ON GENERALIZATION OF SUPERVISED SPEECH SEPARATION

DISSERTATION

Presented in Partial Fulfillment of the Requirements for
the Degree Doctor of Philosophy in the
Graduate School of The Ohio State University

By

Jitong Chen, M.S.

Graduate Program in Computer Science and Engineering

The Ohio State University

2017

Dissertation Committee:

Professor DeLiang Wang, Advisor

Professor Eric Fosler-Lussier

Professor Eric W. Healy

ABSTRACT

Speech is essential for human communication as it not only delivers messages but also expresses emotions. In reality, speech is often corrupted by background noise and room reverberation. Perceiving speech in low signal-to-noise ratio (SNR) conditions is challenging, especially for hearing-impaired listeners. Therefore, we are motivated to develop speech separation algorithms to improve intelligibility of noisy speech. Given its many applications, such as hearing aids and robust automatic speech recognition (ASR), speech separation has been an important problem in speech processing for decades.

Speech separation can be achieved by estimating the ideal binary mask (IBM) or ideal ratio mask (IRM). In a time-frequency (T-F) representation of noisy speech, the IBM preserves speech-dominant T-F units and discards noise-dominant ones. Similarly, the IRM adjusts the gain of each T-F unit to suppress noise. As such, speech separation can be treated as a supervised learning problem where one estimates the ideal mask from noisy speech. Three key components of supervised speech separation are learning machines, acoustic features and training targets. This supervised framework has enabled the treatment of speech separation with powerful learning machines such as deep neural networks (DNNs). For any supervised learning problem, generalization to unseen conditions is critical. This dissertation addresses generalization of supervised speech separation.

We first explore acoustic features for supervised speech separation in low SNR conditions. An extensive list of acoustic features is evaluated for IBM estimation. The list includes ASR features, speaker recognition features and speech separation features. In addition, we propose the Multi-Resolution Cochleagram (MRCG) feature to incorporate both local information and broader spectrotemporal contexts. We find that gammatone-domain features, especially the proposed MRCG features, perform well for supervised speech separation at low SNRs.

Noise segment generalization is desired for noise-dependent speech separation. When tested on the same noise type, a learning machine needs to generalize to unseen noise segments. For nonstationary noises, there exists a considerable mismatch between training and testing segments, which leads to poor performance during testing. We explore noise perturbation techniques to expand training noise for better generalization. Experiments show that frequency perturbation effectively reduces false-alarm errors in mask estimation and leads to improved objective metrics of speech intelligibility.

Speech separation in unseen environments requires generalization to unseen noise types, not just noise segments. By exploring large-scale training, we find that a DNN based IRM estimator trained on a large variety of noises generalizes well to unseen noises. Even for highly nonstationary noises, the noise-independent model achieves similar performance as noise-dependent models in terms of objective speech intelligibility measures. Further experiments with human subjects lead to the first demonstration that supervised speech separation improves speech intelligibility for hearing-impaired listeners in novel noises.

Besides noise generalization, speaker generalization is critical for many applications where target speech may be produced by an unseen speaker. We observe that training a DNN with many speakers leads to poor speaker generalization. The performance on seen speakers degrades as additional speakers are added for training. Such a DNN suffers from the confusion of target speech and interfering speech fragments embedded in noise. We propose a model based on recurrent neural network (RNN) with long short-term memory (LSTM) to incorporate the temporal dynamics of speech. We find that the trained LSTM keeps track of a target speaker and substantially improves speaker generalization over DNN. Experiments show that the proposed model generalizes to unseen noises, unseen SNRs and unseen speakers.

This work is dedicated to my family.

ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere gratitude to my advisor Prof. DeLiang Wang. This dissertation would not be possible without his guidance and support. During my graduate study, Prof. Wang has taught me the key qualities of a strong researcher: creativity, rigor and persistence. I am grateful for his encouragement and insights, which accelerate my progress in cracking challenging research problems. It is my pleasure to work with Prof. Wang on this dissertation topic, which has exciting applications in the real world.

I would like to thank Prof. Eric Fosler-Lussier and Prof. Eric Healy for serving on my dissertation committee. I learn the fundamentals of ASR from Prof. Fosler-Lussier. His classes provide me with key ideas in acoustic modeling, which are inspirational for my research. I am fortunate to have worked with Prof. Healy and his team on evaluating proposed algorithms with rigorous subject tests, which provides insights into the potential of our algorithms for real-world applications.

I am fortunate to have two internships in industry. I thank Dr. Richard Socher for giving me the opportunity to work on natural language processing in his startup company MetaMind in 2015. I also thank Dr. Shankar Kumar for hosting me in Google's speech and language algorithms team in 2016. I benefited a lot from his expertise in ASR and I enjoyed our discussions on research ideas. The two internships have broadened my view of machine learning based applications.

I would like to thank my labmates. I have received a lot of guidance from senior labmates. Dr. Yuxuan Wang helped me start research in the first few years. Our later collaboration on large-scale training has led to significant research progress. Dr. Arun Narayanan patiently answered my questions on ASR and acoustic features. I learned a great deal about experimental design from Dr. Kun Han. I also benefited a lot from Dr. Xiaojia Zhao's expertise on robust speaker recognition, and Dr. Donald Williamson's experience on improving speech quality. Aside from research, I enjoyed going to lunch with my labmates and chatting about graduate life.

Finally, I would like to extend my heartfelt gratitude to my parents, my father Yuekao Chen and my mother Lufen Zheng. Without their love and encouragement, I would have never made it this far in my graduate study.

VITA

October 4, 1989	Born in Taizhou, Zhejiang, China
2011	B.E. in Information Security, Northeastern University, Shenyang, China
2015	M.S. in Computer Science and Engineering, The Ohio State University

PUBLICATIONS

J. Chen, Y. Wang, and D. L. Wang, "A Feature Study for Classification-Based Speech Separation at Very Low Signal-to-Noise Ratio," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 7039–7043, 2014.

J. Chen, Y. Wang, and D. L. Wang, "A Feature Study for Classification-Based Speech Separation at Low Signal-to-Noise Ratios," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 22, pp. 1993–2002, 2014.

J. Chen, Y. Wang, and D. L. Wang, "Noise Perturbation Improves Supervised Speech Separation," in *Proc. International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA)*, pp. 83-90, 2015.

E. Healy, S. Yoho, J. Chen, Y. Wang, and D. L. Wang, "An Algorithm to Increase Speech Intelligibility for Hearing-Impaired Listeners in Novel Segments of the Same Noise Type," *Journal of the Acoustical Society of America*, vol. 138, pp. 1660–1669, 2015.

Y. Wang, J. Chen, and D. L. Wang, "Deep Neural Network Based Supervised Speech Segregation Generalizes to Novel Noises Through Large-Scale Training", *Technical Report OSU-CISRC-3/15-TR02*, 2015.

J. Chen, Y. Wang, and D. L. Wang, "Noise Perturbation for Supervised Speech Separation", *Speech Communication*, vol. 78, pp. 1–10, 2016.

J. Chen, Y. Wang, S. Yoho, D. L. Wang, and E. Healy, "Large-Scale Training to Increase Speech Intelligibility for Hearing-Impaired Listeners in Novel Noises," *Journal of the Acoustical Society of America*, vol. 139, pp. 2604–2612, 2016.

J. Chen, and D. L. Wang, "Long Short-Term Memory for Speaker Generalization in Supervised Speech Separation," in *Proc. Annual Conference of International Speech Communication Association (INTERSPEECH)*, pp 3314–3318, 2016.

J. Chen, and D. L. Wang, "DNN Based Mask Estimation for Supervised Speech Separation," in *Audio Source Separation*, Shoji Makino (ed.), Springer, to appear.

