \sum_{j\in\mathcal{J}, m\in\mathcal{M}_j} \left[\Upsilon(\mathbf{x}_j, \hat{\mathbf{x}}_{j,m}) - \lambda\Gamma_i(\mathbf{x}_j, \hat{\mathbf{x}}_{j,m})\right]^2, \tag{2.3}$$

where $\lambda = \frac{\sum_{j\in\mathcal{J}, m\in\mathcal{M}_j} \Upsilon(\mathbf{x}_j,\hat{\mathbf{x}}_{j,m})\Gamma_i(\mathbf{x}_j,\hat{\mathbf{x}}_{j,m})}{\sum_{j\in J, m\in M_j} \Gamma_i(\mathbf{x}_j,\hat{\mathbf{x}}_{j,m})^2}$ is an optimal scaling of the acoustic feature criterion, and $j \in \mathcal{J}$, $m \in \mathcal{M}_j$ represent a finite frame sequence and a finite set of acoustic perturbations, respectively.

2.2.2 Perturbation analysis

To reduce computational effort, we use the powerful tools of perturbation analysis and the sensitivity matrix [9, 10, 11, 12]. We compute eq. (2.3) by approximating the perceptual distortion measure $\Upsilon(\mathbf{x}_j, \hat{\mathbf{x}}_{j,m})$ by a simpler quadratic measure. Note that perturbation analysis is natural as we consider only small distances in eq. (2.3).

Let consider $\Upsilon(\mathbf{x}_j, \hat{\mathbf{x}}_{j,m})$ to be known. We assume that $\Upsilon(\mathbf{x}_j, \mathbf{x}_j) = 0$ and that this forms a minimum. We furthermore assume that $\Upsilon(\mathbf{x}_j, \hat{\mathbf{x}}_{j,m})$ is analytic in $\hat{\mathbf{x}}_{j,m}$. Then, for sufficiently small perturbations $\hat{\mathbf{x}}_{j,m} - \mathbf{x}_j$, we can make the approximation

$$\Upsilon(\mathbf{x}_j, \hat{\mathbf{x}}_{j,m}) \approx [\hat{\mathbf{x}}_{j,m} - \mathbf{x}_j]^T \mathbf{D}_\Upsilon(\mathbf{x}_j)[\hat{\mathbf{x}}_{j,m} - \mathbf{x}_j], \tag{2.4}$$

where $\mathbf{D}_{\Upsilon,\kappa\mu}(\mathbf{x}_j) = \left.\frac{\partial^2\Upsilon(\mathbf{x}_j,\hat{\mathbf{x}}_{j,m})}{\partial\hat{x}_\kappa\partial\hat{x}_\mu}\right|_{\hat{\mathbf{x}}_{j,m}=\mathbf{x}_j}$ is the *sensitivity matrix*.

2.3 Application to speech recognition

This section illustrates the application of the method to a specific auditory model and specific type of acoustic features and presents the implementation of the feature selection algorithm.

2.3.1 Dau auditory model

The Dau [6, 13] auditory model is a psycho-acoustic masking model that accounts for spectro-temporal processing of sound signals. Thus, the speech signal $\mathbf{x}$ is a time-domain vector. It consists of several stages which simulate the human auditory periphery. A channel l of the Dau model includes the hair-cell model consisting of a gamma-tone filter, a half-way rectifier, and a low-pass filter. Next, an adaptation nonlinear stage incorporates the forward masking prediction of the ear [12]. Finally, a low-pass filter performs a temporal smoothing and the output is the so-called internal representation $\mathbf{a}^{(l)}(\mathbf{x}_j)$. The original paper [6] did not study the distortion prediction properties of the model, an investigation that was later performed in [12]. In the same paper, a distortion measure on the internal representation was introduced as

$$\Upsilon(\mathbf{x}_j, \hat{\mathbf{x}}_{j,m}) = \sum_l \|\mathbf{a}^{(l)}(\mathbf{x}'_j) - \mathbf{a}^{(l)}(\hat{\mathbf{x}}'_{j,m})\|^2, \tag{2.5}$$

where $\mathbf{x}'_j, \hat{\mathbf{x}}'_{j,m}$ are of higher dimension than the $\mathbf{x}_j, \hat{\mathbf{x}}_{j,m}$ vectors, respectively due to the ring-out effect described in [12]. The derivation of the sensitivity matrix in this case is non-trivial and tedious.

The final sensitivity matrix can be computed as the sum of per-channel sensitivity matrices $\mathbf{D}_{\Upsilon}^{(l)}(\mathbf{x}_j)$

$$\mathbf{D}_{\Upsilon}(\mathbf{x}_j) = \sum_l \mathbf{D}_{\Upsilon}^{(l)}(\mathbf{x}_j), \tag{2.6}$$

where

$$\mathbf{D}_{\Upsilon}^{(l)}(\mathbf{x}_j) = 2\left[\prod_k \mathbf{J}_k^{(l)}\right]^H \mathbf{J}_k^{(l)}, \tag{2.7}$$

and $\mathbf{J}_k^{(l)}$ is the Jacobian for processing stage k in channel l.

2.3.2 AMFS algorithm for a spectro-temporal model

We can now discuss the implementation of a AMFS algorithm that accounts for both spectral and temporal masking. To handle the dynamic aspect of the auditory measure we use a superframe J that consists of three overlapped subframes $j-1, j, j+1$ as shown in the Fig. 2.1. The length of the superframe was set at 40 ms with an overlap of 10 ms. The length of each subframe was 20 ms with an overlap of 10 ms for the AURORA 2 database of 8 kHz. The superframe is used to compute the Dau sensitivity matrix and the subframes are used to compute the static MFCCs and their first- and second-order derivatives..

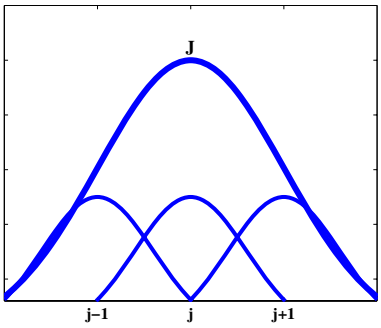

Figure 2.1: The J'th superframe consists of three overlapped subframes $j-1, j, j+1$.

The feature selection operation was divided into two stages. In the first stage all necessary quantities were computed, and in the second stage the feature set was selected.

In the first stage the Dau sensitivity matrix $\mathbf{D}_{\Upsilon}(\mathbf{x}_J)$ was computed for the J'th superframe, $\mathbf{x}_J$. A set of 100 vectors $\hat{\mathbf{x}}_{J,m}$ was computed by adding 100 dB SNR noise to $\mathbf{x}_J$. Next, the 12 mel cepstrum static coefficients were calculated for the central j'th subframe. Their first-order (Δ) and second-order ($\Delta\Delta$) derivatives were computed using the $j-1$'th and the $j+1$'th subframes. The full feature vector $\mathbf{c}_J$ for the superframe J was built as

$$\mathbf{c}_J = [\mathbf{c}_j \;\; \Delta\mathbf{c}_j \;\; \Delta\Delta\mathbf{c}_j]^T. \tag{2.8}$$

Finally, we computed the 100 distorted features vectors $\hat{\mathbf{c}}_{J,m}$'s corresponding to the 100 distorted speech vectors that were previously calculated. The first step of the algorithm ended by performing a cepstral mean and variance normalization (CMVN) in the feature vectors.

The second stage began by reading again the speech data. For each superframe, we considered the quantities that were computed (and saved) in the previous stage. We used them to compute the distortion measure in the perceptual domain according to eq. (2.4). Similarly, we computed the distortion in the feature domain for all the possible feature subsets i according to eq. (2.2). Finally, to find the particular set of features i we computed the measure of dissimilarity as described in eq. (2.3).

2.3.3 Greedy feature selection

Due to the high cardinality of the full feature vector, the search through all combinations of possibly optimal subsets is practically impossible. For example when a subset of 32-out-of-36 features is requested, $58\,905$ different combinations have to be considered. To cope with this problem we used a greedy algorithm. The second stage of the algorithm was run iteratively, reducing the dimensionality by one dimension at each iteration. A feature that was not included in the selected subset was not considered in the next round. We did not consider forward-backward algorithms. Therefore, to select 32 coefficients, we first start from selecting 35 coefficients out of 36 coefficients. Then, in the next step, 34 coefficients are chosen from those previously selected 35 coefficients and the algorithmic process continues until 32 coefficients are selected out of previously selected 33 coefficients.

2.4 Experiments

To test and compare the performance of the new feature-selection system, we performed speech recognition experiments using the test set A of the AURORA2 [14] database, sampled at 8 kHz. We used an MFCC representation. The full set of MFCCs was extracted by using a Hamming window of 20 ms with an overlap of 10 ms. The DFT dimensionality was 256 and we considered 23 mel filters. A set of 12 conventional MFCCs was extracted plus the energy coefficient and their corresponding first and second order time derivatives (resulting in a 39 feature vector). The selection of a subset from the full feature vector was performed with three different feature-selection methods. The first method is the new auditory-model based feature selection (indicated by *amfs* in the tables) described in section 2.3. The other two methods are the well-known linear-discriminant analysis (*lda*) [2] and heteroscedastic linear discriminant analysis (*hlda*) [3] methods.

For the discriminant analysis methods we used the HTK [15] toolkit to build label files for each word-state ('word' refers to the digit to be recognized while 'state' refers to the state of the hidden Markov model (HMM)). Hence, we used information from the classified data for both *lda* and *hlda* methods.

To build the recognizer we also used the HTK [15] toolkit. The digits were modeled as whole word HMMs with 16 states (HTK's notation is 18 states including the beginning and end states) and three Gaussian mixture components per state (we used diagonal covariance matrices). An initial model with global data means and covariances, identical for each digit, was used and 16 iterations were used to build the final model.

2.4.1 Results

Table 2.1 shows the recognition accuracy for training on clean data for various feature subsets starting from 33-out-of-36 and ending at 12-out-of-36 extended with the energy coefficient indicated as "$+E$" and its dynamics "$+\Delta E$" and "$+\Delta\Delta E$" for the velocity and acceleration coefficients, respectively. As reference the corresponding 39-dimensional feature set is provided. The training was performed on the clean training set of $8\,440$ sentences and the testing on the $20\,020$ data of test set A, consisting of

4 004 data from each of the 'clean', '20', '15', '10' and '5' dB SNR groups of data. The performance of the *amfs* remains relatively stable as the number of coefficients is reduced. The *amfs* features generally perform better than both the *lda* and *hlda* feature, particularly under noisy testing conditions.

Table 2.1: AURORA2 clean training results.

feature set		*Data Test Set A*				
		clean	*20 dB*	*15 dB*	*10 dB*	*5 dB*
36+E+ΔE+ΔΔE (full 39-dim)		99.1	96.5	93.0	84.4	66.2
33+E+ΔE+ΔΔE	*amfs*	99.1	96.6	92.9	83.5	64.7
	lda	98.3	89.6	71.6	40.9	19.1
	hlda	98.6	94.8	90.2	80.5	62.3
30+E+ΔE+ΔΔE	*amfs*	99.0	96.5	92.6	82.9	63.3
	lda	98.3	89.9	72.6	42.3	19.7
	hlda	98.7	94.9	89.8	78.6	58.8
27+E+ΔE+ΔΔE	*amfs*	99.1	96.4	92.5	82.9	63.1
	lda	98.4	90.1	73.7	43.3	21.2
	hlda	98.8	94.9	89.2	77.1	55.7
24+E+ΔE+ΔΔE	*amfs*	99.2	96.3	92.1	82.2	62.3
	lda	98.3	89.9	74.2	45.4	22.8
	hlda	98.7	94.7	88.6	75.8	53.8
21+E+ΔE+ΔΔE	*amfs*	99.1	96.2	91.9	81.8	62.5
	lda	98.3	87.3	71.0	42.8	21.9
	hlda	98.8	94.2	88.2	75.0	52.6
18+E+ΔE+ΔΔE	*amfs*	98.5	93.8	88.1	75.2	55.3
	lda	98.3	84.7	68.6	41.5	22.7
	hlda	98.7	94.0	87.4	73.3	49.9
15+E+ΔE+ΔΔE	*amfs*	98.3	91.9	84.8	70.8	49.3
	lda	98.4	86.7	70.8	46.2	24.5
	hlda	98.6	93.7	86.3	71.1	45.1
12+E+ΔE+ΔΔE	*amfs*	96.8	87.3	78.0	61.8	41.0
	lda	98.2	77.6	57.2	34.2	15.1
	hlda	98.6	94.0	86.6	71.3	45.8

Table 2.2 shows the recognition accuracy for noisy only data. The training was performed on the multi-conditioned noisy training set consisting of 6 752 files and the testing on 20 020 noisy data of test set A. Note that the results for the noisy test conditions shown in tables 2.1 and 2.2 are averaged over subway, babble, car, and exhibition additive noise for several SNR values. The performance of the *amfs* subsets are in general better than the *lda* and *hlda*.

2.4.2 Discussion

An interesting aspect of the results is the observation of how many features are necessary to achieve at least 99 % of the original recognition performance of the 39 dimensional feature vector. Studying table 2.1, we see that for the *clean-clean* case *amfs* needs 18 features, for *clean*-20 *dB* case 24 features, for *clean*-15 *dB* 27, for *clean*-10 *dB* 36 and for the case of *clean*-5 *dB* it cannot reach this goal. For the

Table 2.2: AURORA2 multi-conditioning noisy training results.

feature set		*Data Test Set A*				
		20 dB	*15 dB*	*10 dB*	*5 dB*	*0 dB*
36+E+ΔE+ΔΔE (full 39-dim)		97.7	96.9	94.8	87.6	71.4
33+E+ΔE+ΔΔE	*amfs*	97.6	96.7	94.5	86.9	68.9
	lda	94.7	93.1	88.5	77.7	53.0
	hlda	95.4	94.0	90.3	80.4	59.5
30+E+ΔE+ΔΔE	*amfs*	97.6	96.6	94.4	86.9	68.3
	lda	94.6	93.1	88.4	77.3	52.8
	hlda	95.3	93.9	90.0	80.1	59.3
27+E+ΔE+ΔΔE	*amfs*	97.4	96.4	94.0	85.8	66.9
	lda	94.3	92.7	87.7	76.9	52.2
	hlda	95.0	93.5	89.9	79.6	59.0
24+E+ΔE+ΔΔE	*amfs*	97.1	96.0	93.2	84.2	64.2
	lda	94.4	92.8	87.8	76.5	52.0
	hlda	94.8	93.4	89.4	79.3	58.4
21+E+ΔE+ΔΔE	*amfs*	97.1	95.9	93.2	84.0	64.7
	lda	94.2	92.7	87.5	75.7	51.3
	hlda	94.9	93.3	89.3	79.4	59.3
18+E+ΔE+ΔΔE	*amfs*	96.5	95.0	90.5	78.9	56.4
	lda	94.1	92.3	87.3	75.4	50.7
	hlda	94.5	92.9	88.8	78.6	58.2
15+E+ΔE+ΔΔE	*amfs*	96.5	94.8	90.3	78.6	55.6
	lda	94.1	92.4	87.0	75.4	50.1
	hlda	94.5	92.7	89.1	78.7	58.5
12+E+ΔE+ΔΔE	*amfs*	96.5	94.6	89.6	77.3	53.6
	lda	93.6	92.1	86.3	73.9	48.9
	hlda	94.2	92.8	88.9	78.8	58.5

noisy training experiments, table 2.2 shows that the goal of $99\,\%$ can be reached with 24 features for the cases *noisy*-20 *dB* and *noisy*-15 *dB* and 30 for *noisy*-10 *dB*. In the case of *noisy*-5 *dB amfs* needs 33 features while for the case of *noisy*-0 *dB*, it has a lower accuracy. On the other hand, both *lda* and *hlda* cannot reach the $99\,\%$ goal in all cases but for the *clean-clean* case, where 15 dimensional feature vector suffices (although overall *hlda* is significant better than *lda*). The obvious inference is that the recognition process can become faster when *amfs* is used without dropping more than $1\,\%$ of the maximum performance since the speed of a HMM ASR system depends on the feature vector size.

2.5 Conclusions

In this paper, we presented an advanced method to select robust features subsets for speech recognition based on a spectro-temporal advanced auditory model. Under the assumption of small distortion errors we used perturbation analysis and the sensitivity matrix to build the feature selection algorithm. We included the first and second order time derivatives of the features and showed how we succeeded to associate them to the perception information obtained from the sensitivity matrix. The experimen-

tal results verified that the human auditory periphery is a powerful tool in selecting relevant features for robust speech recognition.

Bibliography

[1] R. E. Bellman, *Dynamic Programming*. Princeton University Press, Princeton, NJ, USA, 1957, Republished Dover 2003, ISBN: 0-486-42809-5.

[2] R. Haeb-Umbach and H. Ney, "Linear discriminant analysis for improved large vocabulary continuous speech recognition," *IEEE Int. Conf. on Acoust., Speech, Sig. Proc.*, vol. 1, pp. 13–16, 1992.

[3] N. Kumar and A. G. Andreou, "Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition," *Speech Commun.*, vol. 26, no. 4, pp. 283–297, 1998.

[4] F. Valente and C. Wellekens, "Maximum entropy discrimination (MED) feature subset selection for speech recognition," *IEEE Works. on ASRU*, pp. 327–332, Dec. 2003.

[5] S. van de Par, A. Kohlrausch, G. Charestan, and R. Heusdens, "A new psychoacoustical masking model for audio coding applications," in *IEEE Int. Conf. on Acoust., Speech, Sig. Proc.*, vol. 2, 2002, pp. 1805–1808.

[6] T. Dau, D. Püschel, and A. Kohlrausch, "A quantitative model of the effective signal processing in the auditory system. I. model structure," *Acoustical Society of America*, vol. 99, no. 6, pp. 3615–3622, Jun. 1996.

[7] C. Koniaris, M. Kuropatwinksi, and W. B. Kleijn, "Auditory-model based robust feature selection for speech recognition," *Under Review at the Journal of the Acoustical Society of America (JASA) Express Letters*, 2009.

[8] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. Acoust., Speech, Sig. Proc.*, vol. 28, no. 4, pp. 357–366, Aug. 1980.

[9] W. R. Gardner and B. D. Rao, "Theoretical analysis of the high-rate vector quantization of LPC parameters," *IEEE Trans. Speech, Audio Proc.*, vol. 3, no. 5, pp. 367–381, Sep. 1995.

[10] T. Linder, R. Zamir, and K. Zeger, "High-resolution source coding for non-difference distortion measures: multidimensional companding," *IEEE Trans. Inform. Theory*, vol. 45, no. 2, pp. 548–561, Mar. 1999.

[11] J. Li, N. Chaddha, and R. M. Gray, "Asymptotic performance of vector quantizers with a perceptual distortion measure," *IEEE Trans. Inform. Theory*, vol. 45, no. 4, pp. 1082–1091, May 1999.

[12] J. H. Plasberg and W. B. Kleijn, "The sensitivity matrix: Using advanced auditory models in speech and audio processing," *IEEE Trans. Speech, Audio Proc.*, vol. 15, no. 1, pp. 310–319, Jan. 2007.

[13] T. Dau, D. Püschel, and A. Kohlrausch, "A quantitative model of the effective signal processing in the auditory system. II. simulations and measurements," *Acoustical Society of America*, vol. 99, no. 6, pp. 3623–3631, Jun. 1996.

[14] H. G. Hirsch and D. Pearce, "The AURORA experimental framework for the performance evaluation of speech recognition systems under noise conditions," *ISCA ITRW ASR2000, Paris, France*.

[15] S. Y. et al, *The HTK Book (for HTK Version 3.2)*. Cambridge University, Engineering Department, Dec. 2002.

Chapter 3

Auditory Model Based Optimization of MFCC Features

S. Chatterjee, C. Koniaris and W.B. Kleijn (KTH)

3.1 Introduction

An automatic speech recognition (ASR) system comprises two main tasks: feature extraction and pattern recognition. The feature extraction stage is designed to transform the incoming speech signal into a representation that serves as the input to a later pattern recognition stage. Feature extraction is a dimensionality reduction problem where the output representation should preserve the important aspects of the input speech signal relevant for speech recognition in any environmental condition, clean as well as noisy.

Different feature sets have been proposed in the literature, but the solutions remain ad hoc. We propose to define the features based on a perceptually relevant objective criterion. The human peripheral auditory system enhances the input speech signal for further processing by the central auditory system of the brain. Pre-processing of the input speech signal by the human auditory periphery forms a useful basis for designing an efficient feature set. Commonly used features use knowledge of the auditory system in an *ad hoc* manner. For example, several feature extraction methods perform auditory frequency filtering on a perceptually motivated frequency scale than a linear scale. Another example is the use of a logarithmic function to approximate the non-linear dynamic compression in the auditory system, which allows us to cover the large dynamic range between hearing threshold and uncomfortable loudness level. Using these two auditory motivated signal processing techniques, MFCCs were designed a few decades ago [1]. They are still universally used due to their computational simplicity as well as good performance. Importantly, the MFCCs do not use up-to-date quantitative knowledge of the auditory system.

Several attempts have been made to use quantitative auditory models in a practical ASR system processing chain [2]-[7]. In these techniques, the input speech signal is first processed through a readily available auditory model and then the output signal of the auditory model is formatted to use as an input to the pattern recognition stage of the ASR system. The direct use of an auditory model was shown to provide better speech recognition performance, but at the expense of higher computational complexity. In recent years, the research in quantitative modeling of the complex peripheral auditory system has reached a high level of sophistication [8]-[13], and it is appealing to

use a sophisticated auditory model for designing efficient features. The feature set should not incur the higher computational complexity associated with a full auditory model.

In this paper, instead of the direct on-line use, we investigate the use of an auditory model to design improved MFCCs through off-line optimization. The optimized MFCCs are referred to as *modified MFCCs* (MMFCCs). The off-line approach helps to retain the computational simplicity of MMFCCs. Also, it avoids the difficulty of formatting the output of the auditory model for recognition. Comparing to traditional MFCCs, the MMFCCs have a similar structure as well as computational simplicity.

In our approach, the feature set is optimized in such a way that it emulates the behavior of the human auditory system. The implementation of our method relies on perturbation theory and does not consider any feedback from the ASR system. We conjecture that human-like classification of speech sounds is facilitated by similarity between the local geometries of two domains, the feature domain and the perceptual domain. For improved classification, the preservation of the data geometry near the class boundaries is most critical. This means that 'small' Euclidean distances must be similar in the two different domains, except for an overall scaling. The focus on small distances allows a complex perceptual distance to be reduced to a quadratic distance measure using a sensitivity matrix based analysis. The sensitivity matrix based analysis was first developed in the context of source coding [14]. In [15], the sensitivity matrix was used to simplify an auditory distance measure for audio coding. Here, we extend the sensitivity matrix paradigm to optimize a feature set. Using HTK, the optimized MMFCCs are shown to provide better recognition performance than traditional MFCCs for both clean and noisy acoustic conditions.

3.2 Maximizing Similarity between Spaces

Improvement in sound classification requires a feature representation that provides a good separation of sound classes in the feature space. Noting the high human recognition performance, it can be expected that the output of a sophisticated auditory model provides good separation of sound classes. Therefore, we optimize a feature set to better describe the inter-sound distances of a state-of-the-art auditory model. We conjecture that if the Euclidean distance between two acoustic features approximates the corresponding perceptual distortion for two different speech sounds, then the use of that acoustic feature generally leads to better classification in an ASR system. Ideally, this implies an isometry between the perceptual and feature domains. The mapping from the perceptual to feature domain would then be distance preserving.

3.2.1 Distance Preserving Measure

In practice, it is not be possible to design a feature set that leads to an accurate distance-preserving mapping from perceptual domain to feature domain. However, it is not required to preserve all the distances. For good classification, the preservation of the data geometry near the class boundaries is most critical. More generally, the preservation of small distances (reflecting the local geometry) near the classification boundary is important, whereas the preservation of large distances (reflecting the global geometry) is not required. In principle, to achieve better sound classification, we then simply desire to have the same small distances for the auditory domain and for the feature domain.

A feature set is a function of an input speech signal segment (or frame) and some adjustable design parameters. For example, to design MMFCCs, these design parameters can be the frequency warping parameter to change the shape of a filter bank (such as heights, widths, center frequency of filters),

a parameter to change the shape of a compressing function (like logarithmic function), etc. The objective is to obtain a feature set with optimum parameters for which any small perturbation of the input speech signal segment leads to a Euclidean distance in the feature domain that best approximates the perceptual distortion indicated by the auditory model. Naturally this criterion has to hold for all speech segments. To measure the similarity of the auditory model distortion and the feature domain distance, a suitable objective measure needs to be designed that will provide a means of ensemble averaging over all speech segments and all perturbations. By optimizing the parameters, a higher similarity, through evaluating the objective measure, leads to a better feature set.

We now define an objective measure that relates between the perceptual and feature domains. Let us denote the signal vector for the j'th speech frame as $\mathbf{x}_j \in \mathbb{R}^N$, where $j \in J \subset \mathbb{Z}$, and the perceptual domain representation of $\mathbf{x}_j$ as $\mathbf{y} : \mathbb{R}^N \rightarrow \mathbb{R}^K$. We also denote the design parameters of a feature set by a vector $\mathbf{p} \in \mathbb{R}^S$. Then, we can denote the Q-dimensional feature derived from $\mathbf{x}_j$ using $\mathbf{p}$ as $\mathbf{c} : \mathbb{R}^N \times \mathbb{R}^S \rightarrow \mathbb{R}^Q$. The perceptual domain distortion is defined through a mapping as $\Upsilon : \mathbb{R}^K \times \mathbb{R}^K \rightarrow \mathbb{R}^+$, where $\mathbb{R}^+$ is the set of non-negative reals. For the j'th speech frame, let us denote the l'th perturbed signal as $\hat{\mathbf{x}}_{j,l}$. Often the perceptual distortion measure is based on the L^2 norm of the difference between the perceptual domain signal $\mathbf{y}(\mathbf{x}_j)$ and its distorted version $\mathbf{y}(\hat{\mathbf{x}}_{j,l})$. In that case, $\Upsilon(\mathbf{x}_j, \hat{\mathbf{x}}_{j,l}) = \|\mathbf{y}(\mathbf{x}_j) - \mathbf{y}(\hat{\mathbf{x}}_{j,l})\|^2$. Using the L^2 norm, we can define a distance measure for the feature $\mathbf{c}(\mathbf{x}_j, \mathbf{p})$ as $\Gamma(\mathbf{x}_j, \hat{\mathbf{x}}_{j,l}, \mathbf{p}) = \|\mathbf{c}(\mathbf{x}_j, \mathbf{p}) - \mathbf{c}(\hat{\mathbf{x}}_{j,l}, \mathbf{p})\|^2$. Now, considering the finite sequence of speech frames $j \in J$ and a finite set of acoustic perturbations $l \in L_j$, the objective is to minimize a measure of dissimilarity between perceptual domain distortion and feature domain distortion with respect to the parameter set $\mathbf{p}$. To satisfy this objective, a suitable norm based measure can be defined as

$$\mathbf{O} = \sum_{j \in J} \sum_{l \in L_j} \left[\Upsilon(\mathbf{x}_j, \hat{\mathbf{x}}_{j,l}) - \lambda\, \Gamma(\mathbf{x}_j, \hat{\mathbf{x}}_{j,l}, \mathbf{p})\right]^2, \tag{3.1}$$

where

$$\lambda = \frac{\sum_{j \in J} \sum_{l \in L_j} \Upsilon(\mathbf{x}_j, \hat{\mathbf{x}}_{j,l})\; \Gamma(\mathbf{x}_j, \hat{\mathbf{x}}_{j,l}, \mathbf{p})}{\sum_{j \in J} \sum_{l \in L_j} \left(\Gamma(\mathbf{x}_j, \hat{\mathbf{x}}_{j,l}, \mathbf{p})\right)^2}. \tag{3.2}$$

Here λ is the necessary scaling to eliminate the effect of a scale mismatch between perceptual domain and feature domain. So, the objective is to minimize the norm based distance $\mathbf{O}$ with respect to the parameter vector $\mathbf{p}$.

3.2.2 Perturbation Analysis

While it is possible to minimize the objective measure of eq. (3.1) even for complex distortion measures, this can be computationally expensive. Since we are interested in small distances, we can approximate the perceptual and feature domain distortion measure using simpler quadratic measures, leading to a significant reduction in computational complexity and an increase in mathematical tractability. This approach is based on the sensitivity matrix framework [14], [15].

Let us omit the subscripts for notational brevity where no ambiguity exists. We assume that $\Upsilon(\mathbf{x}, \hat{\mathbf{x}})$ is analytic and $\Upsilon(\mathbf{x}, \mathbf{x}) = 0$. Then, for a sufficiently small perturbation $\hat{\mathbf{x}} - \mathbf{x}$, we can write

$$\Upsilon(\mathbf{x}, \hat{\mathbf{x}}) \approx \frac{1}{2}[\hat{\mathbf{x}} - \mathbf{x}]^T \mathbf{D}_\Upsilon(\mathbf{x})\, [\hat{\mathbf{x}} - \mathbf{x}], \tag{3.3}$$

where $\mathbf{D}_\Upsilon(\mathbf{x})$ is the sensitivity matrix whose elements are $\mathbf{D}_{\Upsilon,ij}(\mathbf{x}) = \frac{\partial^2 \Upsilon(\mathbf{x},\hat{\mathbf{x}})}{\partial \hat{x}_i \, \partial \hat{x}_j}\Big|_{\hat{\mathbf{x}}=\mathbf{x}}$. In certain cases, such as the spectral auditory model of section 3.2.3, $\Upsilon(\mathbf{x},\hat{\mathbf{x}})$ and $\mathbf{D}_\Upsilon(\mathbf{x})$ are known.

Next, we consider a simplification of the distortion in the feature domain i.e., $\Gamma(\mathbf{x},\hat{\mathbf{x}},\mathbf{p})$. If the mapping $\mathbf{c}(\mathbf{x},\mathbf{p})$ is analytic in $\mathbf{x}$, then we can use the Taylor series expansion to make a local approximation around $\mathbf{x}$ as

$$\mathbf{c}(\hat{\mathbf{x}},\mathbf{p}) \approx \mathbf{c}(\mathbf{x},\mathbf{p}) + \mathbf{A}(\mathbf{p})\ [\hat{\mathbf{x}} - \mathbf{x}], \tag{3.4}$$

where $\mathbf{A}(\mathbf{p})$ is a $Q \times N$-dimensional matrix as $\mathbf{A}(\mathbf{p}) = \frac{\partial \mathbf{c}(\mathbf{x},\mathbf{p})}{\partial \hat{\mathbf{x}}}\Big|_{\hat{\mathbf{x}}=\mathbf{x}}$. We can then write the distortion in the feature domain as

$$\begin{aligned} \Gamma(\mathbf{x},\hat{\mathbf{x}},\mathbf{p}) &= \|\mathbf{c}(\mathbf{x},\mathbf{p}) - \mathbf{c}(\hat{\mathbf{x}},\mathbf{p})\|^2 \\ &= [\hat{\mathbf{x}} - \mathbf{x}]^T \mathbf{A}(\mathbf{p})^T\, \mathbf{A}(\mathbf{p})\, [\hat{\mathbf{x}} - \mathbf{x}]. \end{aligned} \tag{3.5}$$

3.2.3 A Spectral Auditory Model

In this paper, we optimize the MMFCCs to minimize the norm based measure of eq. (3.1). The MMFCCs are designed using the power spectrum of the input speech signal. Therefore, for optimization, we use the spectral auditory model developed by van de Par, et al. [13] which is referred to as the van de Par auditory model (VAM). The VAM is a psycho-acoustic masking model that accounts for simultaneous processing of sound signals with different frequencies. To use the VAM, we consider the input signal $\mathbf{x}$ as the power spectrum of a speech frame. The VAM consists of several frequency channels, in each of which the ratio of distortion power to masker power is calculated. Then, the ratios of all the frequency channels are combined together to account for the spectral integration property of the human auditory system. Let $\mathbf{H}$ be a diagonal N-dimensional matrix whose diagonal is formed by the frequency response of the outer and middle ear filter. In the same fashion, a diagonal matrix $\mathbf{G}_i$ is defined, so that the frequency response of the i'th channel Gamma-tone auditory filter forms its diagonal. For the VAM, the diagonal sensitivity matrix is

$$\mathbf{D}_\Upsilon(\mathbf{x}) \approx 2\frac{C_s L_e}{N} \sum_i \frac{[\mathbf{G}_i\,\mathbf{H}]^T [\mathbf{G}_i\,\mathbf{H}]}{\frac{1}{N}[\mathbf{G}_i\,\mathbf{H}\,\mathbf{x}]^T [\mathbf{G}_i\,\mathbf{H}\,\mathbf{x}] + C_a}, \tag{3.6}$$

where C_s and C_a are constants calibrated based on measurement data, and L_e is a constant to account for the influence of temporal integration time in the human auditory system on frame duration.

It is important to mention that each speech frame is independently analyzed in the VAM. Therefore, the use of VAM is appropriate for optimizing a static feature. Note that due to the inability to model the auditory response across speech frames, the use of the VAM is inappropriate for optimizing the temporal dynamic features, such as velocity and acceleration. However, it is possible to compute the dynamic features from any static feature using standard regression method.

3.3 Modified MFCCs

We first generalize the definition of the MFCCs to render a set of features with adjustable parameters $\mathbf{p}$. We refer to this new set of features as *modified MFCCs* (MMFCCs). Let the N-dimensional vector $\mathbf{x} = [x_0\ x_1, \cdots, x_n, \cdots, x_{N-1}]^T$ be the power spectrum of a Hamming windowed speech frame. Then the steps of evaluating the MMFCCs are as follows:

1. Calculation of the energy in each channel:

$$\begin{aligned} z_m &= \mathbf{x}^T \mathbf{w}_m(\alpha) \\ &= \sum_{n=0}^{N-1} x_n \times w_{m,n}(\alpha), 0 \leq m \leq M-1, \end{aligned} \tag{3.7}$$

where $\mathbf{w}_m(\alpha)$ is the N-dimensional vector denoting the triangular filter of the m'th channel and satisfies $\sum_{n=0}^{N-1} w_{m,n}(\alpha) = 1$. M is the total number of channels with a typical value of $M = 26$. The shape of a triangular filter depends on the extent of frequency warping. The warped frequency scale [16] is given as

$$f_{warp} = 2595 \times \log_{10}(1 + (f/\alpha)), \tag{3.8}$$

where α is the warping factor and f is the frequency in Hz. An increase in α leads to a decrease in the extent of warping. For the MMFCCs, α is a parameter to optimize to achieve better recognition performance. In the case of MFCCs, the triangular filters are designed using the *mel* frequency scale where $\alpha = 700$ [16].

2. Compression of the dynamic range of the energy in each channel:

$$s_m = \log_{10}\left[\sum_{r=1}^{R} b_r (z_m)^r\right], \ 0 \leq m \leq M-1, \tag{3.9}$$

where $\sum_{r=1}^{R} b_r = 1$ and $b_r \geq 0$. For the MMFCCs, we optimize the polynomial coefficients $\{b_r\}_{r=1}^{R}$. In the case of MFCCs, $R = 1$ and $b_1 = 1$ [1]. We note that eq. (4.3) implies that our results are scale dependent and require proper normalization.

3. De-correlation using the DCT to evaluate Q-dimensional MMFCC feature vector:

$$c_q = \sum_{m=0}^{M-1} s_m \times \cos\left[q\,(m+0.5)\,\frac{\pi}{M}\right], 1 \leq q \leq Q. \tag{3.10}$$

A typical value of feature vector dimension is $Q = 12$.