FIELDS OF STUDY

Major Field: Computer Science and Engineering

TABLE OF CONTENTS

Abstract ii

Dedication v

Acknowledgments vi

Vita viii

List of Tables xiii

List of Figures xv

CHAPTER: Page

1. INTRODUCTION 1

   1.1 Motivation 1
   1.2 Objectives 4
   1.3 Organization 6

2. BACKGROUND 8

   2.1 Speech Enhancement 8
   2.2 Model Based Separation 10
   2.3 Supervised Separation 11

3. ACOUSTIC FEATURES FOR SUPERVISED SPEECH SEPARATION AT LOW SNR CONDITIONS 14

   3.1 Introduction 14
   3.2 Evaluation Framework 16
   3.3 Existing Features 18

3.4 Multi-Resolution Cochleagram Feature . . . 23
3.4.1 Construction of MRCG . . . 23
3.4.2 Analysis of MRCG . . . 25
3.5 Feature Post-Processing and Combination . . . 27
3.5.1 Feature post-processing . . . 27
3.5.2 Feature combination . . . 28
3.6 Experimental Results . . . 29
3.6.1 Experimental setup . . . 29
3.6.2 Effect of ARMA filtering . . . 29
3.6.3 Comparison among individual features . . . 30
3.6.4 Feature combination results . . . 35
3.6.5 Comparison between MRCG and a complementary feature set 37
3.7 Discussion . . . 37

4. NOISE PERTURBATION FOR NOISE SEGMENT GENERALIZATION 40

4.1 Introduction . . . 40
4.2 System Overview . . . 41
4.3 Noise Perturbation . . . 43
4.3.1 Noise rate (NR) perturbation . . . 43
4.3.2 Vocal tract length (VTL) perturbation . . . 44
4.3.3 Frequency perturbation . . . 44
4.4 Experimental Results . . . 46
4.4.1 Experimental setup . . . 46
4.4.2 Parameters of noise perturbation . . . 49
4.4.3 Evaluation results and comparisons . . . 52
4.5 Concluding Remarks . . . 58

5. LARGE-SCALE TRAINING FOR NOISE TYPE GENERALIZATION AND SNR GENERALIZATION . . . 60

5.1 Introduction . . . 60
5.2 Method . . . 63
5.2.1 Stimuli . . . 63
5.2.2 Algorithm description . . . 64
5.2.3 Subjects . . . 66
5.2.4 Procedure . . . 68
5.3 Results and Discussion . . . 69
5.3.1 Predicted intelligibility results . . . 69
5.3.2 Actual intelligibility results . . . 74
5.4 General Discussion . . . 77
5.5 Conclusion . . . 80

6. MODELING TEMPORAL DYNAMICS FOR SPEAKER GENERALIZATION 82

6.1 Introduction 82
6.2 System Description 86
6.3 Experimental Setup 90
6.3.1 Data preparation 90
6.3.2 Optimization methods 91
6.4 Experimental Results and Analysis 92
6.4.1 Performance trend on seen test speakers 92
6.4.2 Performance trend on unseen test speakers 96
6.4.3 Model comparisons 97
6.4.4 Analysis of LSTM internal representations 99
6.4.5 Impact of future frames 100
6.5 Discussion 103

7. CONTRIBUTIONS AND FUTURE WORK 106

7.1 Contributions 106
7.2 Future Work 109

Bibliography 113

LIST OF TABLES

Table **Page**

3.1 Classification accuracy (in %) for six noises with ARMA post-processing at −5 dB. Boldface indicates best result. 31

3.2 HIT−FA (in %) for six noise types with ARMA post-processing at −5 dB, where FA is shown in parentheses. 32

3.3 HIT−FA (in %) during voiced intervals. 32

3.4 HIT−FA (in %) during unvoiced intervals. 32

3.5 Classification accuracy (in %) of combined feature with ARMA post-processing at −5 dB. 36

3.6 HIT−FA (in %) of combined feature with ARMA post-processing at −5 dB. 36

4.1 Comparison of DNN-based ratio masking (the baseline) with ASNA-NMF in terms of STOI (in %) for six noises at −5 dB. 53

4.2 Classification accuracy (in %) for six noises at −5 dB 53

4.3 HIT−FA rate (in %) for six noises at −5 dB, where FA is shown in parentheses. 53

4.4 STOI (in %) of separated speech for six noises at −5 dB, where STOI of unprocessed mixtures is shown in parentheses. 54

4.5 HIT−FA rate (in %) during voiced intervals, where FA is shown in parentheses. 54

4.6 HIT−FA rate (in %) during unvoiced intervals, where FA is shown in parentheses. 54

4.7 STOI (in %) of separated speech for five unmatched noises at −5 dB, where STOI of unprocessed mixtures is shown in parentheses. 58

5.1 Speech segregation results, for four test noises and their average, at −2 dB SNR measured in short-time objective intelligibility (STOI) values. 70

5.2 STOI values for speech mixed with (unprocessed), and segregated from (processed), babble and cafeteria noise at the SNRs indicated. 74

6.1 Comparison of the DNN and LSTM trained with 77 speakers in terms of the HIT−FA rate on the 6 seen speakers and unseen babble noise at −5 dB SNR. 92

LIST OF FIGURES

Figure **Page**

3.1 Diagram of the feature evaluation framework. 18

3.2 Effects of adding contextual information for speech separation with −5 dB babble. 26

3.3 Visualization of the MRCG feature. Left side shows MRCG features extracted from a mixture, while the right side shows MRCG features extracted from premixed clean speech. In CG2-4, feature patterns of the mixture resemble the ones of clean speech to some extent, indicating the MRCG feature could partially retain spectrotemporal patterns of speech in the presence of noise. 27

3.4 Effect of the ARMA post-processing order for the PLP feature with babble noise at −5 dB SNR. 30

3.5 Effects of ARMA filtering in terms of HIT−FA rate. 31

3.6 Median value and interquartile range of 50 test sentences for average performance on six noises. Results are shown for top four features in terms of classification accuracy and HIT−FA rate. 34

3.7 Average magnitudes of regression coefficients resulted from group Lasso for the cockpit noise. 35

3.8 Comparison of a complementary feature set (AMS+RASTA-PLP+MFCC) and the MRCG feature in terms of HIT−FA. 38

4.1 Diagram of the proposed system. 43

4.2 Illustration of noise rate perturbation. 44

4.3 (a) Mapping function for vocal tract length perturbation. The frequencies below a cutoff are stretched if $\alpha > 1$, and compressed if $\alpha < 1$. (b) Illustration of vocal tract length perturbation. The medium and low frequencies are compressed in this case. 45

4.4 Illustration of frequency perturbation. 45

4.5 The effect of the minimum noise rate γ_{min} for NR perturbation. . . . 51

4.6 The effect of the minimum wrapping factor α_{min} for VTL perturbation. 51

4.7 The effect of the perturbation intensity λ for frequency perturbation. 52

4.8 Average STOI (in %) of separated speech for six noises at -5 dB with respect to the number of training mixtures. 56

4.9 Mask comparisons. The top shows a ratio mask obtained from training on original noises, the middle shows a mask obtained from training on frequency perturbed noise, and the bottom shows the IRM. 57

4.10 The effect of frequency perturbation in three SNR conditions. The average STOI scores (in %) across six noises are shown for unprocessed speech, separated speech by training on original noises, and separated speech by training on frequency perturbed noises. 57

5.1 Pure-tone air-conduction audiometric thresholds for the listeners with sensorineural hearing impairment. Right ears are represented by circles and left ears are represented by Xs. Also displayed are subject number, listener age in years, and gender. 67

5.2 Visualization of 100 filters learned by the bottom hidden layer of a DNN trained on mixtures created using 10,000 noises. Each filter is shown in two dimensions: the abscissa represents time (23 frames) and the ordinate represents frequency (64 channels). 71

5.3 Short-time objective intelligibility (STOI) predictions for a noise-independent model trained and tested in matched and mismatched SNR conditions. 73

5.4 Segregation of an IEEE sentence (The lake sparkled in the red hot sun) from cafeteria noise at 0 dB SNR; (a) cochleagram of the utterance in quiet; (b) cochleagram of the utterance in noise; (c) IRM for this mixture; (d) estimated IRM for this mixture; and (e) cochleagram of the segregated utterance by applying the estimated IRM to the noisy utterance. 73

5.5 Intelligibility of IEEE sentences based on percentage of keywords reported. The top panels represent scores in, or segregated from, babble noise, and the bottom panels represent scores in, or segregated from, cafeteria noise, all at the SNRs indicated. Individual HI listeners are represented by filled symbols and individual NH listeners are represented by open symbols. Scores for unprocessed speech in noise are represented by circles and scores for algorithm-processed noisy speech are represented by triangles. Algorithm benefit is represented by the height of the line connecting these symbols. 76

5.6 Group-mean intelligibility scores and standard errors for HI and NH listeners hearing unprocessed IEEE sentences in noise and sentences following algorithm processing. The top panels show scores for a babble background and the bottom panels show scores for a cafeteria-noise background, at the SNRs indicated. 77

6.1 Performance of a speaker-dependent DNN on seen and unseen speakers with a babble noise in terms of STOI (in %) at -5 dB SNR. 84

6.2 Diagram of an LSTM block with three gates and a memory cell. . . . 88

6.3 Diagram of the proposed system. Four stacked LSTM layers are used to model temporal dynamics of speech. Three time steps are shown here. 88

6.4 Training and test errors of the DNN and LSTM as the number of training speakers increases. All models are evaluated with a test set of 6 seen speakers and a test set of 6 unseen speakers. Training mixtures are created with {6, 10, 20, 40, 77} speakers and 10,000 noises. The two test sets are created with the unseen babble noise at -5 dB SNR. All models are noise-independent. (a) Performance of the DNN on the 6 seen speakers. (b) Performance of LSTM on the 6 seen speakers. (c) Performance of the DNN on the 6 unseen speakers. (d) Performance of LSTM on the 6 unseen speakers. 93