3.3.1 Optimization of the MMFCCs

The parameters that we optimize to obtain the MMFCCs are $\mathbf{p} = \left[\alpha, \{b_r\}_{r=1}^{R}\right]$. To optimize the parameters, we need to minimize the objective measure $\mathbf{O}$ of eq. (3.1). This objective measure is a function of the sensitivity matrix based perceptual domain distortion of eq. (3.3) and the feature domain distortion of eq. (3.5). To evaluate the perceptual domain distortion, we need a closed form sensitivity matrix $\mathbf{D}_{\Upsilon}(\mathbf{x})$ which is given by the VAM as shown in eq. (3.6). We also need a closed form $\mathbf{A}(\mathbf{p})$ for evaluating the feature domain distortion. For an MMFCC feature, the elements of the matrix $\mathbf{A}(\mathbf{p})$ are

$$\begin{aligned} A_{qn} &= \frac{\partial c_q}{\partial x_n} = \frac{\partial c_q}{\partial s_m}\frac{\partial s_m}{\partial z_m}\frac{\partial z_m}{\partial s_n} \\ &= \sum_{m=0}^{M-1} \cos\left[q\,(m+0.5)\,\tfrac{\pi}{M}\right] \\ &\quad \times \frac{\sum_{r=1}^{R} r\, b_r\, (z_m)^{r-1}}{\ln 10 \ \times \sum_{r=1}^{R} b_r\, (z_m)^r}\, w_{m,n}(\alpha). \end{aligned} \tag{3.11}$$

Table 3.1: Phone recognition accuracy (in %) of static 12-dimensional MFCC and MMFCC features using TIMIT

Feature	Number of Gaussian mixtures/state							
	1	2	4	6	8	10	12	14
MFCC	43.19	47.00	48.63	49.57	50.38	51.04	51.60	51.96
MMFCC	45.13	48.63	50.16	51.28	52.10	52.63	52.92	53.30

It is interesting to jointly optimize all the parameters through a closed-form/iterative solution, such as using gradient descent search technique. This requires a closed form gradient expression $\frac{d\mathbf{O}}{d\mathbf{p}}$, which is not easy to evaluate due to the intricate relationship existing between the measure of $\mathbf{O}$ and the parameter vector $\mathbf{p} = \left[\alpha, \{b_r\}_{r=1}^{R}\right]$. Therefore, we use a simple increment-based linear search technique and optimize the parameters one by one. We first optimize $\{b_r\}_{r=1}^{R}$ and then α. For both the cases of wide-band (sampling frequency 16 kHz) and narrow-band (sampling frequency 8 kHz) speech, we use a 32 ms Hamming windowed speech frame with 10 ms frame shift. To evaluate the MMFCCs, we use $M = 26$ and $Q = 12$. The power spectrum of each frame is computed using a standard DFT based periodogram technique and the power spectrum is perturbed with i.i.d Gaussian noise at different SNRs ranging from 120 to 130 dB. Using an increment-based linear search, we evaluate the minimum value of the measure of eq. (3.1) and find that a polynomial order of $R = 2$ is sufficient; the values of the polynomial coefficients are $b_1 = 0.1$ and $b_2 = 0.9$. Next we search for the optimum α. For wide-band speech and narrow-band speech, we find the optimum values are $\alpha = 900$ and $\alpha = 1100$, respectively. We note that standard MFCCs use $b_1 = 1$ and $\alpha = 700$ irrespective of the sampling frequency of input speech, choice of the window length and shift, and the feature dimension (Q) and number of channels (M) [1], [16].

3.4 Recognition Results

Table 3.2: Robust word recognition accuracy (in %) of 39-dimensional MFCC and MMFCC features using Aurora 2

Feature	Test Set a				Test Set b				Test Set c	
	set 1	set 2	set 3	set 4	set 1	set 2	set 3	set 4	set 1	set 2
	Subway	Babble	Car	Exhibition	Restaurant	Street	Airport	Train-Station	Subway	Street
	SNR = 20 dB									
MFCC	95.46	96.67	96.12	94.88	96.87	96.28	96.78	96.33	93.28	94.41
MMFCC	96.49	97.49	97.08	96.42	97.73	97.04	97.58	97.04	94.96	95.62
	SNR = 10 dB									
MFCC	85.05	86.49	83.39	81.70	87.69	83.89	87.50	84.45	74.88	76.00
MMFCC	87.07	88.51	87.00	85.25	87.81	86.06	87.77	87.84	79.49	78.36

Using the HTK toolkit, we performed phone and word recognition experiments to compare between MFCC and MMFCC features. The static 12-dimensional MFCC feature set was extracted using the same setup as that used to extract the 12-dimensional MMFCC feature set. Using the standard approach, 39-dimensional feature vectors were evaluated. To the static features, we appended the log energy of a speech frame and the velocity and acceleration of the features.

Table 3.3: Robust phone recognition accuracy (in %) of 39-dimensional MFCC and MMFCC features at 10 dB SNR

Feature	Performance in Accuracy				
	Clean	White	Pink	Babble	Volvo
MFCC	68.11	37.03	40.51	46.25	59.71
MMFCC	68.34	43.65	46.67	48.94	61.91

We first compared the performance of 12-dimensional static features through a clean speech phoneme recognition experiment. In this case, we used the TIMIT database where the speech is sampled at 16 kHz. HTK training and testing were performed using the training set and the test set of TIMIT respectively. The TIMIT transcriptions are based on 61 phones. Following convention, the 61 phones were folded onto 39 phones as described in [17]. To train the HMMs, we used three states per phoneme and the performance is shown in Table 3.1 for a varying number of Gaussian mixtures per state. We used Gaussian mixtures with diagonal covariance matrices. From Table 3.1, it can be noted that MMFCCs outperform MFCCs for any number of mixtures. In case of the 39-dimensional feature vectors, the performance improvement of using MMFCCs over MFCCs was always positive, but small for clean speech phone recognition.

Next we considered the 39-dimensional feature vectors for robust word and phone recognition experiments where clean speech training and noisy speech testing were performed. For the robust recognition, we used cepstrum mean and variance normalization (CMVN) on the feature sets [18]. For the robust word recognition experiment, we used the Aurora 2 database where the speech is sampled at 8 kHz and the sub-datasets of test set are corrupted by different noise types at varying SNRs. The standard configuration of the HTK setup was used where HMMs were trained using 16 states per word and three Gaussian mixtures per state (diagonal covariance matrices). The robust word recognition performance for 39-dimensional MFCCs and MMFCCs are shown in Table 3.2 at the testing conditions of 20 dB and 10 dB SNRs. We note that MMFCCs perform better than MFCCs for all the sub-datasets corrupted with different noises. In the case of clean speech word recognition, the improvement of 39-dimensional MMFCCs over MFCCs was small like in the case of clean speech phone recognition.

Finally, we consider a robust phone recognition experiment where the clean test speech database of TIMIT was corrupted with additive noise. We used the following noise types from the NoiseX-92 database: white, pink, babble and car (volvo) noise. The test speech database was corrupted by adding each noise at 10 dB SNR. The HMMs consisted of three states per phoneme and 20 Gaussian mixtures per state. The robust phone recognition performance for MFCCs and MMFCCs are shown in Table 3.3 and we note that MMFCCs perform better than MFCCs for all noise types.

3.5 Conclusions

Our development of MMFCCs shows that the use of a sophisticated auditory model can lead to a simple feature set that provides improved speech recognition performance for any environmental condition. The success of our perceptual-distance preserving measure in optimizing features suggests that the auditory system provides as output a signal representation that is 'efficient' for speech recognition. As we developed the static MMFCCs using a static spectral auditory model, further investigation should consider the optimization of dynamic features using a spectro-temporal auditory model, such as that presented in [11].

Bibliography

[1] S.B. Davis and P. Mermelstein, “Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. 28, No. 4, pp. 357-366, Aug. 1980.

[2] J.R. Cohen, “Application of an auditory model to speech recognition," *J. Acoust. Soc. Amer.*, pp. 2623-2629, Vol. 85 (6), June 1989.

[3] O. Ghitza, “Auditory models and human performance in tasks related to speech coding and speech recognition," *IEEE Trans. Speech, Audio Proc.,* vol. 2, No. 1, pp. 115-132, Jan 1994.

[4] B. Strope and A. Alwan, “A model of dynamic auditory perception and its application to robust word recognition," *IEEE Trans. Speech, Audio Proc.,* vol. 5, No. 5, pp. 451-464, Sept. 1997.

[5] D.S. Kim, S.Y. Lee and R.M. Kil, “Auditory processing of speech signals for robust speech recognition in real-world noisy environments," *IEEE Trans. Speech, Audio Proc.*, vol. 7, No. 1, pp. 55-69, Jan 1999.

[6] J. Tchorz and B. Kollmeier, “A model of auditory perception as front end for automatic speech recognition," *J. Acoust. Soc. Amer.*, pp. 2040-2050, Vol. 106 (4), Oct. 1999.

[7] M. Holmberg, D. Gelbart and W. Hemmert, “Automatic speech recognition with an adaptation model motivated by auditory processing," *IEEE Trans. Speech, Audio Proc.*, vol. 14, No. 1, pp. 43-49, Jan 2006.

[8] S. Seneff, “A joint synchrony/mean-rate model of auditory processing," *J. Phonet.*, pp. 55-76, Vol. 85 (1), Jan 1988.

[9] R. Meddis, “Simulation of mechanical to neural transduction in the auditory receptor," *J. Acoust. Soc. Amer.*, pp. 702-711, Vol. 79 (3), March 1988.

[10] J.M. Kates, “Two-tone suppression in a cochlear model," *IEEE Trans. Speech, Audio Proc.,* vol. 3, No. 5, pp. 396-406, Sept. 1995.

[11] T. Dau, D. Puschel, and A. Kohlrausch, “A quantitative model of the effective signal processing in the auditory system. I. Model structure," *J. Acoust. Soc. Amer.*, pp. 3615-3622, Vol. 99 (6), Jun 1996.

[12] A.J. Oxenham, “Forward masking: Adaptation or integration?," *J. Acoust. Soc. Amer.*, pp. 732-741, Vol. 109 (2), Feb 2001.

[13] S. van de Par, A. Kohlrausch, R. Heusdens, J. Jensen and S.H. Jensen, “A Perceptual model for sinusoidal audio coding based on spectral integration" *EURASIP J. Applied Signal Proc.*, vol. 9, pp. 1292-1304, 2005.

[14] W.R. Gardner and B.D. Rao, “Theoretical analysis of the high-rate vector quantization of LPC parameters," *IEEE Trans. Speech and Audio Proc.*, vol. 3, No.5, pp. 367-381, Sept 1995.

[15] J.H. Plasberg and W.B. Kleijn, “The sensitivity matrix: using advanced auditory models in speech and audio processing," *IEEE Trans. Audio, Speech, Language Proc.*, vol. 15, No. 1, pp. 310-319, Jan 2007.

[16] J.W. Picone, “Signal modeling techniques in speech recognition," *Proc. IEEE*, pp. 1215-1247, Vol. 81, No. 9, Sept. 1993.

[17] K.F. Lee and H.W. Hon, “Speaker-independent phone recognition using hidden Markov models," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. 37, No. 11, pp. 1641-1648, Nov. 1989.

[18] J. Droppo and A. Acero, “Environmental robustness," *Handbook of Speech Processing,* Springer, pp. 658-659, Oct. 2007.

Chapter 4

Auditory Model Based Modified MFCC Features

S. Chatterjee and W.B. Kleijn (KTH)

4.1 Introduction

Assuming spectral content stationarity over a short segment of speech signal, the MFCCs can be regarded as static features which are computed frame-by-frame independently. To use spectro-temporal properties of speech signal in an ASR system, it is a standard practice to use the temporal dynamic features (such as velocity and acceleration), which are computed from the static MFCCs using a standard regression model. Even though these temporal features help to improve ASR performance, they are ad hoc in nature and do not use the knowledge of spectro-temporal auditory properties. Noting the fact that MFCCs are unable to use the up-to-date knowledge of auditory response across speech frames, we develop a computationally simple feature that uses both spectral and temporal auditory properties.

In our earlier work [11], we developed a perturbation based optimization framework to design a set of optimum static features using a static spectral auditory model, such as the van de Par auditory model (VAM) [10]. The set of optimum features was developed based on the generalization of existing MFCCs and referred to as modified MFCCs (MMFCCs) in [11]. The MMFCCs were shown to use an optimal warping of the frequency scale to design a set of triangular filters and an optimal logarithmic polynomial function to compress the triangular filter bank energies (FBEs). An important point to note that the MMFCCs are static features like MFCCs, and the velocity and acceleration features are computed using a standard regression model like is commonly done with MFCCs.

In this paper, we propose to use the adaptive compression loops (ACLs) of the spectro-temporal auditory model, namely the Dau auditory model (DAM) [8], to incorporate the auditory response across speech frames. We develop a new set of static and adaptive compression based generalized MFCCs (GMFCCs) to achieve further improvement in ASR performance, but without incurring higher computational complexity. In our method, the FBEs are processed through the use of two compression stages: static and adaptive. The static compression stage is memoryless across speech frames and the adaptive compression stage introduces memory across speech frames. Using HMM based toolkit (HTK), the new feature set is shown to perform better than the standard MFCCs and the recently proposed MMFCCs.

The remainder of this article is organized as follows. Section 4.2 provides a brief description of the spectro-temporal auditory model, DAM. The proposed feature set is discussed in section 4.3. Section 4.4 reports the improvement in ASR performance using HTK and the conclusions are presented in section 4.5.

4.2 A Spectro-temporal Auditory Model

The Dau auditory model (DAM) is a spectro-temporal auditory model that quantifies and transforms an incoming sound waveform into its 'internal' representation [8]. This model describes human performance in the typical psycho-acoustical spectral and temporal masking experiments, e.g., predicting the thresholds in backward, simultaneous, and forward-masking experiments.

A block diagram of DAM is shown in Fig. 4.1 [8], [5]. The input speech signal is decomposed into the critical band signals using a gamma-tone filter bank. Then the output signal of each gamma-tone filter is half-wave rectified and first-order low pass filtered (with a cut-off frequency of 1 kHz) for envelope extraction. Therefore, at this processing stage, each frequency channel contains information about the amplitude variation of the input signal within the channel. This envelope is then compressed using an adaptive circuit consisting of five consecutive nonlinear adaptive compression loops (ACLs). Each of these loops consists of a divider and a first order IIR low pass filter (LPF). The time constants of LPFs of five loops are: $\tau_1 = 5$ ms, $\tau_2 = 50$ ms, $\tau_3 = 129$ ms, $\tau_4 = 253$ ms, and $\tau_5 = 500$ ms (the corresponding cut-off frequencies are 32 Hz, 3.2 Hz, 1.23 Hz, 0.62 Hz, and 0.32 Hz, respectively). For each adaptive loop, the input signal is divided by the output signal of the low pass filter. Sudden transitions in a critical band envelope that are very fast compared to the time constants of the ACLs are amplified linearly at the output due to slow changes in the outputs of low pass filters, whereas the slowly changing portions of the envelope are compressed. Due to this transformation characteristic, changes in the input signal like onsets and offsets are emphasized, whereas the steady state portions are compressed. The ACLs introduce inherent memory in the model and help to take into account the dynamic temporal structure of auditory response. The last processing step is a first order LPF with cut-off frequency 8 Hz to optimize predictions of psycho-acoustical masking experiments. This LPF acts as a modulation filter and attenuates fast envelope fluctuations of the signal in each frequency channel.

In [5], the output of DAM was formatted and directly used as a feature in an ASR system. Recently, the use of ACLs of DAM was investigated for deriving modulation based features in [12] where the input speech signal was decomposed into critical bands and then the temporal envelopes of sub-band signals were compressed using both static (logarithmic) and adaptive (ACLs) compressions. The feature set of [12] is computationally intensive to evaluate and also, the feature vector is of high dimension (the dimension is 476 using 17 critical bands and 28 modulation components per sub-band). To design a computationally simple feature set with a moderate dimensionality, we investigate the use of dynamic ACLs followed by LPF for compressing the FBEs of MMFCCs along-with the static compression.

4.3 Static and Adaptive Compression Based Generalized MFCCs

Conventional speech processing techniques use short term speech analysis method where the input speech signal is framed into short segments (typically 20-40 ms) with a reasonable frame shift (typically 10 ms). Most of the speech features, like MFCCs, are derived using the power spectra computed

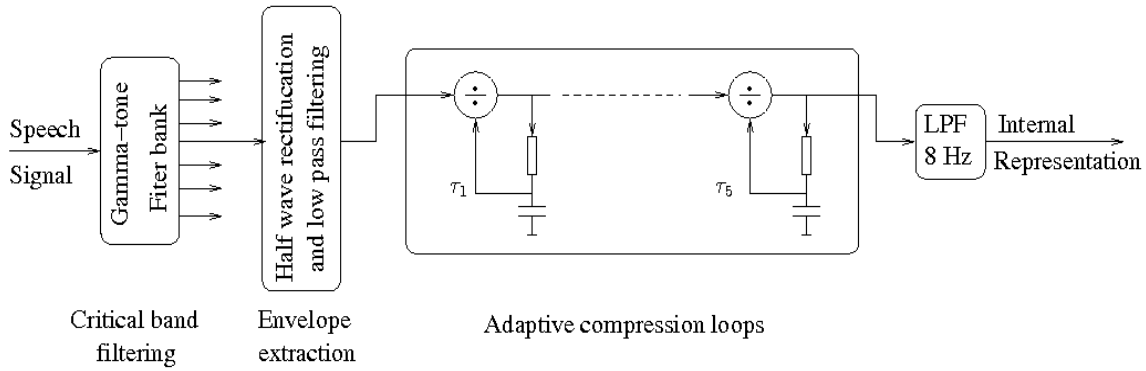

Figure 4.1: A spectro-temporal auditory model: DAM

from the short speech segments. The frame-by-frame based short term analysis along-with the power spectrum computation can be regarded as a time-frequency analysis based method where each power spectrum represents a sample vector of the underlying dynamic process in the production of speech at a given time frame. On the other hand, the sophisticated auditory model, such as DAM, uses alternate framework of frequency-time analysis where the time domain envelope of the output of a gamma-tone filter (frequency channel) is processed. Therefore, it is interesting to see how the ACLs of DAM can be used for the case of time-frequency analysis based feature design.

4.3.1 Feature Extraction

Using the short term speech analysis technique, we develop a new set of features where both static and adaptive compressions are used. The new feature is referred to as generalized MFCCs (GMFCCs). Let the N-dimensional vector $\mathbf{x}_j = [x_{j,0}\ x_{j,1}, \cdots, x_{j,n}, \cdots, x_{j,N-1}]^T$ be the power spectrum of the Hamming windowed j'th speech frame. The GMFCCs consist of two parts: a feature sub-vector derived using a static compression and a feature sub-vector derived using an adaptive compression. The following subsections describe these two parts to construct the full feature vector.

Static Part

For the j'th speech frame, the static part of GMFCCs is nothing but the MMFCCs developed in [11]. The steps of evaluating the static MMFCCs are as follows:

1. Computation of filter bank energies (FBEs): Calculation of the energy in each filter as

$$\begin{aligned} z_{j,m} &= \mathbf{x}_j^T\ \mathbf{w}_m(\alpha) \\ &= \sum_{n=0}^{N-1} x_{j,n} \times w_{m,n}(\alpha)\,, 0 \le m \le M-1, \end{aligned} \tag{4.1}$$

where $\mathbf{w}_m(\alpha)$ is the N-dimensional vector denoting the m'th triangular filter that satisfies $\sum_{n=0}^{N-1} w_{m,n}(\alpha) = 1$. M is the total number of filters with a typical value of $M = 26$. The shape of a triangular filter depends on the extent of frequency warping. The warped frequency scale [13] is given as

$$f_{warp} = 2595 \times \log_{10}\left(1 + (f/\alpha)\right), \tag{4.2}$$

where α is the warping factor and f is the frequency in Hz. An increase in α leads to a decrease in the extent of warping. For the MMFCCs, α was a parameter to optimize to achieve better recognition performance. Using a perturbation based optimization method and a spectral auditory model (VAM), we had found in [11] that $\alpha = 1100$ and 900, respectively for narrowband and wideband speech cases with a frame length of 32 ms. In the case of MFCCs, $\alpha = 700$ [13] irrespective of input speech sampling frequency and frame length.

2. Static compression: Compression of the dynamic range of the energy in each channel as

$$s_{j,m} = \log_{10}\left[\sum_{r=1}^{R} b_r \left(z_{j,m}\right)^r\right],\ 0 \leq m \leq M-1, \tag{4.3}$$

where $\sum_{r=1}^{R} b_r = 1$ and $b_r \geq 0$. For the MMFCCs [11], we optimized the polynomial coefficients $\{b_r\}_{r=1}^{R}$ and found that $R = 2$ is sufficient with polynomial coefficients as $b_1 = 0.1$ and $b_2 = 0.9$. In the case of MFCCs, $R = 1$ and $b_1 = 1$ [1]. We note that eq. (4.3) implies that our results are scale dependent and require proper normalization.

3. De-correlation using the DCT to evaluate Q-dimensional static MMFCCs part:

$$c_{j,q} = \sum_{m=0}^{M-1} s_{j,m} \times \cos\left[q\left(m+0.5\right)\frac{\pi}{M}\right], 1 \leq q \leq Q. \tag{4.4}$$

A typical value of static feature vector dimension is $Q = 12$.

Therefore the static MMFCCs of j'th speech frame is $\mathbf{c}_j = [c_{j,1}\ c_{j,2}, \cdots, c_{j,Q}]^T$ which is used as the static part of GMFCCs.

Adaptive Part

We use ACLs to adaptively compress the FBEs across the speech frames. The FBEs are already computed in the evaluation of static part. For the m'th triangular filter, the FBE signal $z_{j,m}$ can be viewed as a time domain signal where the index j denotes the time variable. In case of a typical frame shift of 10 ms, the $z_{j,m}$ signal is of the rate of 100 Hz. The time domain $z_{j,m}$ signal, $0 \leq m \leq M-1$, is passed through the ACLs to incorporate the auditory response across speech frames into the new features. For the j'th frame, the steps to compute the adaptive part of GMFCCs are as follows:

1. Adaptive compression: For m'th triangular filter, compression of $z_{j,m}$ signal as

$$u_{j,m} = acl(z_{j,m}^{\kappa}),\ 0 \leq m \leq M-1, \tag{4.5}$$

where $acl(.)$ is the model function of ACLs as used in DAM and κ is introduced to make the argument representation close to the envelope of the output of the corresponding gamma-tone filter of DAM, albeit at a much lower rate of 100 Hz.

2. Modulation filtering: For the m'th triangular filter, the signal $u_{j,m}$ is passed through a first order IIR low pass filter as used in DAM.

3. De-correlation using the DCT to evaluate $\mathcal{Q}$-dimensional adaptive part of the feature vector:

$$v_{j,q} = \sum_{m=0}^{M-1} u_{j,m} \times \cos\left[q\left(m+0.5\right)\frac{\pi}{M}\right], 1 \leq q \leq \mathcal{Q}. \tag{4.6}$$

Table 4.1: Phone recognition accuracy (in %) using TIMIT

Feature	Clean	Noise Types; SNR=10 dB			
		White	Pink	Babble	Volvo
MFCC	68.11	37.03	40.51	46.25	59.71
MMFCC	68.34	43.65	46.67	48.94	61.91
GMFCC	69.10	44.51	47.53	49.06	63.48

Like the static part, the DCT is applied across the triangular filters (frequency channels) and we choose the dimension of adaptive part as $\mathcal{Q} = 12$.

Therefore, the sub-vector $\mathbf{v}_j = [v_{j,1}\ v_{j,2}, \cdots, v_{j,\mathcal{Q}}]^T$ is used as the adaptive part of GMFCCs. Through a series of ASR experiments, we choose $\kappa = 0.5$ and the modulation LPF filter cut-off frequency as $f_c = 4$ Hz. Also, for the ACLs, we keep all the time constants of the LPFs same as prescribed in DAM, except the first LPF; for the first LPF, we choose $\tau_1 = 20$ ms (cut-off frequency of 7.95 Hz)[1]. Note that the $acl(.)$ function has long time constants. It means that the current processing depends on its long memory. We typically address the issue by starting analysis as far as possible in the silence region preceeding the speech signal.

Full Feature Vector

Following the standard approach, we append the logarithmic energy of speech frame to the static features, and then compute the velocity and acceleration of the features. Then, we append the adaptive features to make the full feature vector of GMFCCs. For the typical static feature dimension of $Q = 12$ and adaptive feature dimension $\mathcal{Q} = 12$, the total dimension of GMFCCs is $(3 \times (12+1)) + 12 = 51$.

4.3.2 Examples of compression

The effects of static logarithmic polynomial compression and adaptive ACLs' compression on FBEs are illustrated in Fig. 4.2. A portion of 1 sec speech signal of 16 kHz sampling frequency is shown in Fig. 4.2 (a). Using 32 ms frame length and 10 ms frame shift, the trajectory of FBEs for sixth triangular filter over the frame indices is shown in Fig. 4.2 (b). Fig. 4.2 (c) shows the results of static compression (memoryless) and adaptive compression (with memory) on FBEs in a normalized scale. It is seen that the use of ACLs emphasizes the onsets and offsets of FBEs.

4.4 Recognition Results

Using HTK, we performed phone and word recognition experiments to compare the new GMFCCs with the recently proposed MMFCC and standard MFCC features. For both experiments, we use a 32 ms Hamming windowed speech frame with 10 ms frame shift. To evaluate the features, the power spectrum of each frame is computed using a standard DFT based periodogram technique.

[1]We note that the adaptive part of GMFCCs is a function of input FBEs and some adjustable design parameters. For example, the design parameters can be κ, f_c, and $\{\tau_i\}_{i=1}^{5}$. Like our earlier work on the optimization of adjustable parameters of the static part [11], we are currently working on developing a framework where the adjustable parameters of the adaptive part are optimized without any feedback from an ASR system.

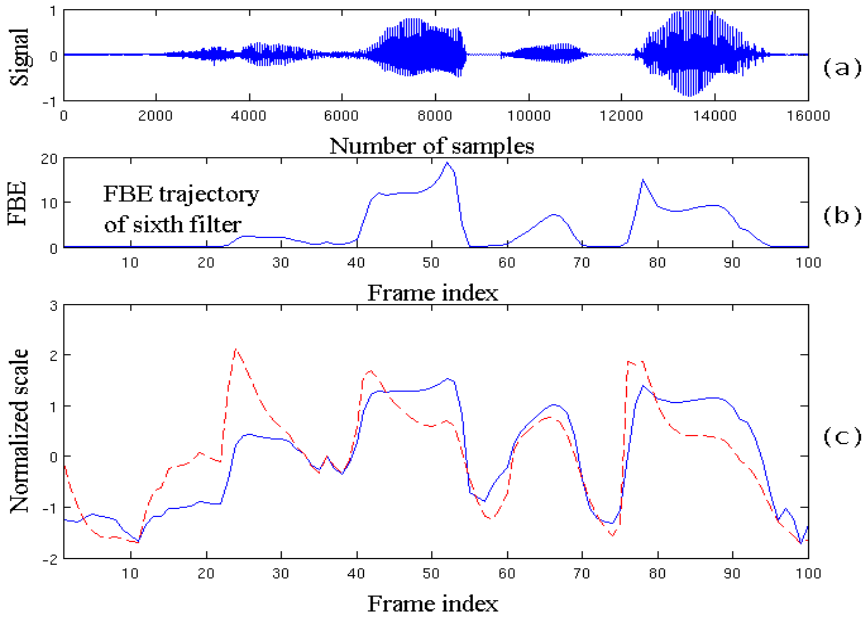

Figure 4.2: (a) A portion of 1 sec speech signal with 16 kHz sampling frequency. (b) The trajectory of FBEs for sixth triangular filter. (c) Outputs of static and adaptive compressions in a normalized scale.

For the cases of MFCCs and MMFCCs, 39-dimensional feature vectors were evaluated using the standard approach. To the 12-dimensional static features, we appended the log energy of a speech frame and the velocity and acceleration of the features to construct the 39-dimensional MFCC and MMFCC features. The details of constructing 51-dimensional GMFCCs are already explained in section 4.3.1 which is nothing but appending the 12-dimensional adaptive feature part derived from ACLs with the 39-dimensional MMFCCs. For robust recognition, we used cepstrum mean and variance normalization (CMVN) on the 39-dimensional MFCCs and MMFCCs feature sets [15]. On the other hand, the 39-dimensional MMFCCs part of GMFCCs was normalized using CMVN, but the 12-dimensional adaptive part is kept unnormalized. It was experimentally found that the normalization of 12-dimensional adaptive part could not improve the recognition performances, even resulted in a deterioration in some cases[2].

We first considered phone recognition experiment using the TIMIT database where the speech is sampled at 16 kHz. The TIMIT transcriptions are based on 61 phones. Following convention, the 61 phones were folded onto 39 phones as described in [14]. HTK training and testing were performed using the training set and the test set of TIMIT respectively. To train the HMMs, we used three states per phone and 20 Gaussian mixtures per state (with diagonal covariance matrices). For robust phone recognition experiment, the clean test speech database of TIMIT was corrupted with additive

[2]It is a standard practice to normalize the feature vectors (like MFCCs) and their temporal dynamic parts (velocity and acceleration) [15]. Most of the features are static in nature which are computed independently across the speech frames and the dynamic parts are computed using a regression method which acts as a FIR filter with limited memory across the speech frames. In this context, it is worth to investigate in future the issue of normalizing the adaptive part which is evaluated by using the ACLs. Note that the ACLs has a considerable duration of memory across the speech frames.

Table 4.2: Word (TIDIGITS) recognition accuracy (in %) using Aurora 2

Feature	Test Set a				Test Set b				Test Set c		Average Performance
	set 1	set 2	set 3	set 4	set 1	set 2	set 3	set 4	set 1	set 2	
	Subway	Babble	Car	Exhibition	Restaurant	Street	Airport	Train-Station	Subway	Street	
	Clean Speech										
MFCC	99.05	98.91	98.99	99.11	99.05	98.91	98.99	99.11	98.99	98.70	98.98
MMFCC	99.26	98.97	99.05	99.29	99.26	98.97	99.05	99.29	99.26	98.97	99.13
GMFCC	99.42	99.15	99.28	99.38	99.42	99.15	99.28	99.38	99.29	99.15	99.29
	SNR = 20 dB										
MFCC	95.46	96.67	96.12	94.88	96.87	96.28	96.78	96.33	93.28	94.41	95.70
MMFCC	96.59	97.49	97.08	96.42	97.73	97.04	97.58	97.04	94.96	95.62	96.75
GMFCC	97.67	98.34	97.55	97.47	98.16	98.00	98.27	98.03	96.32	96.89	97.67
	SNR = 10 dB										
MFCC	85.05	86.49	83.39	81.70	87.69	83.89	87.50	84.45	74.88	76.00	83.10
MMFCC	87.07	88.51	87.00	85.25	87.81	86.06	87.77	87.84	79.49	78.36	85.51
GMFCC	89.44	90.18	89.47	88.71	90.36	89.24	89.83	90.00	83.24	82.98	88.34

noise. We used the following noise types from the NoiseX-92 database: white, pink, babble and car (volvo) noise. The test speech database was corrupted by adding each noise at 10 dB SNR. The phone recognition performance for the features are shown in Table 4.1 and we note that GMFCCs perform better than other features for clean as well as all noise types.

Next we performed word recognition experiment. In this case, we used the Aurora 2 database where the speech is sampled at 8 kHz and the sub-datasets of test set are corrupted by different noise types at varying SNRs. The standard configuration of the HTK setup was used where HMMs were trained using 16 states per word and three Gaussian mixtures per state. The word recognition performance for the features are shown in Table 4.2 at the testing conditions of clean and varying SNR conditions. We note that GMFCCs perform better than MFCCs and MMFCCs for all the sub-datasets in clean condition as well as noisy condition.

4.5 Conclusions

Our development of GMFCCs shows that the judicious use of sophisticated auditory models can lead to a simple feature set that provides improved speech recognition performance for any environmental condition. The use of ACLs is shown to provide complimentary information across the speech frames, assisting ASR task. Further investigation should consider the incorporation of ACLs (with optimized parameters) into several other time-frequency analysis based existing and new features.

4.6 Acknowledgement

The authors would like to thank the Medical Physics group at the Carl von Ossietzky - Universitat Oldenberg for code fragments implementing adaptive compression loops.

Bibliography

[1] S.B. Davis and P. Mermelstein, “Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. 28, No. 4, pp. 357-366, Aug. 1980.

[2] J.R. Cohen, “Application of an auditory model to speech recognition," *J. Acoust. Soc. Amer.*, pp. 2623-2629, Vol. 85 (6), June 1989.

[3] O. Ghitza, “Auditory models and human performance in tasks related to speech coding and speech recognition," *IEEE Trans. Speech, Audio Proc.,* vol. 2, No. 1, pp. 115-132, Jan 1994.

[4] D.S. Kim, S.Y. Lee and R.M. Kil, “Auditory processing of speech signals for robust speech recognition in real-world noisy environments," *IEEE Trans. Speech, Audio Proc.*, vol. 7, No. 1, pp. 55-69, Jan 1999.

[5] J. Tchorz and B. Kollmeier, “A model of auditory perception as front end for automatic speech recognition," *J. Acoust. Soc. Amer.*, pp. 2040-2050, Vol. 106 (4), Oct. 1999.

[6] M. Holmberg, D. Gelbart and W. Hemmert, “Automatic speech recognition with an adaptation model motivated by auditory processing," *IEEE Trans. Speech, Audio Proc.*, vol. 14, No. 1, pp. 43-49, Jan 2006.

[7] J.M. Kates, “Two-tone suppression in a cochlear model," *IEEE Trans. Speech, Audio Proc.,* vol. 3, No. 5, pp. 396-406, Sept. 1995.

[8] T. Dau, D. Puschel, and A. Kohlrausch, “A quantitative model of the effective signal processing in the auditory system. I. Model structure," *J. Acoust. Soc. Amer.*, pp. 3615-3622, Vol. 99 (6), Jun 1996.

[9] A.J. Oxenham, “Forward masking: Adaptation or integration?," *J. Acoust. Soc. Amer.*, pp. 732-741, Vol. 109 (2), Feb 2001.

[10] S. van de Par, A. Kohlrausch, R. Heusdens, J. Jensen and S.H. Jensen, “A Perceptual model for sinusoidal audio coding based on spectral integration" *EURASIP J. Applied Signal Proc.*, vol. 9, pp. 1292-1304, 2005.

[11] S. Chatterjee, C. Koniaris and W.B. Kleijn, “Auditory model based optimization of MFCCs improves automatic speech recognition performance," *Proc. INTERSPEECH*, pp. 2987-2990, Sept 2009, UK.

[12] S. Ganapathy, S. Thomas and H. Hermansky, “Modulation freequency features for phone recognition in noisy speech," *J. Acoust. Soc. Amer.*, vol. 125, Issue 1, pp. EL8-EL12, Jan 2009.

[13] J.W. Picone, “Signal modeling techniques in speech recognition," *Proc. IEEE*, pp. 1215-1247, Vol. 81, No. 9, Sept. 1993.

[14] K.F. Lee and H.W. Hon, “Speaker-independent phone recognition using hidden Markov models," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. 37, No. 11, pp. 1641-1648, Nov. 1989.

[15] J. Droppo and A. Acero, “Environmental robustness," *Handbook of Speech Processing,* Springer, pp. 658-659, Oct. 2007.

Chapter 5

Unsupervised learning of time-frequency patches as a noise-robust representation of speech

M.V. Segbroeck and H. Van hamme (KUL)

5.1 Introduction

It is remarkable how babies exposed to a language acquire it naturally without deliberate efforts of teaching or learning. Before they can even speak, infants gather immense amounts of information while listening to human voices in their surroundings. During this stage, their brains are being tuned to a specific language. With only a surprisingly small amount of supervision, they succeed in learning new words from a spoken language. Most words are not being explained to them, but are learned from their significance in the world they live in. Although these streams of language data are large and appear to be unsegmented, infants show the ability to distinguish the different units of the language and to acquire how these units are linked in meaningful patterns, such as words.

Evidence exists [1] that the basic ability to acquire language is innate to the child, e.g. the basics of human *speech* is built into babies' brains. However, no specific structural property of *language* has yet been proven to be innate and any infant seems equally capable of acquiring any language. Future research still has to reveal what in human language is inborn into the infant's brain and how they succeed in learning the language through experience and exposure to a specific speech community. Moreover, during their lifespan, humans are exposed to variations of what is being uttered. These variations can be acoustical (such as different speaking styles, accents or speech distortions caused by e.g. background noises) as well as on the level of interpretation in the context used. Nevertheless, humans have the ability to continuously learn and adapt to these new situations.

Although current systems for automatic speech recognition (ASR) show to be successful in some aspects, their performance can only be guaranteed if these systems are task-specifically programmed and adjusted to the (predicted) acoustic challenges in which they will operate. This way, ASR systems are unable to adapt to situations different from the one seen during training and are mostly unreliable in real life situations.