6.5 Comparison of the DNN and LSTM in terms of STOI improvement (in %) with the unseen babble noise. (a) Performance of the DNN and LSTM on 6 seen speakers at −5 dB SNR. (b) Performance of the DNN and LSTM on 6 unseen speakers at −5 dB SNR. (c) Performance of the DNN and LSTM on 6 seen speakers at −2 dB SNR. (d) Performance of the DNN and LSTM on 6 unseen speakers at −2 dB SNR. 94

6.6 Visualization of the estimated masks by the DNN (top) and LSTM (middle) and the IRM (bottom). The mixture is created by mixing an unseen male speaker with the unseen babble noise at −5 dB SNR. . . 96

6.7 Comparison of speaker-set-dependent models (trained on 77 speakers and tested on 6 seen speakers) and speaker-dependent models in terms of STOI. Group means and standard errors are shown. 97

6.8 Comparison of speaker-independent models (trained on 77 speakers and tested on 6 unseen speakers) and speaker-dependent models in terms of STOI. Group means and standard errors are shown. 98

6.9 Visualization of speech patterns and memory cell values. Four utterances of two unseen speakers (male and female) are concatenated and mixed with the unseen babble noise at 0 dB SNR. The top four plots depict noisy speech cochleagram, clean speech cochleagram, the IRM and the estimated mask by LSTM, respectively. The bottom three plots show values of three different cells across time, respectively. . . 101

6.10 Impact of resetting the internal states of LSTM. The top five plots show the clean speech cochleagram, noise cochleagram, noisy speech cochleagram, the IRM and the estimated mask by LSTM, respectively. The 6th and 9th plots show the estimated masks when LSTM internal states are reset during speech-dominant intervals. The 7th and 8th plots show the estimated masks when LSTM internal states are reset during noise-dominant intervals. 102

6.11 Impact of future frames on the performance of the DNN and LSTM in terms of STOI improvement (in %). The input contains 11 past frames, a current frame and {0, 1, 2, 5, 8, 11} future frames. The models are evaluated with 6 unseen speakers and the unseen babble noise. (a) Performance of the DNN and LSTM at −5 dB SNR. (b) Performance of the DNN and LSTM at −2 dB SNR. 103

CHAPTER 1

INTRODUCTION

1.1 Motivation

Speech plays an essential role in human communication. In real environments, speech is usually corrupted by background noise and room reverberation. The human auditory system is remarkable in separating speech from interference. However, perception of speech in noise can be challenging for hearing-impaired listeners [24]. In United States, less than 25% of people who need hearing aids actually use them. One major criticism of hearing aids is that they amplify both speech and noise. It is desirable to develop speech separation algorithms for such devices. Another important application of speech separation is enhanced telecommunication. We are often asked to repeat ourselves during a phone call in a noisy environment. Cell phones with good speech separation performance have the potential to disrupt the market. The goal of speech separation is to improve speech intelligibility and quality, both of which are important for real-word applications.

The application of speech separation is not limited to human communication. Today's Internet connects people and smart devices. In the past decade, we observe increasing interactions between humans and devices such as Amazon Echo and Google

Home. For human-computer interactions, speech is the most natural one. There is a huge demand for implementing speech interfaces for home appliances and automobiles. These interfaces depend on robust automatic speech recognition (ASR). Speech separation helps these devices to recognize voice commands in our daily life, where ambient noise is almost always present. Given that the ASR performance on clean speech has come close to the human level, speech separation is perhaps one of the biggest challenges for deploying speech interfaces.

Because of its important applications, speech separation has been extensively studied in the speech processing community. Speech separation techniques can be categorized into two classes: monaural processing and microphone-array processing. Monaural separation is especially challenging as it only utilizes single-microphone recordings. However, monaural separation is more flexible in terms of deployment. Without spatial cues, monaural separation usually requires prior knowledge about speech and noise. In signal processing, there are mainly two approaches for monaural separation. The speech enhancement approach makes assumptions about the statistics of noise and speech, and does not perform well with nonstationary noises and at low signal-to-noise ratios (SNRs). The model based approach deals with speech separation by learning dictionaries or explicit models for speech and noise. However, they do not scale well with many noise types and speakers.

Besides the signal processing perspective, speech organization is studied in auditory perception. Research in auditory scene analysis (ASA) [10] suggests that auditory segregation consists of two stages: segmentation and grouping. The segmentation stage decomposes the input sound into time-frequency (T-F) segments, each containing a region of T-F units coming from a single source. The grouping

stage uses characteristics of sound, such as harmonicity, onset/offset and temporal continuity, to organize T-F segments into streams corresponding to different sources.

Inspired by the principles in ASA, computational auditory scene analysis (CASA) formulates speech separation as a mask estimation problem [115]. In a T-F representation of noisy speech, the ideal binary mask (IBM) assigns value 1 to a T-F unit if it is speech-dominant, 0 otherwise [44] [113]. Speech separation can be achieved by applying an estimated IBM to attenuate noise and preserve speech. Alternatively, instead of a binary decision on a T-F unit, a soft decision leads to the definition of the ideal ratio mask (IRM), which is calculated from the energy ratio of speech and noise in a T-F unit [101] [119].

With the ideal mask as the computational goal, speech separation can be formulated as a supervised learning problem [113]. This formulation is a milestone in speech separation for two reasons. First, powerful learning machines, such as deep neural networks (DNN), can be employed to learn the mapping from noisy acoustic features to the ideal mask. Instead of making assumptions on noisy speech, we learn speech and noise patterns from training data. Second, the training data for supervised speech separation is relatively easy to obtain compared to, e.g., ASR. Noisy speech can be simulated by mixing a speech signal with a noise signal at a certain SNR. The training label, i.e., the ideal mask, is easily calculated from the premixed speech and noise. Aided by advances in deep learning research, supervised speech separation has significantly pushed the state-of-the-art performance [120].

Like other supervised learning tasks, supervised speech separation must deal with the generalization issue. A test condition usually differs from training conditions.

The utility of supervised speech separation depends on its generalizability. For example, a model trained with a factory noise may not perform well on a restaurant noise. Generalization of supervised speech separation mainly depends on four factors: training targets, acoustic features, training data and learning machines. Masking based training targets, such as the IBM and the IRM, have been shown to generalize better than mapping based targets like clean spectral magnitude [119]. This dissertation discusses the impact of acoustic features, training data and learning machines on generalization of supervised speech separation.

1.2 Objectives

Supervised speech separation has been shown to outperform the traditional speech enhancement approach and model based approach. However, the performance of supervised speech separation is limited by its generalizability. This dissertation aims to develop a supervised speech separation system that generalizes to conditions that are not observed during training. The generalization issue is addressed in the following aspects:

- *Exploring acoustic features.* Acoustic features are essential for discriminating speech-dominant T-F units from noise-dominant ones. Supervised speech separation at low SNRs needs acoustic features that are noise-robust. In the last few decades, many acoustic features have been proposed for robust ASR. Acoustic features have also been used in speech separation and robust speaker recognition. We systematically evaluate an extensive list of acoustic features for supervised speech separation at low SNRs. By examining performance and principles

of different acoustic features, we plan to design a new feature specifically for supervised speech separation.

- *Investigating data augmentation for noise segment generalization.* To train a speech separation model for a specific noise/environment, we must deal with the mismatch between training noise segments and test noise segments. This issue is significant for nonstationary noises such as babble and cafeteria noises. With limited noise samples, learning machines likely overfit training segments and do not perform well on test segments. We investigate data augmentation techniques to expand training noise segments and improve the performance of noise-dependent models.

- *Exploring noise type generalization and SNR generalization.* Compared to noise segment generalization, noise type generalization is a more ambitious goal since there is a large variety of noises in the real world. We ask two questions. What kind of data is required for training a noise-independent model? Does the DNN have the capacity for noise type generalization? In addition, SNR generalization is investigated. We evaluate models with both objective speech intelligibility measures and subject tests.

- *Improving speaker generalization.* Besides noise generalization, it is important for a speech separation system to generalize to unseen speakers. A speaker-dependent model learns features for a specific speaker, whereas a speaker-independent model must deal with many speakers. In the latter case, the confusability of speech and noise increases. For example, it is challenging to differentiate target speech from a multi-talker babble noise. We investigate

whether a deep feedforward network is capable for speaker generalization. In addition, long-term contexts are explored for speaker generalization.

1.3 Organization

This dissertation is organized as follows. Chapter 2 provides a review of existing speech separation techniques including speech enhancement, model based methods and supervised speech separation.

In Chapter 3, we study the performance of acoustic features for supervised speech separation at low SNR conditions. The evaluated features include robust ASR features, robust speaker recognition features and speech separation features. In addition, we propose a new separation feature called the Multi-Resolution Cochleagram (MRCG). All features are evaluated using a fixed IBM estimator based on a multi-layer perceptron (MLP). We evaluate estimated masks with accuracy and hit minus false-alarm (HIT-FA) rate [64].

Chapter 4 presents the work on noise segment generalization. We study noise perturbation for data augmentation. Three perturbation methods, namely vocal tract length (VTL) perturbation, noise rate perturbation and frequency perturbation, are evaluated with a DNN based IRM estimator. We compare speech separation models trained with and without perturbed noises in terms of short time objective intelligibility (STOI) [105] and HIT-FA rate [64].

In Chapter 5, we investigate noise type and SNR generalization. The impact of the amount of training noises on generalization is studied. We train a fixed DNN based IRM estimator on 100 and 10,000 noises and test both models with four unseen

nonstationary noises. The noise-independent models are also compared with noise-dependent models in terms of STOI improvement. In addition, SNR generalization is evaluated. Finally, subject tests with normal-hearing and hearing-impaired listeners are carried out to evaluate the performance of a noise-independent model.

Chapter 6 discusses the issue of speaker generalization. We first evaluate the performance of a DNN based IRM estimator for speaker generalization. By increasing training speakers, we examine the performance trends for seen and unseen test speakers. We propose a speech separation model based on recurrent neural network (RNN) with long short-term memory (LSTM) to account for the temporal dynamics of speech. We compare the performance of the DNN and LSTM in terms of STOI improvement. Additional experiments are conducted to analyze the contextual information encoded in LSTM states and the impact of future information on mask estimation.

Chapter 7 summarizes contributions of this dissertation and discusses future work.

CHAPTER 2

BACKGROUND

In this chapter, we review existing speech separation approaches. First, we discuss the basics of the traditional speech enhancement approach and model based approach. Then, we introduce recently proposed supervised speech separation.

2.1 Speech Enhancement

Speech separation is a long-standing problem in signal processing. Over the past decades, many speech enhancement algorithms [72] have been developed. One popular method is spectral subtraction, which is originally proposed by Weiss et al. [121] and Boll [9]. The idea is to estimate clean speech by subtracting noise spectrum from mixture spectrum. Spectral subtraction algorithms typically use noisy phase for resynthesis since phase does not significantly degrade speech intelligibility [86]. Therefore, the problem becomes the estimation of noise magnitude or power. The noise estimate is usually computed from initial time frames or non-speech intervals of a signal. The key assumption of spectral subtraction is small noise variations, and it does work well for a nonstationary noise, whose spectrum changes over time. Another problem of spectral subtraction is that the estimated clean speech spectrum may contain negative values. Although algorithms [59] [73] [72] have been developed

to deal with these two issues, the performance of spectral subtraction algorithms degrades significantly for highly nonstationary noises.

Another popular speech enhancement method is Wiener filtering, which operates in the complex domain. The clean speech estimate $\hat{X}(w)$ is obtained by applying a filter $H(w)$ to noisy speech $Y(w)$:

$$\hat{X}(w) = H(w)Y(w) \tag{2.1}$$

Minimizing the mean-square error of estimated clean speech leads to the optimal Wiener filter:

$$H(w) = \frac{P_x(w)}{P_x(w) + P_d(w)} \tag{2.2}$$

where $P_x(w)$ and $P_d(w)$ denote power spectra of clean speech and noise, respectively. The key of Wiener filtering is to estimate the *a priori* SNR [94] [40], which is the ratio of $P_x(w)$ and $P_d(w)$. The calculation of the *a priori* SNR depends on $P_d(w)$, which is typically estimated by noise tracking algorithms. These algorithms typically assume that speech is more nonstationary than noise. However, this assumption does not hold for many noises.

Statistical speech enhancement makes assumptions about speech distribution given noisy observations. A representative algorithm is minimum mean-square error (MMSE) estimation [26] [28] [55]. The MMSE estimator minimizes the difference between estimated and true magnitudes of speech. Like Wiener filtering, the MMSE estimator requires an estimate of noise power spectrum, which is nontrivial for nonstationary noises.

2.2 Model Based Separation

In the model based approach, the structures of speech and noise are learned from data. Early works apply machine learning models to two-talker or multi-talker separation. Roweis [92] proposes to recover sound sources by a nonstationary reweighting of frequency sub-bands of a mixture. The weights are predicted by a factorial hidden Markov model (HMM) where each individual HMM models a single speaker. Bach and Jordan [6] apply spectral clustering on CASA based features to segment a mixture spectrogram into subsets, each of which represents a sound source. While the above methods separate one speaker from another, it is nontrivial to adapt them for speech-nonspeech separation since noise is less structured than speech and more difficult to model.

A representative method for model based speech-nonspeech separation is non-negative matrix factorization (NMF) [70], which models each source of a mixture using a basis matrix and a weight matrix. Therefore, a mixture is represented by the product of a concatenated basis matrix and a concatenated weight matrix:

$$Y = BW = [B_1, \ldots, B_n] \left[W_1^T, \ldots, W_n^T\right]^T \tag{2.3}$$

where B_k and W_k are non-negative basis matrix and weight matrix for source k, respectively. During training, a basis matrix is learned for each source. During inference, we keep B fixed and adjust W to minimize the reconstruction error for Y. Then, source k can be estimated as $B_k W_k$. With an overcomplete basis matrix B, a sparse solution for W is not guaranteed. Therefore, a penalty term is usually introduced to encourage a sparse solution. Application of NMF to speech separation requires the modeling of two sources: speech and noise [100] [111]. One disadvantage

of NMF is that noises, especially nonstationary ones, are difficult to model. Besides, NMF has a high computation complexity during inference, which impedes real-time applications.

2.3 Supervised Separation

Besides the speech enhancement and model based approaches, speech separation can also be treated as a supervised learning problem [113]. Early algorithms use an MLP to map a mixture segment to a speech segment in the time domain or spectral domain [107] [108] [126]. Those early works only use shallow neural networks and small training data, and have not demonstrated the full potential of the supervised approach.

In the past two decades, research in CASA has reignited the interest in supervised speech separation. Roman et al. [91] train a classifier to estimate the IBM for binaural speech separation. A maximum *a posteriori* (MAP) classifier is trained with two binaural features, namely interaural time differences (ITD) and interaural intensity differences (IID), to classify T-F units as speech-dominant or noise-dominant. This system produces a large improvement in speech intelligibility for matched training and test conditions. Seltzer et al. [97] apply a Bayessian classifier to predict and remove noise-dominant T-F units for robust ASR. Jin and Wang [56] train sub-band MLPs to classify T-F units as speech or noise dominant in the grouping stage of CASA based speech separation. Kim et al. [64] apply the Gaussian mixture model (GMM) for IBM estimation in the mel-spectral domain (see also [97]). With low SNRs and matched training and test noise segments, this method has been shown to improve speech intelligibility for normal-hearing listeners.

Three key components of supervised speech separation are training targets, learning machines and acoustic features [15]. The first proposed training target is the IBM, which is inspired by the auditory masking phenomenon in auditory perception. The IBM assigns the value 1 to speech-dominant T-F units and 0 otherwise (see Section 1.1). Subject tests have shown that ideal binary masking dramatically improves speech intelligibility for normal-hearing and hearing-impaired listeners [11] [71] [116]. Similar to the IBM, the target binary mask (TBM) [66] classifies T-F units by comparing target speech with the reference speech-shaped noise, and has also been shown to dramatically improve speech intelligibility. Alternatively, instead of a binary decision for a T-F unit, a soft decision leads to the definition of IRM [101] [82] [51], which has been shown to slightly improve speech quality over the IBM [119]. While IBM estimation is a classification problem, IRM estimation is a regression problem. Besides masking based targets, mapping based targets have also been used in supervised speech separation. Mapping based targets are typically T-F representations of clean speech, such as log spectrum. Although mapping based targets seem more straightforward, a recent study has shown that they tend to underperform masking based targets in terms of speech intelligibility and quality [119]. In this dissertation, we focus on speech separation systems using masking based targets.

Learning machines are crucial for supervised speech separation. DNNs have been very successful in many supervised learning tasks such as image classification [21] [37], ASR [31] [93] and machine translation [104] [125]. In 2013, Wang and Wang [120] introduced the DNN for supervised speech separation for the first time, and demonstrated substantial speech separation improvement over the previous state-of-the-art. In each sub-band, a DNN is trained to extract high-level features, which are sent to

a linear SVM for IBM estimation. The power of the DNN comes from its capability of learning hierarchical features. Going from the bottom layer to the top layer of a DNN, successive hidden activations represent more and more abstract features, which help separate classes that are difficult to separate in the input space. Two types of DNNs are commonly used for supervised speech separation. They are MLPs and RNNs. Their generalization capabilities are discussed later in this dissertation.

Acoustic features provide discriminative information for mask estimation. Early studies in supervised speech separation use binaural features, such as ITD and IID, for binaural separation [91]. Pitch based features [56] [118] and amplitude modulation spectrogram (AMS) features [64] are explored for monaural separation. A recent study investigates robust ASR features and speaker recognition features, including mel-frequency cepstral coefficient (MFCC), perceptual linear prediction (PLP) [41], relative spectral transform PLP (RASTA- PLP) [42] and gammatone frequency cepstral coefficient (GFCC) [99] [131], for monaural separation. To understand how various features perform at low SNRs, a systematic feature study is presented in this dissertation.