Our work is motivated by the idea that engineering approaches have fallen short in the design of ASR-systems and that inspiration has to come from human language learning and speech perception,

an idea that was also postulated in other research work [2, 3]. This paper does not claim to explain human language learning, neither does it have the intention to learn grammar, world knowledge or pragmatics. However, we will show that a small vocabulary can be learned from scratch using a bottom-up approach from a spectral analysis of speech signals. To this end, we intend to build a system that automatically discovers the structure in the data, learns the patterns, links them with the words of a vocabulary and finally recognizes them in unseen (noisy) speech data. Our work is related to previously reported approaches of unsupervised language learning [4, 5, 6, 7, 8]. However, in these approaches the units are phones, phonemes or sub-word items, while in this work we search for recurring acoustic patterns in the time-frequency plane. Moreover, instead of acquiring the words of the language by concatenating these units, we assume that words can be represented by a sparse combination of these patterns and can be learned by discovering similarities in the activations of these patterns.

A first step in language acquisition is to build representations of speech that are to a great extent speaker independent and robust to noise. The first part of the paper explains how recurring acoustic patterns are discovered in speech data without supervision, a problem that was also addressed in a variety of other research work, see e.g. [9, 10, 11, 12]. The learning algorithm involved makes use of non-negative matrix factorization (NMF) introduced by [13]. Thanks to the non-negativity constraints, NMF decomposes a matrix in additive (not subtractive) components, resulting in a parts-based representation of the data. NMF can therefore be seen as a learning algorithm that, when applied to an appropriate feature space, finds the parts or objects that the training data are built of.

We will apply NMF to magnitude spectrograms in order to discover typical patterns in the time-frequency plane (the parts) that can be combined additively to form spectrograms of speech. We will consider spectral analyses over longer time windows than the centisecond scale usually considered in automatic speech recognition. Instead, the spectral patterns that are found have a duration in the order of hundreds of milliseconds. Other researchers have also observed that speech features spanning a longer time interval such as TRAPs [14] and its variants show improved robustness to noise [15]. Other examples of long-span features are MRASTA filtering [16] or modulation spectra [17, 18]. Some work also explicitly looks at time-frequency representations [12, 19, 20]. An important difference with the current work is that our time-frequency representation results from a parts-based representation that is learned without supervision. Other authors have used NMF for this purpose. Our approach is most closely related to that of [21], who additionally imposes temporal continuity. Convolutional NMF by [11] or the variant by [22] can also be used to find speech patterns in the time-frequency plane. The discovered speech units seem to be best described as phones, while our units are best described as "acoustic events", such as bursts or formant trajectories. The current work differs in other respects. Firstly, the patterns are discovered from a combination of two complementary feature representations that either reveal timing or frequency structure and which are derived from a time-frequency reassignment spectrogram. Subsequently, these spectral features are segmented into two-dimensional overlapping time-slices which are stacked into column vectors. Recurring time-frequency dependent patterns and bases are then found by applying NMF on these vectors. By enforcing sparsity constraints in NMF, both timing and frequency information are modeled by the obtained bases. Secondly, we use conventional NMF instead of convolutive NMF (cNMF). Although the convolutive variant is appealing from a theoretical point of view, we have found from analyses of parts based on cNMF that it is less resistant to noise. Also, the computational requirements are significantly higher for cNMF. Thirdly, we also add a pattern recognition step to show speech recognition based on the discovered time-frequency patterns and demonstrate the robustness to noise thus obtained.

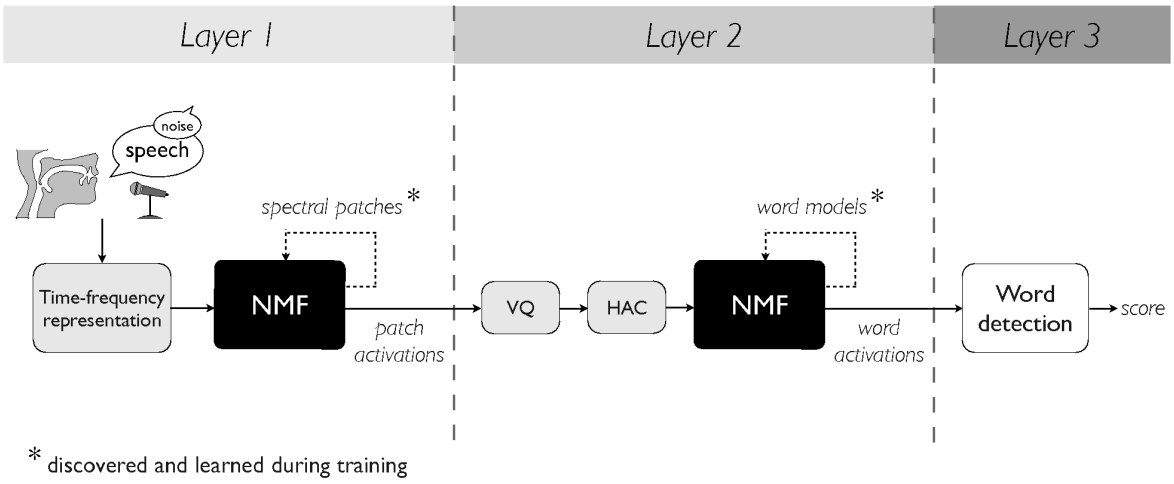

Figure 5.1: Structural representation of the proposed learning system.

The bases are acoustic patterns and they will also be referred to as *time-frequency patches* of speech. From a neuroscience point of view, we could relate the process of discovering and acquiring these patches with the learning and/or evolutionary process by which humans have developed an auditory system that is exceedingly sensitive to speech sounds, though we do not claim that what we present here is a validated model of the neurophysiological mechanism. By describing speech in terms of these patches, we show how meaningful objects of the language such as words are linked to patch activation patterns. These patterns appear to be unrelated to speaker-specific properties and remain clearly visible when noise is added to the speech signals. Similarly to our auditory system, the proposed model seems to be skilled in easily distinguishing speech from other environmental sounds, without the use of noise models or noise reduction techniques.

In the second part of this paper, our aim is to let a computer system acquire the vocabulary of a language by detecting, segmenting and learning the recurring activation patterns from the continuous stream of speech. To this end, the above mentioned speech model will be exploited in a language learning system. Similar to human speech recognition, the proposed system is able to acquire a language on clean training speech using weak supervision without knowing the words until after the acquisition process is completed. Key to the success of the system is the ability to discover recurring patterns in the activations of the *time-frequency patches* in speech across time. Therefore, the speech data is transformed into a high-dimensional vector representation, called "histograms of acoustic co-occurrence" (HAC) which are computed by accumulating the co-occurrence counts of acoustic events [23]. For this work, these events are quantized patch activation vectors. Subsequently, a learning algorithm with weak supervision and which is again based on non-negative matrix factorization (NMF) is proposed to discover recurring patterns through the use of HAC-features and link them with the lexical items of a language. Hence, in contrast to hidden Markov model (HMM) based speech recognition systems, no expert knowledge from audiology or phonology is incorporated in our system, neither do we need any a-priori information about what the words are and how they are composed. After the learning process, the system shows a remarkably good performance in detecting the words of the language in both clean and noisy speech data. Hence, the learning system could form the basis of an alternative framework for robust speech recognition.

Figure 5.1 shows the structure of the proposed system. In the learning stage, the NMF in the

first layer is performed on the time-frequency representation to acquire time-frequency patches in clean training speech. From these learned patches, the patch activations along the time axis are then computed. In the second layer of the system, the patch activation vectors are quantized and transformed to the HAC representation. On these HAC-features, another NMF is performed to learn the HAC of the vocabulary words. During testing, the first NMF computes the patch activations from the learned time-frequency patches and the second NMF computes the word activations from the learned HAC-models to detect the words in the utterance. To assess the efficacy of the language learning, a third layer was added to the system to detect the words from the word activations on unseen speech data.

The outline of this chapter is as follows: section 5.2 explains how the time-frequency patches of the first layer are learned from speech and how their activations along the time axis are computed. In section 5.3, the concept of HAC-models is restated from which the HAC of the words are learned in the second layer. Section 5.4 describes how this learning system can be applied to detect words in speech utterances. A small vocabulary word discovery experiment was conducted on the Aurora2 digit database and experimental results are given in section 5.5. Finally, conclusions can be found in section 5.6.

5.2 Layer 1: Time-frequency patch discovery

The production of human speech can be regarded as a process of combining a small number of spectral patterns into many more different sequences. In this section, our goal is to find a set of patterns by analyzing continuous speech recordings. Moreover, we would like that (i) these patterns accurately represent a wide variety of acoustic speech events with well localized energy regions to model e.g. formant tracks or energy bursts due to plosives, and in a later stage (ii) that they can be robustly detected in unseen noisy speech. To this end, we transform the speech data into a time-frequency reassignment spectrogram [24] which is subsequently smoothed in time and frequency domain. The reassignment method (briefly restated in section 5.2.1) produces sharpened time and frequency estimates for each spectral component from partial derivatives of the short-time phase spectrum. Instead of locating the spectral density value at the geometrical center of the analysis window, as in traditional short-time spectral analysis (e.g. STFT), the components are reassigned to the center of gravity of the energy distribution. In a next step, we perform a non-negative matrix factorization (NMF) on a data matrix containing consecutive spectral slices of this speech representation. By imposing non-negativity constraints, NMF generates parts that are additive, unlike factorization techniques such as PCA or SVD. Furthermore, NMF was chosen among other unsupervised learning method such as hierarchical clustering, self-organizing maps or neural networks, since (i) it is a recent and promising technique that has shown it merits in other research questions; (ii) it provides a more stable, intuitive and meaningful decomposition of non-negative data. By combining two complementary time-frequency reassignment representations that either reveal timing or frequency structure, the discovered speech patterns are acoustic patches of correlated energy that are well localized in both time and frequency. The involved smoothing process allows to make these patches speaker-independent. Besides the fact that the reassignment method produces highly detailed patterns, another motivation to use RTFR is the impact it has on the higher level of word learning (section 5.3). This will be illustrated in the experiments of section 5.5 by comparing the approach of RTFR with the approach where the patches are derived from the short time Fourier transform (STFT) of the speech.

5.2.1 Time-frequency reassignment

Time-frequency reassignment [24, 25, 26] allows perfect localization of (well-separated) impulses, cosines and chirps, which constitute a reasonable model for speech. The corresponding reassigned time-frequency representation (RTFR) has an increased sharpness of localization of the signal components without sacrificing the frequency resolution.

In this paper, the reassignment principle is applied to the short time Fourier transform (STFT) although it can be applied to different time-frequency representations each characterized by a different analysis kernel. The STFT is often used as the basis for a time-frequency representation of speech signals and is written as

$$\text{STFT}\{x(t)\} = \int_{-\infty}^{+\infty} x(u)h^*(t-u)\text{e}^{-j\omega u}\text{d}u \tag{5.1}$$

where $x(t)$ is the analyzed signal and $h(t)$ is the analysis kernel function. The spectrogram is then defined as the magnitude of the STFT and can also be expressed as a two-dimensional smoothing of the Wigner-Ville distribution [24]

$$|\text{STFT}\{x(t)\}|^2 = \frac{1}{2\pi}\int\int_{-\infty}^{+\infty} W_x(u,\nu)W_h(t-u,\omega-\nu)\text{d}u\text{d}\nu \tag{5.2}$$

with

$$W_x(t,f) = \int_{-\infty}^{+\infty} x(t+\frac{1}{\tau})x^*(t-\frac{1}{\tau})\text{e}^{-j\omega\tau}\text{d}\tau. \tag{5.3}$$

From expression (5.2) it can be seen that the spectral density value of each time-frequency component is the weighted sum of all the Wigner-Ville distribution values at the points $(t-u,\omega-\nu)$ and thus located at the geometrical center (t,ω) of the spectral analysis kernel function. The principle of the reassignment method is then to reallocated the energy from the geometric center of the kernel function to the center of gravity of the energy distribution. Therefore, the RTFR takes into account the phase of the STFT, which is omitted in the classical spectrogram, but contains important temporal information and this results in an improved localization of the energy in the time-frequency plane.

The reassignment points can be computed from the partial derivatives of the phase of the STFT using the principle of stationary phase [27]. According to this principle, the maximal contribution to the values of (5.2) occurs at the points where the phase is changing most slowly with respect to time and frequency. If $\phi(t,\omega)$ denotes the short-time phase spectrum, then these points are computed as [25]

$$\left(\hat{t},\hat{\omega}\right) = \left(t - \frac{1}{2\pi}\frac{\partial}{\partial\omega}\phi(t,\omega),\ \omega + \frac{1}{2\pi}\frac{\partial}{\partial t}\phi(t,\omega)\right) \tag{5.4}$$

which represents the group delay and instantaneous frequency of the windowed signal. It has been shown in [24], that a more efficient implementation is possible using two additional STFTs rather than using the derivatives of the phase. Let $H(t,\omega)$, $D(t,\omega)$ and $T(t,\omega)$ denote the STFT of the signal obtained with the window of choice $h(t)$, the derivative of $h(t)$ and the time weighted $th(t)$ respectively and let $\Re(X)$ and $\Im(X)$ be the real and imaginary part of X, then the energy at (t,ω) is reassigned to the center of gravity [24]:

$$\left(\hat{t},\hat{\omega}\right) = \left(t - \Re\left[\frac{T(t,\omega)}{H(t,\omega)}\right],\ \omega + \Im\left[\frac{D(t,\omega)}{H(t,\omega)}\right]\right) \tag{5.5}$$

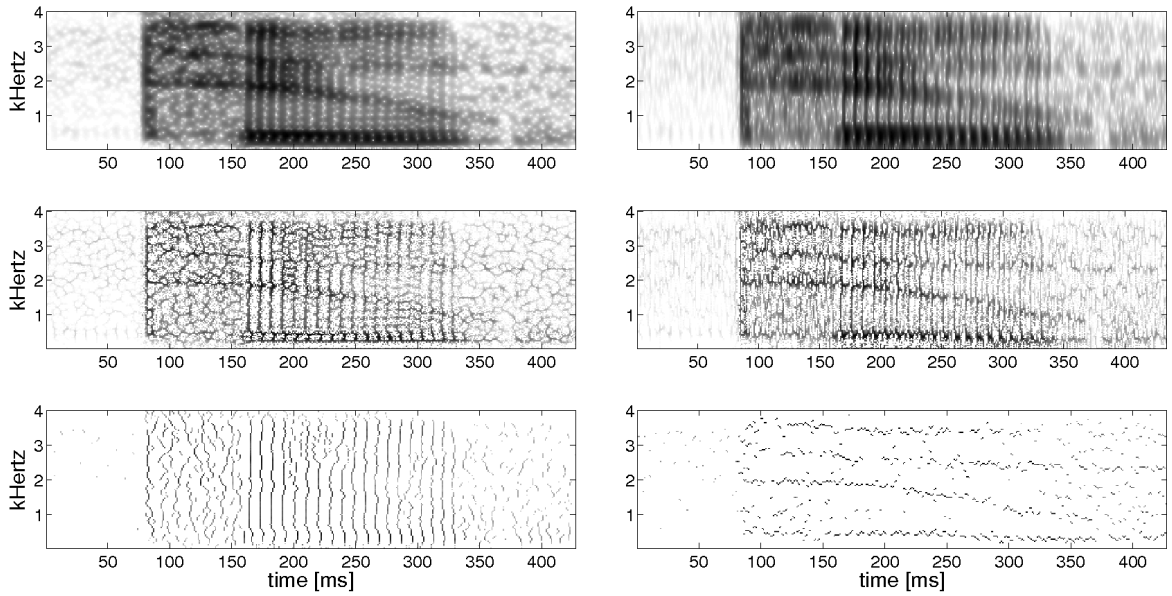

Figure 5.2: STFT representation (top), RTFR (middle) and enhanced RTFR (bottom) for the word "two" with focus on timing (left) and frequency (right) structure. In the bottom left panel, vertical lines correspond to plosive burst and vocal fold bursts, while the enhanced RTFR at the bottom right reveals the horizontal structure in the word, e.g. pitch and formant contours.

where the time and frequency offsets are now computed from the ratios of the three STFTs.

To improve the visibility of acoustic events with short duration, we further enhance the localization of the energy along the time axis. Therefore, we first search for zero-crossing points in the time offsets of (5.5) and only those of them that are connected in the vertical direction (i.e. along the frequency axis) are retained. Finally, the corresponding energy of the RTFR is assigned to the retained points. When applied to speech with a sufficiently short analysis window, the enhanced RTFR clearly shows the vertical (i.e. well-localized in time) lines that are related to the burst of plosives and affricatives and energy releases by the vocal folds. By repeating the same procedure using the frequency offsets of (5.5), the tracks of time-varying spectral features such as pitch and formants can be clearly localized in frequency. We have found that formant structure is more apparent if we use shorter windows in the RTFR.

The different steps of the enhanced reassignment procedure are shown figure 5.2 for the word "two". Firstly, a time-frequency representation is computed using a 128-point STFT. Subsequently, a RTFR is produced by reallocating the spectral energy to the gravity centers according to (5.5). The above mentioned enhancement steps are then applied to the RTFR to reveal either the timing or the frequency structure. Experiments have shown that an optimal choice for the window length is respectively 11 ms and 7 ms for male speakers and 6 ms and 4 ms for female speakers. The analysis window is shifted by 1 ms. To prevent ambiguity in later formulations, we will use the word *subframe* to denote a frame of the enhanced RTFR.

5.2.2 Constructing the input matrix

In this section, we explain how the input matrix is created to which we will apply NMF for finding acoustic time-frequency patterns in speech signals. These patterns are discovered on clean training data. After pre-emphasizing the speech signals, we compute the enhanced RTFRs by the approach described in 5.2.1. Both representations are used to exploit the spectral information that is more apparent in either the vertical or the horizontal direction.

Two additional steps are also performed; a smoothing in time and frequency followed by a cube root compression. If these steps are not be applied, speaker-dependent "bases" will be learned to model the different pitch characteristics of training speakers. For reasons that become clear in the next section, however, we want to prevent overfitting to the training set, e.g. we want the resulting time-frequency patches not to be speaker dependent. Time smoothing is performed by reframing the enhanced RTFR by a sliding triangular window with a length of 30 subframes and a frameshift of 10 subframes. After conversion of the frequency axis from the Hertz scale to the Mel scale followed by a frequency smoothing using $N = 128$ triangular overlapping windows with a window size of 3 frequency bins using a weight of 1 for the center bin and 0.5 for the adjacent bins, we obtain the final N-dimensional feature vectors. Subsequently, spectral changes are emphasized by adding first and second order derivatives resulting in a static (S), a velocity (V) and an acceleration (A) stream. For these feature representations we use the word *frames* and these are shown in figure 5.3(a) for the same uttered "two" as was used in figure 5.2. Note that the vertical lines corresponding to pitch bursts are dissolved by the smoothing process, but the overall energy bursts and releases are retained. In figure 5.3(b) we also show the feature vectors derived by applying the STFT using the same time window parameters as in the enhanced and smoothed RTFR. Note that, despite the smoothing process, formant contours still remain clear in figure 5.3(a) and are not confused with pitch harmonics as is the case in conventional Fourier transform. In section 5.5, we will compare the final accuracy results obtained by the STFT features with those of the enhanced and smoothed RTFR features.

The feature representations corresponding to the timing and frequency structure contain complementary information and therefore both will be used in the discovery of the speech patches. Let us now define the spectral vector at a certain frame t for a feature stream ρ ($\rho = S$, V or A) as $\boldsymbol{v}_{\rho,t}$ and $\boldsymbol{h}_{\rho,t}$, corresponding to the feature representation that reveals respectively the timing (vertical) and frequency (horizontal) structure. Since NMF requires the data to be comprised of non-negative values only, we split these vectors into a positive ($\boldsymbol{v}^+$ and $\boldsymbol{h}^+$) and a negative ($\boldsymbol{v}^-$ and $\boldsymbol{h}^-$) stream by zeroing out those values that are respectively < 0 and > 0 and taking absolute value of the negative stream. Finally, we stack all these vectors in one real and non-negative column vector of dimension $4N$:

$$\boldsymbol{c}_{\rho,t} = \begin{bmatrix} \boldsymbol{v}^+_{\rho,t} \\ |\boldsymbol{v}^-_{\rho,t}| \\ \boldsymbol{h}^+_{\rho,t} \\ |\boldsymbol{h}^-_{\rho,t}| \end{bmatrix} \tag{5.6}$$

For static features, $\boldsymbol{v}^-_{\rho,t}$ and $\boldsymbol{h}^-_{\rho,t}$ are all-zero vectors and their rows can be removed from (5.6). Note that the above mentioned procedure to handle input data with mixed sign in NMF, can also be seen as an alternative for the semi-NMF as was proposed by [28].

At each frame step t, we take k consecutive frames of $\boldsymbol{c}_{\rho,t}$ representing the spectro-temporal structure of a short-time speech segment (i.e. of length $10k$ ms). These k frames are then reshaped into a column vector $\mathbf{C}_t^{k,\rho}$ of dimension $4kN$ as schematically illustrated in figure 5.4. From these

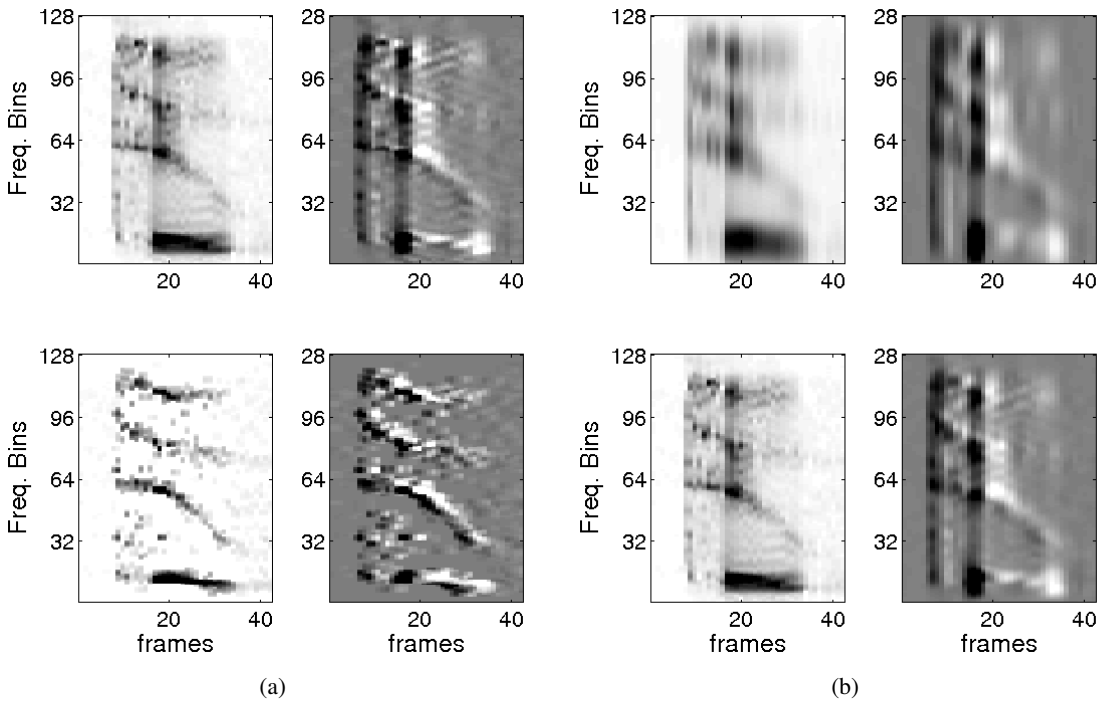

Figure 5.3: (a) Static and velocity feature vectors derived from the enhanced RTFRs that reveal timing (top) and frequency (bottom) structure for the word "two". (b) The corresponding STFT feature vectors using the same time window parameters.

column vectors, we construct a data matrix:

$$\mathbf{C}^k = \begin{bmatrix} \mathbf{C}_1^{k,S} & \dots & \mathbf{C}_t^{k,S} & \dots & \mathbf{C}_T^{k,S} \\ \mathbf{C}_1^{k,V} & \dots & \mathbf{C}_t^{k,V} & \dots & \mathbf{C}_T^{k,V} \\ \mathbf{C}_1^{k,A} & \dots & \mathbf{C}_t^{k,A} & \dots & \mathbf{C}_T^{k,A} \end{bmatrix} \tag{5.7}$$

with T the total number of frames used from the clean training set.

5.2.3 Matrix factorization for unsupervised learning

By applying non-negative matrix factorization to the matrix $\mathbf{C}$ (dropping index k for notational convenience), it is approximated by the product of factors $\mathbf{B}$ and $\mathbf{A}$ which are of size $4kN \times P$ and $P \times T$:

$$\mathbf{C} \approx \mathbf{BA} \tag{5.8}$$

subject to the constraint that all matrices are non-negative and where the common dimension P of $\mathbf{B}$ and $\mathbf{A}$ is much smaller than T and $4kN$. Hence, equation (5.8) contains only additive linear combinations such that the factorization leads to a parts-based representation, where P parts are found in the columns of $\mathbf{B}$ and their activation across time are given by the corresponding rows of $\mathbf{A}$.

In order to capture all feature streams in each basic vector, namely the timing and frequency spectral structure and their corresponding positive and negative part, additional sparsity constraints

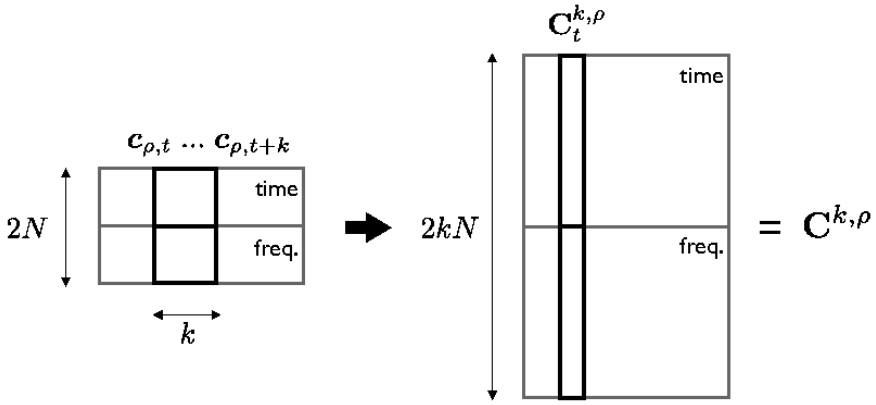

Figure 5.4: Schematic representation of the construction of data matrix $\mathbf{C}^{k,\rho}$ from feature vectors $\boldsymbol{c}_{\rho,t}$ which contain the feature representations that contain timing ($\boldsymbol{v}_{\rho,t}$) and frequency ($\boldsymbol{h}_{\rho,t}$) structure.

must be enforced on $\mathbf{A}$. Otherwise, NMF tends to model all these parts in multiple columns of $\mathbf{B}$. Therefore, we use sparse NMF [29] where the factorization is approximated by minimizing the objective function:

$$G(\mathbf{C}||\mathbf{BA}, \lambda) = D(\mathbf{C}||\mathbf{BA}) + \lambda \sum_{i,j} \mathbf{A}_{ij} \tag{5.9}$$

The first term in (5.9) is a generalized version of the Kullback-Leibler divergence [13], defined as:

$$D(\mathbf{C}||\mathbf{BA}) = \sum_{ij} \left(\mathbf{C}_{ij} \log \frac{\mathbf{C}_{ij}}{(\mathbf{BA})_{ij}} - \mathbf{C}_{ij} + (\mathbf{BA})_{ij} \right) \tag{5.10}$$

The second term in equation (5.9) enforces sparsity on $\mathbf{A}$ by minimizing the L_1-norm of its columns. The trade off between reconstruction accuracy and sparseness is controlled by the parameter λ.

An algorithm for finding $\mathbf{B}$ and $\mathbf{A}$ given $\mathbf{C}$ based on multiplicative updates and with the additional sparseness constraint can be found in [22]. To address scaling, the constraint that each column of $\mathbf{B}$ sums to unity is imposed. Experiments have shown that with the settings used in this paper, a good choice for λ is 1000.

5.2.4 Interpreting the time-frequency patches and their activation in time

The columns of matrix $\mathbf{B}$ correspond to spectral patches which describe the recurrent time-varying spectra of speech. A selection of these patches are shown in figure 5.5 and 5.6. Just for visualization, the rows of each feature stream were extracted from $\mathbf{B}$ and were reshaped back into $N \times k$ matrices, then the positive and negative parts were recombined and the feature representations corresponding to the timing and frequency structure were plotted onto each other by means of the $\max$-operator. These figures illustrate the patches found for the static and velocity features. The parameter k of section 5.2.2 is set to 10 (figure 5.5) and 20 (figure 5.6) corresponding to a patch length of resp. 100 and 200 ms. Most patches describe formant movements over the duration of about a phone. A smaller set of time-frequency patches resemble wideband sounds and short-time energy bursts. Others are modeling the beginning or ending of phones and phone-pair transitions. Since we have discovered

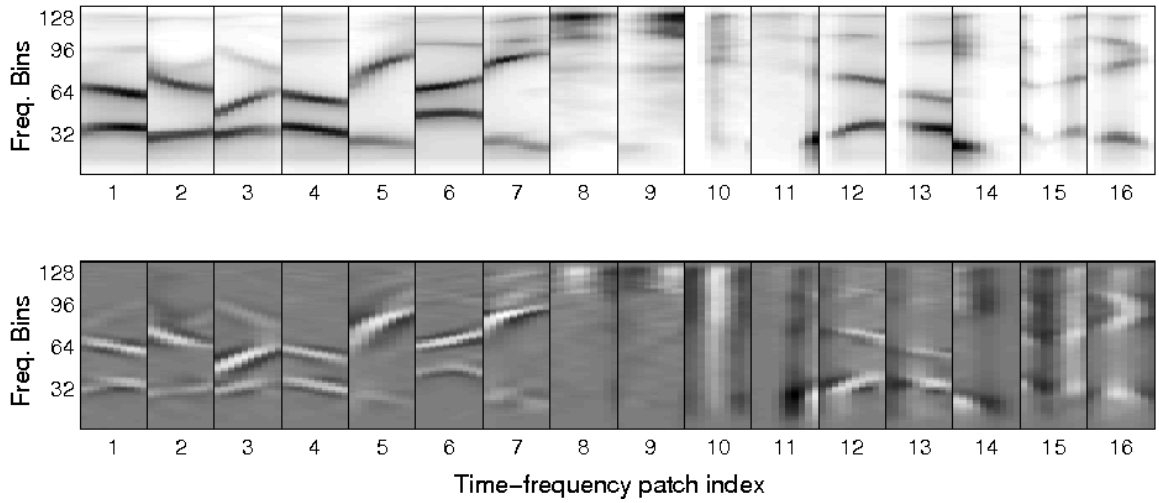

Figure 5.5: A collection of the discovered time-frequency patches for static (top row) and velocity features (bottom row) with a duration of 100 ms. Some patches show formant patterns, wideband spectra and bursts; others model inter-phone or silence-phone transitions.

the acoustic patterns from recordings composed of a sufficiently large set of different speakers, the patches are assumed to be speaker independent (will be confirmed in section 5.5).

To discover the patches that are present in test utterances of the Aurora2 database, the same procedure as in training is used except that we compute $\mathbf{A}$ in equation (5.8) by holding $\mathbf{B}$ fixed to the one obtained from training. As an example, figure 5.7 shows the time-frequency representations (a) and their corresponding patch activation matrices (b) of three examples of the word "four" each uttered by different male speakers in clean conditions (left column) and noisy conditions (right column). A set of $D = 100$ time-frequency patches were discovered from the training set as described above with a patch length of 100 ms. As can be seen from the clean speech examples in figure 5.7(b) (left pane), only few patches are highly active (black) at a certain time and sparse patterns can be discovered in the activation data. Despite variations in speaking style and speaker characteristics, the figure also suggest that each word corresponds to similar, speaker-independent activation patterns and that different words can be discriminated by comparing these patterns.

For the noisy speech examples, the babble noise type of the aurora2 database were added at different levels of signal-to-noise ratio (SNR) to the clean word utterances of figure 5.7(a), namely at 15, 10 and 5 dB SNR. In these noisy conditions, the activation patterns of the words remain similar and are hardly distorted by the different noise types. The noise robustness properties of the proposed speech model will be investigated in more detail later on (section 5.5).

Finally, figure 5.8 displays ten time-frequency patches for the word "four". The selected patches have a patch length of 100 ms and are those with the highest activation values in the utterance. The patches are ordered chronologically, e.g. patch 1 is activated from frames 31 to 33, patch 10 from frames 61 to 65.

5.2.5 Comparison with convolutive NMF

Alternatively, convolutive NMF (cNMF) could be used to obtain a parts-based representation of the data [22]. Therefore, cNMF can be applied onto the sequence of feature vectors $\boldsymbol{c}_{\rho,t}$ of equation (5.6).

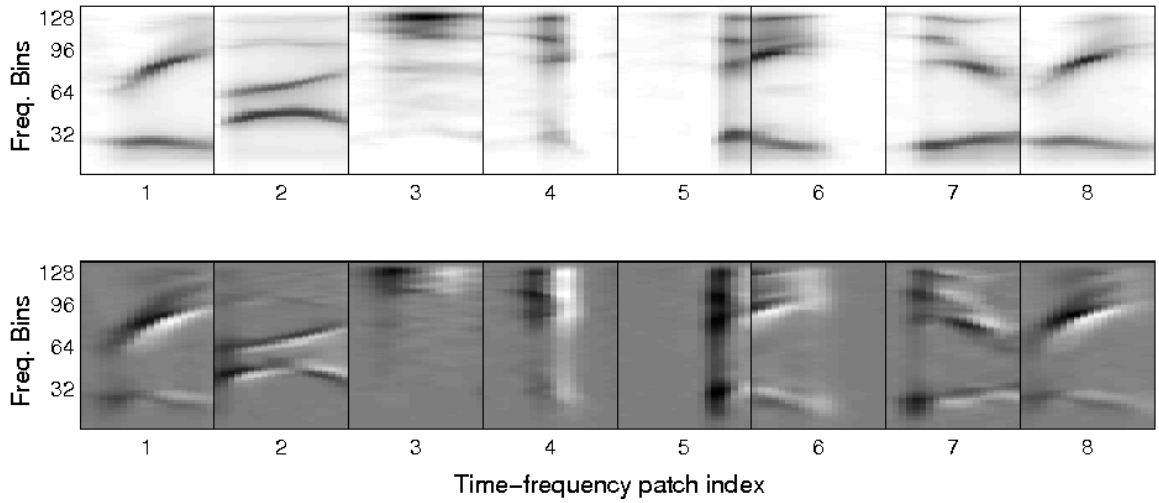

Figure 5.6: A collection of the discovered time-frequency patches with a duration of 200 ms.

However, experiments show two major drawbacks in disfavor of cNMF. Firstly, the computational requirements for cNMF are higher than those of the NMF procedure as described in section 5.2.3. The training of the time-frequency patches with cNMF involves more computational time than NMF for the same number of iteration steps. During testing, the processing time spent per iteration is similar for both factorization techniques, but cNMF requires more iterations to convergence. Secondly, although cNMF also produces speaker independent activation patterns, it turned out that these patterns are less robust to additional noise sources. From figure 5.7(c) it can be seen that the activations patterns of cNMF are more distorted in noisy speech than the patterns of NMF. These observations will be confirmed in section 5.5 by discussing and comparing the final results.

5.3 Layer 2: Acquiring activation patterns of time-frequency patches

Our bottom-up approach for language learning is driven by the assumption that a particular language is characterized by similarities in the activation patterns of time-frequency patches, as was illustrated by figure 5.7. If we assume that the patches correspond with (groups of) auditory neurons, each sensitive to a specific time-frequency pattern, then a "snapshot" of their firing rate at a certain time is represented by a column of $\mathbf{A}$. Hence, if recurring acoustic patterns in speech correspond to recurring neural firing patterns, one can hypothesize that the meaningful objects in a language (e.g. words) are characterized by similar activation patterns of time-frequency patches.