It is important to improve generalization of supervised speech separation since a test condition usually differs from training conditions. Three main aspects of generalization are noise generalization, SNR generalization and speaker generalization. This dissertation focuses on improving these generalization aspects by investigating acoustic features, data argumentation techniques and learning machines.

CHAPTER 3

ACOUSTIC FEATURES FOR SUPERVISED SPEECH SEPARATION AT LOW SNR CONDITIONS

This chapter studies acoustic features for supervised speech separation at low SNR conditions. The work presented in this chapter has been published in the *Proceedings of 2014 IEEE International Conference on Acoustic, Speech, and Signal Processing* [17] and *IEEE/ACM Transactions on Audio, Speech, and Language Processing* [16].

3.1 Introduction

The current formuation of supervised speech separation originates from CASA. The IBM is often considered as the computational objective of CASA [113]. Subject tests show that IBM separation dramatically improves speech intelligibility in noise for both normal-hearing and hearing-impaired listeners [11] [71] [116] [2]. The IBM is a T-F mask constructed from premixed speech and noise, and it is defined as follows.

$$\mathrm{IBM}(t,f) = \begin{cases} 1 & \text{if } \mathrm{SNR}(t,f) > \mathrm{LC} \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

where t denotes time and f denotes frequency. The IBM assigns the value 1 to a T-F unit if the local SNR within the unit exceeds a local criterion (LC), and 0 otherwise. The estimation of the IBM amounts to a binary classification problem where

supervised learning is employed to predict the label of each T-F unit [33]. Recent studies show that classification-based speech separation improves speech intelligibility for human listeners in background noise [64] [39].

The two key components of classification-based speech separation are acoustic features extracted from an input mixture and classifiers used for supervised learning. While previous studies have emphasized classifiers, the present study focuses on features. Our goal is to reveal how various features perform in classification-based speech separation. To obtain a fair comparison, we choose and fix an MLP as the classifier to simplify and speedup training, as we are mainly concerned with the relative performance [56]. In addition, we choose a set of six representative nonstationary noises and fix the evaluation SNR to -5 dB. This very low SNR level is selected with the goal of improving speech intelligibility in mind. It is well known that human listeners, even those with significant hearing loss, perform nearly perfectly unless the SNR is in the negative range [48] [80] [116].

In terms of features chosen for evaluation, since the classification approach is only recently established for speech separation, not many features have been developed for this task. We have therefore performed a systematic literature search for robust features published for ASR in noise, a task that is expected to be related to speech separation. Feature robustness has been extensively studied in the ASR literature. With low SNR and nonstationary noise in mind, we have selected a subset of promising features in our evaluation, such as relative autocorrelation sequence MFCC (RAS-MFCC), Gabor filterbank (GFB) features and power normalized cepstral coefficients (PNCC). These features, together with those previously investigated for speech separation [118], form the existing feature set. Based on our evaluation, we also propose

a new feature called Multi-Resolution Cochleagram (MRCG), specifically designed to achieve the best separation performance. Additionally, we investigate auto-regressive moving average (ARMA) filtering as a post-processing technique to enhance feature robustness for further improving speech separation performance.

We should point out that a recent study [118] has evaluated several features for classification-based speech separation. Our study goes beyond [118] in several aspects. First, our evaluation is conducted on more challenging noisy mixtures using a different classifier (MLP instead of support vector machine). More importantly, features are chosen more systematically in our study, which results in a significantly more expansive list. Finally, while the study in [118] emphasizes feature combination, our study results in a new, effective feature which performs better than the complementary feature set suggested in [118].

This chapter is organized as follows. Section 3.2 describes feature evaluation framework for classification-based speech separation. The existing features are described in Section 3.3. We introduce the proposed MRCG feature in Section 3.4. Section 3.5 covers feature post-processing and feature combination. We present experimental results in Section 3.6. Section 3.7 concludes the chapter.

3.2 Evaluation Framework

In classification-based speech separation, the computational goal typically is to estimate the IBM that is calculated from premixed signals. The time-frequency representation of a cochleagram is frequently used to construct the IBM. In this study, we use a 32-channel cochleagram with 20 ms frame length and 10 ms frame shift. The LC of the IBM is set to -10 dB to preserve enough speech information (see [39]). Note

that, once a binary mask is computed, it can be used to synthesize a time-domain signal by weighting T-F unit signals in an appropriate way (see Chapter 1 of [115] for more details).

Fig. 3.1 shows the diagram of the evaluation system, which consists of the feature extraction component and the MLP classification component. All mixtures are sampled at 16 kHz. We extract acoustic features from an input signal at the frame level, which are sent to an MLP classifier for IBM estimation. We use a full-band input signal for feature extraction and one MLP for predicting a mask across all channels. In other words, the MLP is trained to predict a T-F mask frame by frame as opposed to sub-band classification in [118].

The features are evaluated based on the mask estimation quality. There are several criteria for measuring the quality of an estimated IBM. One straightforward criterion is to compute classification accuracy, where the percentage of correctly labeled T-F units is calculated for the whole mask. However, this criterion is agnostic to different classification errors. Recent work shows that the HIT$-$FA criterion well correlates with human intelligibility [64], where HIT refers to the percentage of correctly classified target-dominant T-F units and FA refers to false alarm or the percentage of wrongly classified interference-dominant T-F units. A good IBM estimate should have high HIT and low FA, which leads to high HIT$-$FA rate. We use both classification accuracy and HIT$-$FA rate in this study.

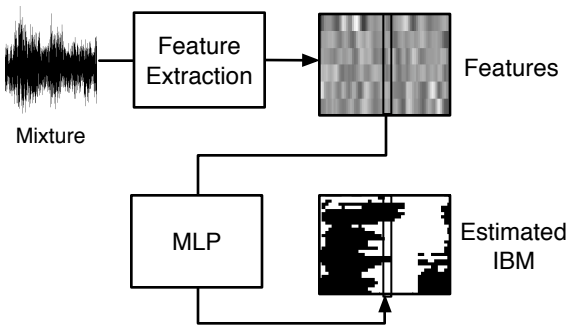


Figure 3.1: Diagram of the feature evaluation framework.

3.3 Existing Features

We evaluate an extensive list of existing acoustic features, consisting of widely used and promising robust speech recognition and separation features. Below we briefly describe a set of 16 such features, and more details can be found in the references.

- *Mel-frequency Cepstral Coefficient (MFCC).* To compute MFCC, an input signal is divided into 20 ms frames with 10 ms frame shift. We apply a Hamming window to each frame and derive power spectrum using short-time Fourier transform. Then we convert power spectrum into mel scale. Finally, log compression and discrete cosine transform (DCT) are applied to compute 31-dimensional (31-D) MFCC.

- *Perceptual Linear Prediction (PLP).* PLP is designed to minimize the differences between speakers while keeping important formant structure [41]. To compute PLP, the power spectrum of an input signal is converted into bark scale, followed by loudness preemphasis and applying intensity loudness law. Then we derive linear prediction coefficients, which are then converted to cepstral coefficients. By using the 12th order linear prediction model, we end up with 13-D PLP.

- *Relative Spectral Transform PLP (RASTA-PLP).* RASTA-PLP introduces RASTA filtering to PLP [42]. To compute RASTA-PLP, the power spectrum of an input signal is wrapped to the bark scale. The resulting spectrum is log-compressed and filtered with the RASTA filter, which emphasizes the modulation frequencies that are relevant to human speech. The filtered log-spectrum is then expanded by an exponential function. Finally, we perform linear prediction analysis to derive 13-D RASTA-PLP.

- *Gammatone Frequency Cepstral Coefficient (GFCC).* To compute GFCC [99] [131], we pass an input signal through a 64-channel gammatone filterbank to derive sub-band signals. Each sub-band signal is decimated to 100 Hz, amounting to 10 ms frame shift. We then apply cubic root compression to the magnitude of the decimated signals and perform DCT to derive 31-D GFCC.

- *Gammatone Frequency Modulation Coefficient (GFMC).* To compute GFMC [75], we first follow the GFCC procedure to compute 31-D GFCC. Then we calculate the modulation spectrum of each coefficient. The modulation spectrum corresponds to the Fourier transform of the temporal trajectory of each coefficient. We use 160 ms frame length and 10 ms frame shift to calculate the modulation spectrum. For each modulation spectrum, we calculate the energy for 2 - 16 Hz modulation frequencies, which are mostly relevant to speech signals [75]. Finally, we concatenate the energy calculated from each coefficient to form 31-D GFMC.

- *Gammatone Feature (GF).* We compute 64-D GF by following the GFCC procedure except that the DCT step is skipped.