In this section, a learning algorithm is proposed that is able to acquire the objects of a language. The objects that are found are words, but could also be phone-like units in a different setting. The algorithm will discover the latent structures in patch activations by using "histograms of acoustic co-occurrence" (HAC) which are described by [30] and will be restated next. HAC-features can represent a given segment of speech in a unique high-dimensional vector without requiring a segmentation of the segment, time warping or a constraint of its duration. Moreover, each word in the utterance contributes additively to the HAC of the utterance which motivates to apply the HAC representation in association with NMF. Word identities are being provided to the algorithm to bring the discovered activation patterns in relation with the words in the vocabulary.

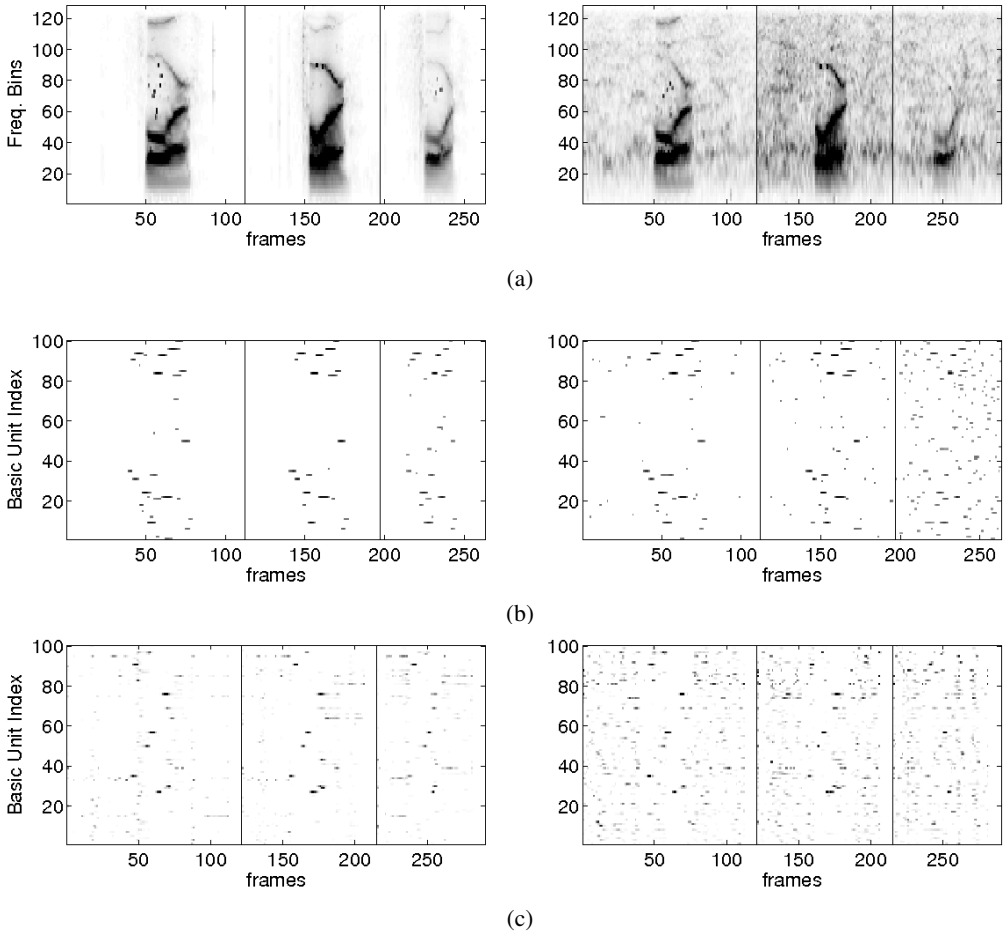

Figure 5.7: (a) Time-frequency representations of the word "four" uttered by three different speakers in clean conditions (left) and noisy conditions (right) and their corresponding patch activation matrix $\mathbf{A}$ for patches derived from the procedure with (b) NMF (section 5.2.3) and (c) cNMF (section 5.2.5).

5.3.1 Histograms of Acoustic Co-occurrences

HAC-models can describe speech by the co-occurrence statistics of acoustic events. In this work, these models are used to recover recurring patterns in patch activation events which are here the occurrence of quantized vectors in the activation matrix $\mathbf{A}$.

The activation data of the time-frequency patches will be characterized by its similarity to examples. Therefore, the columns of the activation matrix $\mathbf{A}$ are clustered into Σ centroids using the K-means algorithm. Given the Euclidean distance metric used in clustering, each centroid can be represented by a Gaussian with spherical covariance. As a consequence, the posterior probabilities $P_{i,n}$ of all centroids n characterize any frame i of $\mathbf{A}$ in terms of its similarity to each of the centroids. For each frame i, the posterior probabilities satisfy:

$$\sum_{n=1}^{\Sigma} P_{i,n} = 1 \tag{5.11}$$

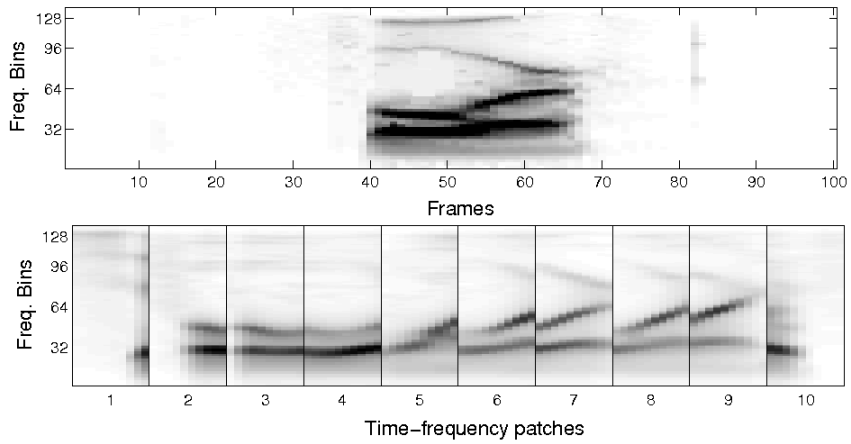

Figure 5.8: Time-frequency representation of the word "four" (top) and the time-frequency patches (bottom) with highest activation values in the utterance.

A special case is obtained in a "winner takes all" setting, where all posteriors are zero except for the centroid closest to the observation, which is assigned the value 1. This setting is related to a vector quantization (VQ) approach where the centroids are the codebook entries labeled from 1 to Σ. After decoding, each frame of the activation matrix $\mathbf{A}$ is then replaced by the best matching centroid of the codebook, which allows to reduce the activation matrix to a single row vector of VQ-labels.

The HAC-representation is then the number of times all VQ-label pairs $(m, n) \in \Sigma \times \Sigma$ are observed τ frames apart. In other words, a histogram of lag-τ co-occurrences is constructed where each co-occurrence signifies that the input of activation frames is encoded into a VQ-label m at time i, while encoded into VQ-label n at time $i + \tau$. For a given utterance u, the lag-τ co-occurrence is weighted with the (approximated) probability of the event

$$[\mathbf{V}_u^\tau]_{mn} = \sum_{i=1}^{I_u-\tau} P_{i,m} P_{i+\tau,n} \quad \text{with } m, n = 1 \ldots \Sigma \tag{5.12}$$

where I_u is the number of frames in the utterance. Also note that $[\mathbf{V}_u^\tau]_{mn} \neq [\mathbf{V}_u^\tau]_{nm}$, such that these co-occurrences are directed.

By stacking all (m, n)-combinations, each utterance can be represented by a single column vector $\mathbf{V}_u^\tau$ where the elements express the sum of all Σ^2 possible lag-τ co-occurrences. We will refer to this vector as a *HAC* (histogram of acoustic co-occurrence).

This procedure can be performed for different τ-values and for a given set of time-frequency patches with a patch length of k frames. For a set of U utterances, the data matrix for a choice of k and τ is then formed by

$$\mathbf{V}^{k,\tau} = \left[\begin{array}{cccccc} \mathbf{V}_1^{k,\tau} & \ldots & \mathbf{V}_u^{k,\tau} & \ldots & \mathbf{V}_U^{k,\tau}. \end{array} \right] \tag{5.13}$$

Note that thanks to the vector quantization approach, matrix $\mathbf{V}^{k,\tau}$ has a high sparsity. Furthermore, all its entries are non-negative such that NMF-methods can be applied.

5.3.2 Semi-supervised learning with NMF

Suppose that the utterances are composed of R recurring acoustic events such as words, each constructed from the set of time-frequency patches. Since (5.13) is a sum over time of activations, the words will contribute additively to the corresponding column of $\mathbf{V}^{k,\tau}$. As each word is characterized by a HAC, the HAC of each utterance will be a (integer) linear combination of these histograms.

If the HAC of the words are placed in separate columns of a matrix $\mathbf{W}$, and if the corresponding rows of $\mathbf{H}$ would contain the presence of each word in each utterance, one would have (leaving out indices k and τ):

$$\mathbf{V} \approx \mathbf{W}\mathbf{H} \tag{5.14}$$

Given their interpretation, all entries of $\mathbf{W}$ and $\mathbf{H}$ are constrained to be positive or zero. Because of these constraints and given the fact that equation (5.14) will not hold exactly since the observed symbols are subject to variability and uncertainty, $\mathbf{W}$ and $\mathbf{H}$ are estimated by NMF. Factorization of $\mathbf{V}$ is performed using the approach of section 5.2.3 without enforcing sparsity constraints, e.g. we set λ to 0.

Once $\mathbf{W}$ is estimated on a training set, new utterances can be analyzed with factorization (5.14) by estimating $\mathbf{H}$. The degree to which each discovered activation pattern is present in each new utterance is then found by examining the columns of $\mathbf{H}$.

In this work, the words are unknown and NMF is used to separate them out from the utterances. However, since utterances can be seen as a sequence of words, but also as, for instance, a sequence of phones, constraints have to be imposed on (5.14) by exploiting grounding information. If it is known which words occur in each utterance, this information can be exploited to associate a word identity to each column of $\mathbf{W}$. Therefore, the $L \times U$ grounding matrix $\mathbf{G}$ is formed, which holds in its l-th row and u-th column the number of times the l-th word occurs in the u-th utterance. Here, L is the number of word identities and U is the number of training utterances. Subsequently, we compute:

$$\begin{bmatrix} \mathbf{G} \\ \mathbf{V} \end{bmatrix} = \begin{bmatrix} \mathbf{W}_g \\ \mathbf{W}_v \end{bmatrix} \mathbf{H} \tag{5.15}$$

which expresses that word identities need to be explained jointly with the acoustic data by common model activations $\mathbf{H}$. Given the properties of multiplicative updates [13], grounding forces the NMF decomposition to associate word models in $\mathbf{W}_v$ also to the utterances containing those words. Without augmenting $\mathbf{V}$ with the grounding matrix, NMF tends to spend columns of $\mathbf{W}_v$ preferably on the more frequent acoustic patterns since this has the most impact on minimizing the modeling error. Experiments have shown that the common dimension R is better overestimated, hence $R \geq L$. This allows to model acoustic events that have no relevance to grounding such as silence or filler words.

5.3.3 Improving learning by modeling multiple streams

As explained in section 5.3.1, the data matrix $\mathbf{V}^{k,\tau}$ contains lag-τ co-occurrences in the activation data of time-frequency patches with a duration of k frames. For each individual configuration (k, τ), patterns can be learned using the approach of section 5.3.2 by connecting acoustic information and by assigning a meaning to these patterns. We may assume that the performance of the learning algorithm will increase if multiple configurations are combined in the input matrix. This idea has already been exploited by jointly capturing static, velocity and acceleration feature information in each time-frequency patch. On the activation level, the data matrix of (5.14) can be further extended by

incorporating different sets of patches and including co-occurrence data at different time lags. By allowing the use of patches with different durations, we could also compensate for the time differences of phones. For instance, one can expect that plosives cause more neural activation at neurons modeling time-frequency patches with a duration around 50 ms, while neurons corresponding to patches of 100 ms better represent diphthongs and vowels. Units with even longer duration (e.g. 200 ms) can be used to model intra- and inter-phoneme transitions. Therefore P sets of time-frequency patches are included in the model, each exploiting Q values of τ, augmented with grounding information that relates to the spoken words:

$$\begin{bmatrix} \mathbf{G} \\ \mathbf{V}^{k_1,\theta} \\ \vdots \\ \mathbf{V}^{k_P,\theta} \end{bmatrix} = \begin{bmatrix} \mathbf{W}_g \\ \mathbf{W}_v \end{bmatrix} \mathbf{H} \quad \text{with} \quad \mathbf{V}^{k_i,\theta} = \begin{bmatrix} \mathbf{V}^{k_i,\tau_1} \\ \vdots \\ \mathbf{V}^{k_i,\tau_Q} \end{bmatrix}. \tag{5.16}$$

For these joint streams, the generative parts-based model still holds: the joint stream co-occurrences of utterances can be written as an additive combination of parts. As we will show in the experiments of section 5.5, it is indeed advantageous to exploit multiple combination of k and τ in the HAC-model.

5.4 Layer 3: Detecting words in activation patterns

After the semi-supervised training procedure, $\mathbf{W}_g$ and $\mathbf{W}_v$ are known. Recognition on unseen test utterances (from which grounding information $\mathbf{G}$ is unknown), is achieved by first computing the histograms of co-occurrence $\mathbf{V}$ and then estimating the matrix factor $\hat{\mathbf{H}}$ in $\mathbf{V} \approx \mathbf{W}_v\hat{\mathbf{H}}$ by holding $\mathbf{W}_v$ fixed. This matrix $\hat{\mathbf{H}}$ reveals to which extent the internal representations of the trained words are present in the new test utterance. By estimating the grounding information as:

$$\hat{\mathbf{G}} = \mathbf{W}_g\hat{\mathbf{H}} \tag{5.17}$$

we obtain estimates for the presence of the words in the test utterances. Hence, for a word that is present, the corresponding element of $\hat{\mathbf{G}}$ tends to 1 and to 0 if it is absent. This way, a word detection system can be build from the content of matrix $\hat{\mathbf{G}}$ by comparing $\hat{\mathbf{G}}$ with a threshold ξ. Two types of errors are involved in the system: missed detections ($\hat{\mathbf{G}}_{ij} < \xi$ while utterance j contains word i) and false alarms ($\hat{\mathbf{G}}_{ij} \geq \xi$ while utterance j does not contain word i). The trade-off of both error types can be visualized by means of a Detection Error Trade-off (DET) curve [31]. In figure 5.9, the DET-curve is shown for the word detection task where the model is trained on lag-10 co-occurrence counts (τ=10) computed on a set of time-frequency patches with a length of 100 ms (k=10). The performance of the system in clean speech conditions is compared with a noisy test case where the speech is distorted by babble noise at 10 dB SNR. The estimated grounding matrix $\hat{\mathbf{G}}$ of five different utterances for both test cases are shown in figure 5.10 where high values (black) indicate that the corresponding words have a high probability to be present in the utterance.

In the experimental evaluation of section 5.5, we will not apply this "per word detection" paradigm, which is relevant for tasks such as keyword spotting. Instead we will measure correct word recognition per utterance. Assuming that the number D_u of different digits occurring in the u-th test utterance is given, the D_u candidates with highest activation according to equation (5.17) are selected. Notice that the recognition result is unordered, a problem that is addressed in [30] by a sliding window decoder that estimates at which time each word occurs in the utterance. Word error rate is thus defined

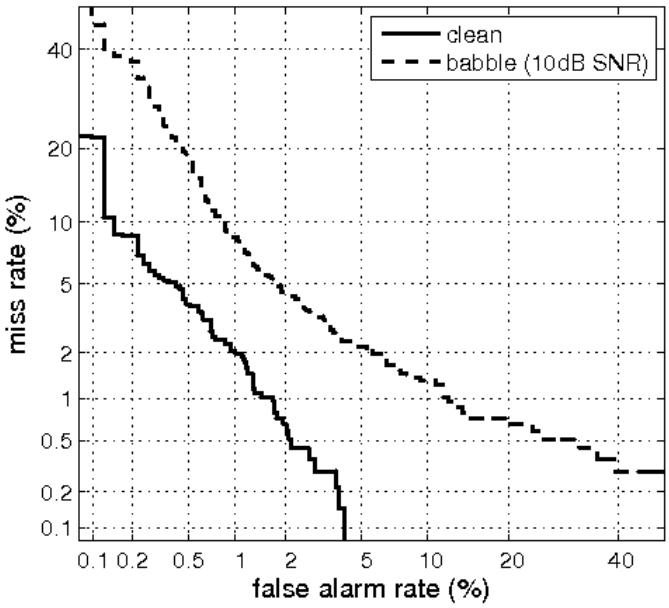

Figure 5.9: DET-curves of the word detection system for clean and noisy speech (babble noise at 10 db SNR).

as the sum of the number of incorrect digits that end up in the top D_u, divided by the sum of D_u over the complete test set.

5.5 Experiments

The speech data are taken from Aurora2, a small vocabulary, speaker independent database for connected digit recognition as defined by [32]. All utterances are derived from the TI-Digits database that contains recordings of male and female US-American adults, downsampled to 8 kHz sampling frequency. The database contains isolated digits and sequences of up to seven digits. The Aurora2 clean-condition training set consists of 8440 utterances that are used for the discovering the time-frequency patches and their activation patterns from which the HAC-models are derived. The test experiments are conducted on test set A consisting of 4004 utterances from the TI-Digits test data, split into 4 subsets of 1001 utterances each and to which 4 different noises are added at different SNR-levels: noise recorded in a subway (N1), babble noise (N2), car noise (N3) and noise recorded in an exhibition hall (N4).

5.5.1 Reference experiment

Baseline recognition results are produced by a conventional HMM-based recognition system using the complex back-end configuration as defined by the ETSI Aurora group [33]. Whole word digit models were trained on the clean speech training database using the HTK software package version 2.2 from Entropic [34]. The digit models have 16 emitting states with 20 Gaussians per state. The optional inter-word silence is modeled by 1 or 3 states with 36 Gaussians per state, while leading and

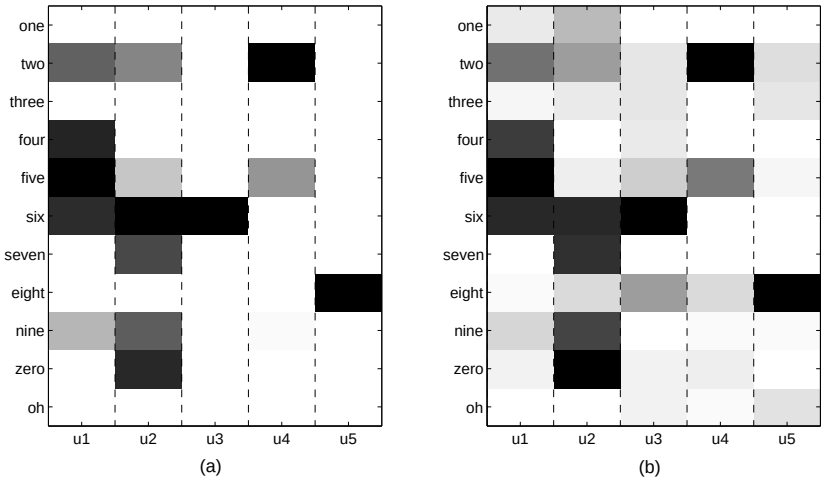

Figure 5.10: Estimated grounding matrix $\hat{\mathbf{G}}$ for (a) clean and (b) noisy speech (babble noise at 10 db SNR) of the utterances "four six two five" (u1), "zero seven nine six two" (u2), "six" (u3), "two five two" (u4) and "eight" (u5).

trailing silence have 3 states. The total number of Gaussians is 3628. Feature were extracted by the Aurora WI007 front-end [35], a cepstral analysis scheme where 12 Mel-scaled cepstral coefficients and c_0 (no log-energy) are determined for a speech frame of 25 ms length using a frame shift of 10 ms. These features are combined with their dynamic coefficients to yield in 39-dimensional feature vectors for recognition, as explained in [36].

Here, we assume that for each test utterance u the number N_u of digit sequences is known. This information is then used in the language model by forcing the decoder to recognize exactly N_u digits. From the recognition result, only the different digits are retained to obtain an unordered string result of at most N_u digits. Similarly as in section 5.4, a detection error is accounted for each digit from the set of D_u different and correct digits in the utterance that is not present in the unordered recognition result. The error rate of this HMM-based word detection system is shown in table 5.1. Results were averaged over the four noise types of Aurora2.

clean	15dB	10dB	5dB
0.16	2.98	11.92	34.88

Table 5.1: Unordered word error rate of an HMM-based word detection system on the Aurora2 database averaged over the four noise types.

5.5.2 Training procedure

In the experiments, multiple sets of time-frequency patches, modeling acoustic patterns of different durations, were trained on the clean training set of Aurora2. The patches are learned from a data matrix constructed from static, velocity and acceleration features as explained in section 5.2, while using the following values for k: 5, 10, 15, 20. For each set, the number of patches P to be discovered

k				τ							
5	10	15	20	5	10	15	20	clean	15dB	10dB	5dB
	×			×				2.10	4.29	7.07	12.54
	×				×			1.98	3.74	6.37	11.65
	×					×		2.22	4.02	6.78	11.58
	×						×	2.67	4.90	7.33	12.64
	×				×		×	2.17	3.80	5.79	10.34
	×			×	×	×		1.89	3.58	5.86	10.83
	×				×	×	×	1.96	3.74	5.82	10.86
	×			×	×	×	×	1.93	3.56	5.64	10.67
	×		×	×	×	×	×	1.94	3.41	5.39	9.35
×	×	×		×	×	×	×	**1.83**	3.26	5.24	9.35
	×	×	×	×	×	×	×	2.01	3.36	5.12	8.93
×	×	×	×	×	×	×	×	1.87	**3.08**	**4.94**	**8.67**

Table 5.2: Unordered word error rate results on the Aurora2 database averaged over the four noise types for the proposed recognition system using a combination of different sets of time-frequency patches and different lag-τ co-occurrence counts. The time-frequency patches are derived from the enhanced and smoothed RTFR features using the NMF procedure described in section 5.2.3. The $\times$-symbol indicates which configurations (k, τ) are integrated in the input matrix of equation (5.16).

is 100. Experiments, not reported in this paper, have shown that this number suffices to model the different spectral patterns of the small vocabulary task of Aurora2. The obtained patches are modelled by the columns of the four matrices $\mathbf{B}^k$ which are stored for the recognition task on test data.

For each set of time-frequency patches, the patch activation vectors in $\mathbf{A}^k$ are quantized using a codebook of 250 entries, resulting in 4 sequences of VQ-labels. Subsequently, the VQ-label co-occurrence histograms are computed for all utterances using different lag-τ values: 5, 10, 15, 20. The VQ histogram counts are divided by a fixed constant (100) such that the acoustic and grounding information have roughly the same weight in the data matrix $\mathbf{V}^{k,\tau}$. Experiments have shown that the value of this constant is not critical: it can be changed over several orders of magnitude without significant impact. To acquire all eleven words of Aurora2, namely the digits "one" to "nine", "zero" and "oh", the training procedure as described in section 5.3 was performed with R = 12 using the utterances of the clean training set. After factorization (5.15), $\mathbf{W}_g$ and $\mathbf{W}_v$ are stored for recognition.

5.5.3 Evaluating the results

To discover the digits that are present in the test utterances, the same procedure as in training is used except that we compute the patch activation matrix $\mathbf{A}$ in (5.8) by holding $\mathbf{B}$ fixed to the one obtained from training. Similarly, the word activation matrix $\mathbf{H}$ is found by holding $\mathbf{W}$ fixed in (5.14) to the one estimated from the training set.

Table 5.2 shows the unordered word error rate on the Aurora2 test set averaged over the four noise types, using different stream configurations (k, τ). For clean speech, the self-learning algorithm performs worse than the HMM-based system that makes use of expert speech knowledge that arises from audiology and linguistics. However, our system performs comparably to the HMM-based rec-

k				τ							
5	10	15	20	5	10	15	20	clean	15dB	10dB	5dB
	×		×	×	×	×	×	2.06	4.27	6.48	11.62
×	×	×		×	×	×	×	**1.81**	4.23	6.78	12.08
	×	×	×	×	×	×	×	2.06	4.20	**6.40**	**11.06**
×	×	×	×	×	×	×	×	1.91	**4.12**	6.43	11.56

Table 5.3: Unordered word error rate results for Aurora2 averaged over the four noise types using time-frequency patches derived from STFT features by the NMF procedure.

k				τ							
5	10	15	20	5	10	15	20	clean	15dB	10dB	5dB
	×		×	×	×	×	×	3.45	6.57	10.23	18.16
×	×	×		×	×	×	×	**2.76**	5.83	9.48	16.36
	×	×	×	×	×	×	×	3.18	5.90	9.44	16.49
×	×	×	×	×	×	×	×	2.85	**5.50**	**9.08**	**15.78**

Table 5.4: Unordered word error rate results for Aurora2 averaged over the four noise types using time-frequency patches derived from the enhanced and smoothed RTFR features by the cNMF procedure discussed in section 5.2.5.

ognizer at 15 dB SNR and has a remarkably higher accuracy for noisy speech at lower SNRs without using any noise compensation techniques. From the table 5.2, we can also observe that the robustness can be increased by exploiting more knowledge sources. The reason for this noise robustness is three-fold: (i) thanks to the parts-based representation of speech, the system easily detects, even in noisy conditions, which time-frequency patches are active; (ii) these time-frequency patches provide static and dynamic spectral information over large time windows; (iii) multi-window time-frequency representations can be exploited by the joint modeling of different streams.

For comparison, we added the results of the STFT features with the same time windows parameters for the timing and frequency structure (see table 5.3) and those where cNMF are performed onto the feature vectors of equation (5.6) (see table 5.4). As can be seen from both tables, the word error rates are worse than those shown in table 5.2. This indicates respectively that the enhanced and smoothed RTFR features are more robust than STFT features and that the activation patterns acquired by the proposed approach using NMF to discover time-frequency patches deliver a more robust input stream to the word detection system than those obtained by the cNMF approach.

5.6 Conclusions

In this paper, we proposed a bottom-up approach for learning words of a language. An unsupervised technique was presented to discover a set of spectral patches that can describe speech. We exploited the use of two complementary feature representations derived from a reassigned time-frequency spectrogram to obtain a representation that can cope with acoustic events with short and long durations. The non-negative matrix factorization (NMF) algorithm using sparsity constraints was applied to discover latent recurring patterns in static and dynamic features. The obtained basis vectors correspond

to phone-sized spectral patterns which we referred to as time-frequency patches. Experiments on the Aurora2 database revealed that these patches are activated in speaker-independent patterns which are related to the words of a language.

Next, a learning algorithm was built that automatically discovers and acquires the recurring patterns in the activation data by applying NMF on the co-occurrence counts of activation events. The obtained patterns were associated with the words of a language and finally the system was able to detect the words in unseen (noisy) speech data.

Experimental evidence was given for the noise robustness of the proposed word detection system, based on the Aurora2 digit recognition task. Although a conventional HMM-based approach using cepstral features obtained better results on clean speech data, the proposed learning algorithm showed a superior performance for speech that is distorted by the noise down to 5 dB SNR. The NMF learning algorithm was shown to be sufficiently versatile to apply it at both levels of speech representations for discovering structure in the data. NMF has less parameters to be tuned in comparison to HMM-based systems. The most important parameters are the number of time-frequency patches P, the sparsity parameter λ in the NMF of the first layer and the number of VQ-labels Σ in the second layer. Moreover, experiments not reported here have shown that for the small vocabulary task as was considered in this paper, the performance of the system is not very sensitive to these parameters.

Inspired by research on the auditory cortex of mammals, researchers have suggested that ASR systems should trigger on the presence of spectro-temporal patches. Such biologically inspired systems might exhibit properties of human audition such as robustness to noise. In this paper, we have shown that such an auditory representation with good robustness can be obtained through unsupervised learning (the first layer). We have also shown how the activation patterns can be exploited to build a speech recognizer. Further work involves extending the word detection system to a speech recognition system that also provides information related to the order in which the words occur in the test utterances. Therefore, the HAC-models in the second layer can be extended by moving a sliding window over the utterance to detect the time of occurrence of the different words in the utterance and hence the word order. In our current implementation, the noise has been left uncompensated and we would like to investigate to which extent the performance of the recognition system can be further improved by exploiting noise reduction techniques. Though, the layered architecture offers scalability towards vocabularies in the sense that the patch set is reusable across words, more research is required to reveal how well the proposed system is suited for large vocabulary continuous speech recognition. At his point, the presented three-layered architecture is not capable to deal with these vocabularies because of the large data requirements per word. However, the scalability of the system can be increased by adding more layers in cascade to model the words as a combination of subword patches instead of learning all the words from scratch.

To conclude, we believe that the presented system is an ideal platform for future research as in its baseline implementation it already yields competitive results and could open new avenues of research on automatic speech recognition.

5.7 Acknowledgement

This research was funded by the Institute for the Promotion of Innovation through Science and Technology in Flanders, Belgium (I.W.T.-Vlaanderen) and by the European Commission under contract FP6-034362 (ACORNS).

Bibliography

[1] N. Chomsky, *New Horizons in the Study of Language and Mind*. Cambridge, UK: Cambridge University Press, 2000.

[2] J. M. Baker, L. Deng, S. Khudanpur, C.-H. Lee, J. Glass, and N. Morgan, "Minds historical development and future directions in speech recognition and understanding," Report of the Speech Understanding Working Group, Tech. Rep., 2006–2007, http://www.itl.nist.gov/iaui/894.02/MINDS/ FINAL/speech.web.pdf.

[3] O. Scharenborg, D. Norris, L. ten Bosch, and M. J.M., "How should a speech recognizer work?" *Cognitive Science*, vol. 29, no. 6, pp. 867–918, 2005.

[4] O. Scharenborg, M. Ernestus, and V. Wan, "Segmentation of speech: Child's play?" in *Proc. International Conference on Spoken Language Processing*, Antwerp, Belgium, Aug. 2007, pp. 1953–1956.

[5] Y. Qiao, Shimomura, and N. N. Minematsu, "Unsupervised phoneme segmentation using transformed cepstrum features," pp. 287–290, 2008, 1-11-20.

[6] F. Brugnara, D. Falavigna, and M. Omologo, "Automatic segmentation and labeling of speech based on hidden markov models," *Speech Communication*, vol. 12, no. 4, pp. 357–370, 1993.

[7] G. Aversano, Esposito, and M. Marinaro, "A new text-independent method for phoneme segmentation," *Proc. IEEE Int. Workshop Circuits and Systems*, vol. 2, pp. 516–519, 2001.

[8] V. Siivola, T. Hirsimäki, M. Creutz, and M. Kurimo, "Unlimited vocabulary speech recognition based on morphs discovered in an unsupervised manner," in *Proc. Eurospeech*, 2003, pp. 2293–2296.

[9] A. Park and J. Glass, "Towards unsupervised pattern discovery in speech," in *Proc. ASRU*, San Juan, Puerto Rico, Dec. 2005, pp. 53–58.

[10] V. Stouten, K. Demuynck, and H. Van hamme, "Discovering phone patterns in spoken utterances by nonnegative matrix factorisation," *IEEE Signal Processing Letters*, vol. 15, pp. 131–134, 2008.

[11] P. Smaragdis, "Convolutive speech bases and their application to speech separation," *IEEE Transactions of Speech and Audio Processing*, vol. 15, no. 1, pp. 1–12, Jan. 2007.

[12] B. T. Meyer and B. Kollmeier, "Optimization and evaluation of gabor feature sets for ASR," in *Proc. International Conference on Spoken Language Processing*, Brisbane, Australia, Sep. 2008, pp. 906–909.

[13] D. Lee and H. Seung, "Algorithms for non-negative matrix factorization," *Advances in Neural Information Processing Systems*, vol. 13, pp. 556–562, 2001.

[14] H. Hermansky and S. Sharma, "TRAPS - classifiers of temporal patterns," in *Proc. International Conference on Spoken Language Processing*, Sydney, Australia, Nov. 1998, pp. 1003–1006.

[15] ——, "Temporal patterns (TRAPs) in ASR of noisy speech," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, vol. 1, Phoenix, Arizona, Mar. 1997, pp. 289–292.

[16] H. Hermansky and P. Fousek, "Multi-resolution RASTA filtering for TANDEM-based ASR," in *Proc. International Conference on Spoken Language Processing*, Lisbon, Portugal, Oct. 2005, pp. 361–364.

[17] B. Kingsbury, N. Morgan, and S. Greenberg, "Robust speech recognition using the modulation spectrogram," *Speech Communication*, vol. 25, pp. 117–132, Aug. 1998.

[18] V. Tyagi, I. McCowan, H. Misra, and H. Bourlard, "Mel-cepstrum modulation spectrum (MCMS) features for robust ASR," in *Proc. ASRU 2003 Workshop*, St. Thomas, Virgin Islands, Dec. 2003, pp. 399–404.

[19] M. Kleinschmidt, "Localized spectro-temporal features for automatic speech recognition," in *Proc. Eurospeech*, Geneva, Switzerland, Sep. 2003, pp. 2573–2576.

[20] T. Ezzat, Bouvrie, J., and T. Poggio, "Spectro-temporal analysis of speech using 2-D gabor filters," in *Proc. International Conference on Spoken Language Processing*, Antwerp, Belgium, Aug. 2007, pp. 506–509.

[21] T. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," *IEEE Transactions of Audio, Speech and Language Processing*, vol. 15, no. 3, pp. 1066–1074, March 2007.

[22] P. D. O'Grady and B. A. Pearlmutter, "Discovering speech phones using convolutive non-negative matrix factorisation with a sparseness constraint," *Neurocomputing*, vol. 72, pp. 88–101, Dec. 2008.

[23] H. Van hamme, "Integration of asynchronous knowledge sources in a novel speech recognition framework," in *ISCA ITRW workshop on Speech Analysis and Processing for Knowledge Discovery*, Aalborg, Danmark, June 2008.

[24] F. Auger and P. Flandrin, "Improving the readability of time-frequency and time-scale representations by the reassignment method," *IEEE Transactions on Signal Processing*, vol. 43, no. 5, pp. 1068–1089, 1995.

[25] F. Plante, G. Meyer, and W. Ainsworth, "Improvement of speech spectrogram accuracy by the method of reassignment," *IEEE Transactions on Speech and Audio Processing*, vol. 6, no. 3, pp. 282–286, 1998.

[26] S. Hainsworth and M. Macleod, "Time-frequency reassignment: a review and analysis," Cambridge University Engineering Department, Tech. Rep. CUED/FINFENG/TR.459, 2003.

[27] K. Kodera, R. Gendrin, and C. Villedary, "Analysis of time-varying signals with small bt values," *IEEE Transactions of Audio, Speech and Language Processing*, vol. 26, no. 1, pp. 64–76, feb 1978.

[28] C. Ding, T. Li, and M. Jordan, "Convex and semi-nonnegative matrix factorizations for clustering and low-dimension representation," Lawrence Berkeley National Laboratory, US, Tech. Rep. LBNL-60428, 2006.

[29] P. Hoyer, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research*, vol. 5, pp. 1457–1469, 2004.

[30] H. Van hamme, "HAC-models: a novel approach to continuous speech recognition," in *Proc. International Conference on Spoken Language Processing*, Brisbane, Australia, 2008, pp. 2554–2557.

[31] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The DET curve in assessment of detection task performance," in *Proc. Eurospeech*, Rhodes, Greece, Sep. 1997, pp. 1895–1898.

[32] H. G. Hirsch and D. Pearce, "The AURORA experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *Proc. ISCA ITRW ASR2000 Workshop*, Paris, France, Sep. 2000, pp. 18–20.

[33] ETSI standard document, *Speech processing, transmission and quality aspects (STQ); distributed speech recognition; advanced front–end feature extraction algorithm; compression algorithm*, Apr. 2000, eTSI ES 202 050 v1.1.1 (2002-10).

[34] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland, *The HTK book – version2.2*. Entropic, 1999.

[35] ETSI standard document, *Distributed Speech Recognition; Front end Feature Extraction Algorithm; Compression Algorithm*, Apr. 2000, eTSI ES 201 108 v1.1.2.

[36] D. Macho, L. Mauuary, B. Noé, Y. Cheng, D. Ealey, D. Jouvet, H. Kelleher, D. Pearce, and F. Saadoun, "Evaluation of a noise-robust DSR front-end on Aurora databases," in *Proc. International Conference on Spoken Language Processing*, Denver, Colorado, U.S.A., Sep. 2002, pp. 17–20.