- *Zero-crossings with Peak-amplitudes (ZCPA).* ZCPA is a speech recognition feature based on zero-crossings [63]. To compute ZCPA, an input signal is decomposed into sub-band signals by a 32-band gammatone filterbank. We divide each sub-band signal into 100 ms frames with 10 ms frame shift. For each frame, we calculate the intervals between every two upward zero-crossings. We classify each interval into 31 frequency bins where the frequency of an interval is the inverse of the interval. Then we identify the peak amplitude within each interval and add a nonlinear-compressed peak amplitude to the corresponding frequency bin. The frequency bins are accumulated across all sub-bands and form a histogram, i.e. 31-D ZCPA.

- *Relative Autocorrelation Sequence MFCC (RAS-MFCC).* RAS-MFCC is designed to suppress background noise by filtering in the autocorrelation domain [129]. To compute RAS-MFCC, we calculate one autocorrelation sequence for each frame of an input signal. A high pass filter is applied to the temporal trajectory of each dimension of autocorrelation sequences to suppress slow-varying components. The filtered autocorrelation sequences are treated as the input to the standard MFCC procedure to derive 31-D RAS-MFCC.

- *Autocorrelation Sequence MFCC (AC-MFCC).* AC-MFCC is also an autocorrelation feature. It reduces the interference from background noise by discarding low-lag autocorrelation coefficients [98], by assuming that the effect of the noise is usually concentrated in low-lag autocorrelation coefficients. To compute AC-MFCC, an input signal is divided into frames where the autocorrelation of each

frame is computed. We discard low-lag, i.e. less than 2 ms, autocorrelation coefficients. Hamming window is applied to high-lag autocorrelation coefficients and the corresponding magnitude spectrum is computed. The remaining steps follow the MFCC procedure to derive 31 cepstral coefficients.

- *Phase Autocorrelation MFCC (PAC-MFCC).* PAC-MFCC is an ASR feature similar to RAS-MFCC. PAC-MFCC computes the angle between a signal and its shifted version [53]. It is assumed that angle sequences are less variant than autocorrelation sequences in the presence of background noise. The standard MFCC procedure is applied to the resulting angle sequences to compute 31-D PAC-MFCC.

- *Power Normalized Cepstral Coefficients (PNCC).* PNCC is a recent ASR feature that utilizes medium-time processing to mitigate noise corruption and employ power-law compression instead of log compression in traditional features [61]. First, the power spectrum of an input signal is integrated using gammatone frequency integration. Then, based on medium-duration temporal analysis, we perform asymmetric filtering and temporal masking to subtract background noise. Finally we apply power-law nonlinearity and DCT to derive 31 coefficients.

- *Gabor Filterbank (GFB) Features.* GFB is a recent feature designed for robust ASR by taking into account the spectrotemporal modulation frequencies [95]. To derive GFB, we compute the log mel-spectrum from an input signal. The spectrum is filtered by a Gabor filterbank which consists of 41 carefully designed

Gabor filters. Representative channels of each filtered spectrum are selected and concatenated to form 311-D GFB.

- *Amplitude Modulation Spectrogram (AMS).* The AMS feature is a feature used in speech separation [64]. To compute AMS, the full-wave rectified envelope of an input signal is decimated by a factor of 4. As in [64], AMS features are extracted from 32-ms frames (frame shift is still 10 ms). We apply Hamming window and 256-point FFT. Finally, the 15-D feature is derived by integrating the FFT magnitudes using 15 triangular windows uniformly centered from 15.6 to 400 Hz.

- *Pitch-based Features (PITCH).* Pitch-based features are used in a recent separation study [118]. These are T-F unit level features derived from pitch analysis. We calculate a cochleagram for an input signal and derive six features described in [118] (see also [47]) for each T-F unit. These features capture how likely a T-F unit is dominated by the target speech by utilizing periodicity and instantaneous frequency. In our classification-based speech separation, the ground truth pitch is used during training while the pitch estimated by a recently proposed robust pitch tracker, PEFAC [30], is used during testing.

- *Delta-spectral Cepstral Coefficient (DSCC).* DSCC is an ASR feature very similar to MFCC except that a delta operation is applied to the spectrum [68]. To compute DSCC, we first follow the standard MFCC procedure to compute the mel-spectrum. Then a delta operation is applied to derive delta spectral features, whose histogram is normalized to give a Gaussian distribution. DCT

is applied to compute 31 cepstral coefficients, based on which we further derive 31-D delta cepstral coefficients. Finally, we add traditional MFCC cepstral coefficients to form 93-D DSCC.

- *Suppression of Slowly-varying Components and the Falling Edge of The Power Envelope (SSF).* SSF has been designed to remove slowly-varying components to reduce noise interference and suppress the falling edge of power envelope in order to mitigate reverberation [62]. An input signal is divided into 50 ms medium-duration frames with 10 ms frame shift. The FFT of each frame is integrated across frequencies using gammatone weighting functions. Then we apply SSF processing to the resulting power spectrum. The SSF procedure produces an enhanced version of the original signal. We apply the MFCC procedure to the enhanced version to derive 31-D SSF.

3.4 Multi-Resolution Cochleagram Feature

Besides the existing features, we propose a new acoustic feature called the Multi-Resolution Cochleagram (MRCG), which encodes multi-resolution power distributions in the time-frequency representation of a signal. We combine four cochleagrams at different resolutions to construct the MRCG feature. A high resolution cochleagram captures the local information while three low resolution cochleagrams capture spectrotemporal contexts at different scales.

3.4.1 Construction of MRCG

The construction of MRCG is based on the cochleagram representation, which is widely used in the CASA literature [115]. To compute the cochleagram, we first

pass an input signal to a gammatone filter bank, where the impulse response of a particular gammatone filter is [89],

$$g_{f_c}(t) = t^{N-1} \exp[-2\pi t b(f_c)] cos(2\pi f_c t) u(t), \tag{3.2}$$

where f_c denotes the center frequency, N the filter order, and $u(t)$ the step function. The function $b(f_c)$ decides the bandwidth given f_c. To imitate human auditory filters, the center frequencies f_c are uniformly spaced on the equivalent rectangular bandwidth ($\mathrm{ERB_N}$) scale. The relation between $b(f_c)$ and f_c is shown in Equation 3.3.

$$b(f_c) = 1.019 * \mathrm{ERB_N}(f_c) = 1.019 * 24.7 * (4.37 * f_c/1000 + 1). \tag{3.3}$$

The bandwidth $b(f_c)$ increases as f_c increases, leading to higher resolutions at low frequencies and lower resolutions at high frequencies on the linear frequency scale. After getting response signals from the gammatone filterbank, we divide each response signal into 20 ms frames with a 10 ms frame shift. We derive the cochleagram by computing the power of each frame at each channel [115].

Each T-F unit in the cochleagram contains only local information, which may not be sufficient for estimating the mask. To compensate for this, the MRCG feature provides contextual information by including the power distribution in the neighborhood of each T-F unit. The MRCG feature is similar to the GFB feature in the sense that both are designed to encode the spectrotemporal context systematically (see also [45] [83]).

The steps for computing MRCG are described as follows.

1. Given an input mixture, compute the first 64-channel cochleagram, CG1. A log operation is applied to each T-F unit.

2. Similarly, compute CG2 with the frame length of 200 ms and frame shift of 10 ms.
3. CG3 is derived by averaging CG1 across a square window of 11 frequency channels and 11 time frames centered at a given T-F unit. If the window goes beyond the given cochleagram, the outside units take the value of zero (i.e. zero padding).
4. CG4 is computed in a similar way to CG3, except that a 23×23 square window is used.
5. Concatenate CG1-4 to obtain the MRCG feature, which has 64×4 dimensions for each time frame.

Note that, although the IBM is defined using a 32-channel cochleagram, features can be extracted from a different sized cochleagram (see Section 3.2). We found that 64-channel features extracted in Step 1 perform a little better than 32-channel features. Also, using zero padding in Step 3 for outside T-F units leads to slightly better results than simply averaging the units inside a window.

3.4.2 Analysis of MRCG

In the MRCG feature, CG1 contains the local information embedded in a typical cochleagram while CG2-4 provide fine-grain and coarse-grain contexts. The parameters used in the construction of MRCG are decided experimentally as follows. First, the frame length of CG1 is chosen to match the frame length of the IBM. Then we fix CG1 and determine CG2 by expanding to different frame lengths to select the best length. Similarly, we decide the size of the averaging window for CG3, and then for

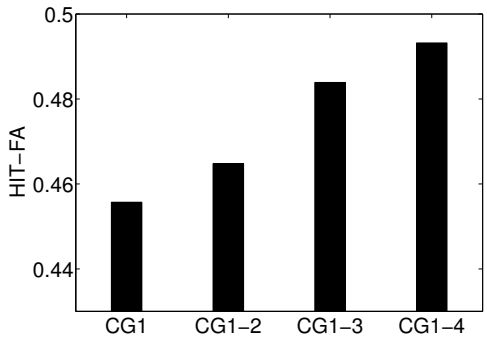


Figure 3.2: Effects of adding contextual information for speech separation with −5 dB babble.

CG4. After obtaining CG1-4, we find that adding more cochleagrams does not provide further performance improvements. Fig. 3.2 illustrates the effects of adding T-F contexts on the separation results. As shown in Fig. 3.2, adding CG2-4 consistently improves the results for babble noise at −5 dB SNR. Similar trends are observed for the other noises.

A visualization of the MRCG feature is given in Fig. 3.3, where the left plots features extracted from a babble mixture at −5 dB SNR and the right from the corresponding clean speech. As shown in Fig. 3.3, CG1 is the regular cochleagram, CG2 captures temporal context, CG3 encodes relatively small spectrotemporal context and CG4 encodes relatively large spectrotemporal context. The broad rationale behind MRCG is that a T-F unit is more likely to be speech-dominant if it resides in a cluster of many speech-dominant T-F units. In other words, a speech-dominant T-F unit not likely appears alone in a cochleagram.

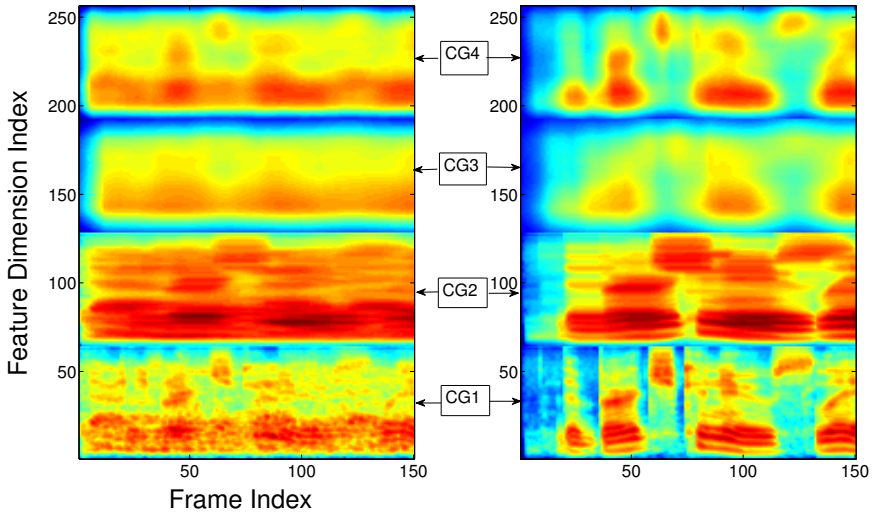


Figure 3.3: Visualization of the MRCG feature. Left side shows MRCG features extracted from a mixture, while the right side shows MRCG features extracted from premixed clean speech. In CG2-4, feature patterns of the mixture resemble the ones of clean speech to some extent, indicating the MRCG feature could partially retain spectrotemporal patterns of speech in the presence of noise.

3.5 Feature Post-Processing and Combination

3.5.1 Feature post-processing

In speech processing, delta (Δ) and double-delta ($\Delta\Delta$) features are widely used to capture temporal dynamics. Adding those features is a popular feature post-processing technique. For example, Δ+$\Delta\Delta$+MFCC yields better speech recognition results than MFCC alone. Recent research shows that Δ and $\Delta\Delta$ features also improve speech separation results [118]. In this study, we thus expand each feature by adding Δ and $\Delta\Delta$ features.

It has been suggested that applying ARMA filtering to mean variance normalized features improves speech recognition results [13]. The ARMA filter is defined below,

$$\bar{C}(m) = \frac{\bar{C}(m-M) + \cdots + \bar{C}(m-1) + C(m) + \cdots + C(m+M)}{2M+1} \tag{3.4}$$

where $C(m)$ denotes the feature vector at frame m, $\bar{C}(m)$ denotes the filtered feature vector at frame m and M denotes the order of the filter. The idea behind ARMA filtering is to smooth temporal trajectory of each feature dimension so that the interference of background noise is reduced. However, the effect of ARMA filtering in classification-based speech separation is unknown. In this study, we add ARMA filtering as an optional post-processing step and evaluate if it improves speech separation results.

3.5.2 Feature combination

A recent study shows that a proper combination of features can lead to better performance in classification-based speech separation [118]. A straightforward way of finding complementary features is to try all combinations of features. However, the number of combinations is exponential with respect to the number of features. As in [118], we utilize group Lasso (least absolute shrinkage and selection operator) to quickly identify complementary features. The idea of group Lasso is to impose ℓ_1/ℓ_2 mixed norm regularization on logistic regression. It is known that ℓ_1/ℓ_2 regularization leads to sparsity between groups (i.e. feature types) [77]. Group Lasso solves the following optimization problem:

$$\hat{\beta}_\lambda = \arg\min_{\beta,\alpha} \sum_i \log\left(1+\exp\left(-y_i\left(\beta^T x_i+\alpha\right)\right)\right) + \lambda \sum_{g=1}^{G} \left\|\beta_{\mathcal{I}_g}\right\|_2 \tag{3.5}$$

where x_i is an input feature vector, y_i is its label (taking value of 1 or -1), β denotes the response coefficients which we use to identify complementary groups, $\mathcal{I}_g$ denotes the index set of the gth group, $\|\cdot\|_2$ refers to ℓ_2 norm, and λ controls group sparsity. We minimize both the first term, which represents the classification error,

and the second term, which imposes ℓ_1/ℓ_2 mixed norm regularization. The input to the logistic regression is the concatenation of all feature types where the training labels are provided by the IBM. The regression is carried out channel by channel. The resulting response coefficients are averaged across channels. The features that have relatively large responses are selected as the complementary features.

3.6 Experimental Results

3.6.1 Experimental setup

In our experiments, we create mixtures using the IEEE corpus recorded by a male speaker [52] and six types of nonstationary noise from the NOISEX corpus [110]. The noise types include factory floor noise (Factory), speech babble (Babble), jet cockpit noise (Cockpit), destroyer engine room noise (Engine), military vehicle noise (Vehicle), and tank noise (Tank). The duration of each noise is about 4 minutes. Each mixture is created from one IEEE sentence and one noise type at -5 dB SNR. To create the training set, we use 480 IEEE sentences and the first half of each noise. As for the test set, we use another 50 IEEE sentences and the second half of the noises. Using different parts of a nonstationary noise ensures that the noise segments used in the test set are different from those in the training set. We train and test on the same type of noise. An MLP with one hidden layer is used as the classifier for IBM estimation. The hidden layer includes 300 sigmoidal activation units. We set aside 50 mixtures from the training set as a cross validation set for early stopping.

3.6.2 Effect of ARMA filtering

We first examine the effect of ARMA filtering, a feature post-processing technique, on every feature type. The only tunable parameter in the ARMA filter is the

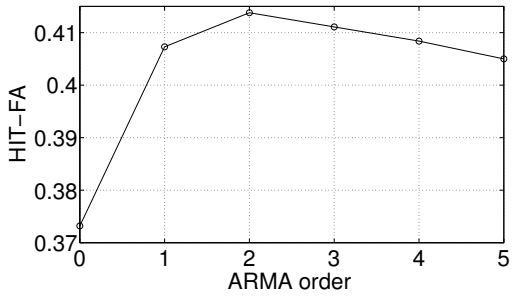


Figure 3.4: Effect of the ARMA post-processing order for the PLP feature with babble noise at −5 dB SNR.

filter order. The experimental results show that 2nd order ($M = 2$) ARMA filtering improves the HIT−FA rate for most feature and noise types. For example, the effect of filter order for the PLP feature with babble noise is shown in Fig. 3.4, where one can see the HIT−FA rate peaks when $M = 2$, and is significantly better than without using ARMA ($M = 0$). In the following experiments, we set ARMA filter order to 2.

Fig. 3.5 shows the effects of ARMA filtering on MRCG, GFCC, MFCC and PLP in each noise condition. The MRCG feature does not benefit from ARMA filtering, likely because the averaging windows used in MRCG have already embodied spectrotemporal smoothing. On average we observe 4% improvement in HIT−FA due to ARMA filtering for all noise types.

3.6.3 Comparison among individual features

Due to its effectiveness, we apply ARMA filtering to all 16 feature types plus MRCG in our comparisons. For the 50 test sentences, the overall classification accuracy and the overall HIT−FA rate of each feature are shown in Table 3.1 and Table 3.2, respectively, in decreasing order of average performance. In addition, Fig. 3.6 shows the median and interquartile range for the test sentences for the top four

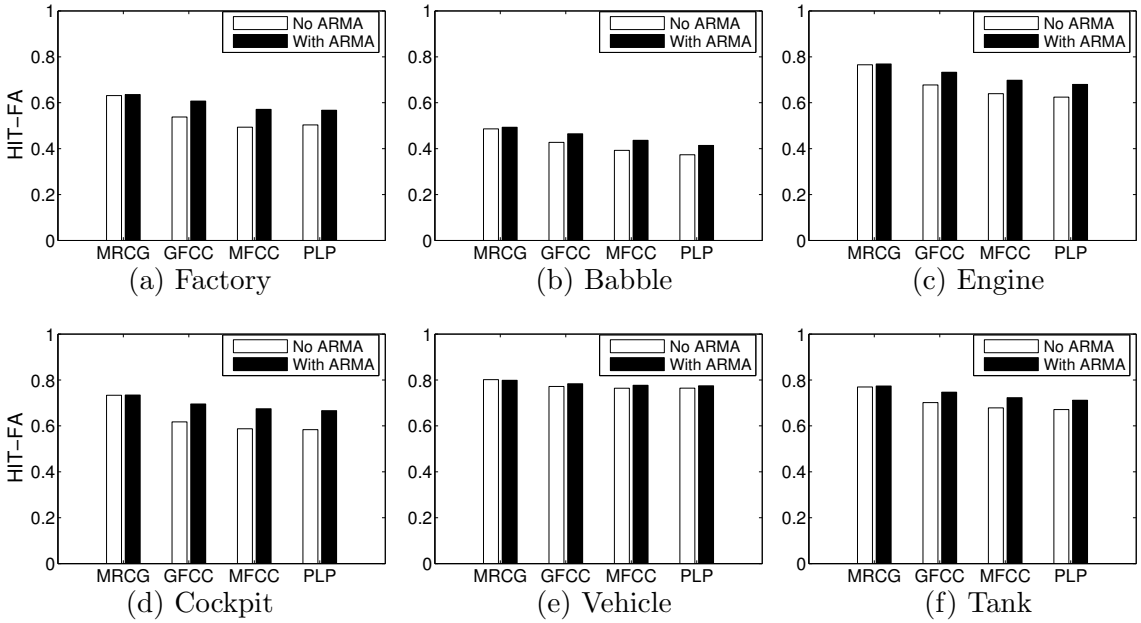


Figure 3.5: Effects of ARMA filtering in terms of HIT−FA rate.

Table 3.1: Classification accuracy (in %) for six noises with ARMA post-processing at −5 dB. Boldface indicates best result.

Feature \ **Noise**	Factory	Babble	Engine	Cockpit	Vehicle	Tank	Average
MRCG	**88.0**	**79.5**	**92.2**	**92.4**	**89.9**	**90.5**	**88.8**
GF	87.6	77.4	91.9	92.1	89.9	90.2	88.2
GFCC	87.7	78.3	91.3	91.9	89.2	89.7	88.0
DSCC	86.6	77.2	90.5	90.9	88.8	88.8	87.1
MFCC	86.5	77.5	90.2	91.1	88.8	88.6	87.1
PNCC	86.6	77.2	90.1	90.9	88.6	88.3	87.0
PLP	86.9	77.4	89.5	90.9	88.7	88.2	87.0
AC-MFCC	86.7	77.0	89.3	90.5	88.7	88.1	86.7
RAS-MFCC	86.9	76.9	89.4	90.9	87.8	88.1	86.7
GFB	86.3	74.5	89.3	90.9	87.6	87.6	86.0
ZCPA	85.4	75.2	89.6	90.5	87.4	87.7	86.0
SSF	85.7	75.6	89.0	89.5	88.2	87.4	85.9
RASTA-PLP	85.9	75.9	88.2	89.7	87.9	86.8	85.7
GFMC	84.1	74.3	87.5	89.1	83.5	83.7	83.7
PITCH	85.5	69.6	84.8	88.9	79.2	82.3	81.7
AMS	82.5	74.0	84.8	87.8	75.4	79.1	80.6
PAC-MFCC	77.9	69.8	78.1	81.1	70.8	67.9	74.3

Table 3.2: HIT−FA (in %) for six noise types with ARMA post-processing at −5 dB, where FA is shown in parentheses.

Feature \ Noise	Factory	Babble	Engine	Cockpit	Vehicle	Tank	Average
MRCG	**63** (7)	**49** (13)	**77** (4)	**73** (4)	**80** (10)	**77** (6)	**70** (7)
GF	61 (7)	45 (15)	75 (4)	71 (3)	80 (10)	76 (6)	68 (8)
GFCC	61 (6)	46 (14)	73 (4)	70 (3)	78 (11)	74 (6)	67 (7)
DSCC	56 (7)	42 (14)	70 (5)	66 (3)	77 (11)	73 (6)	64 (8)
MFCC	57 (7)	43 (14)	69 (5)	67 (4)	77 (11)	72 (7)	64 (8)
PNCC	56 (6)	44 (14)	69 (5)	66 (4)	77 (11)	71 (7)	64 (8)
PLP	56 (6)	41 (12)	68 (5)	66 (4)	77 (11)	71 (7)	63 (8)
AC-MFCC	56 (6)	42 (14)	67 (5)	65 (4)	77 (11)	71 (7)	63 (8)
RAS-MFCC	57 (6)	41 (14)	68 (5)	66 (4)	76 (11)	71 (7)	63 (8)
GFB	57 (7)	41 (18)	67 (5)	66 (4)	75 (12)	70 (7)	63 (9)
ZCPA	55 (8)	40 (16)	68 (5)	65 (4)	75 (13)	70 (8)	62 (9)
SSF	54 (7)	39 (15)	67 (5)	60 (4)	76 (11)	69 (7)	61 (8)
RASTA-PLP	52 (6)	38 (15)	64 (5)	61 (4)	76 (12)	67 (7)	60 (8)
GFMC	48 (7)	35 (15)	61 (6)	60 (5)	67 (17)	59 (9)	55 (10)
PITCH	46 (3)	29 (22)	50 (5)	50 (2)	59 (16)	53 (7)	48 (9)
AMS	40 (6)	27 (9)	49 (5)	52 (4)	50 (31)	45 (11)	44 (11)
PAC-MFCC	17 (5)	11 (8)	30 (9)	29 (7)	40 (48)	21 (17)	25 (16)

Table 3.3: HIT−FA (in %) during voiced intervals.

Feature \ Noise	Factory	Babble	Engine	Cockpit	Vehicle	Tank	Average
MRCG	**67**	**46**	**78**	**76**	**73**	**77**	**70**
GF	66	43	76	75	73	76	68
GFCC	66	45	75	73	72	75	68
MFCC	61	41	71	71	71	72	65
RAS-MFCC	61	39	70	70	68	71	63

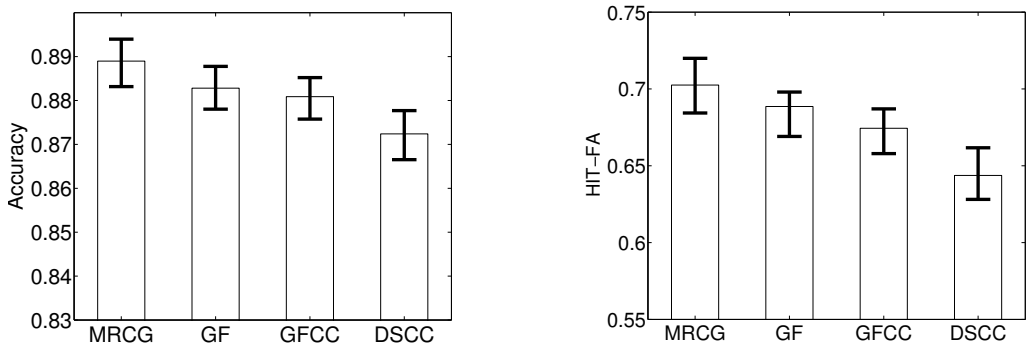
Table 3.4: HIT−FA (in %) during unvoiced intervals.

Feature \ Noise	Factory	Babble	Engine	Cockpit	Vehicle	Tank	Average
MRCG	**36**	**39**	**63**	**49**	**74**	**62**	**54**
GF	30	33	60	42	74	59	50
GFCC	28	31	56	40	73	55	47
MFCC	26	30	54	38	72	54	46
RAS-MFCC	25	30	50	38	68	51	44

features from Tables 3.1 and 3.2. The features can be roughly categorized into the following groups:

1. Gammatone-domain features: MRCG, GF and GFCC
2. Autocorrelation-domain features: RAS-MFCC, PAC-MFCC and AC-MFCC
3. Modulation-domain features: GFMC, AMS, GFB, and RASTA-PLP
4. Linear prediction features: PLP
5. MFCC variants: MFCC and DSCC
6. Medium-time processing features: PNCC, SSF
7. Zero-crossing feature: ZCPA
8. Pitch-based feature: PITCH

The results indicate that the gammatone-domain features (MRCG, GF, GFCC) perform better than other features. It is interesting to note that, although the modulation-domain feature GMFC is derived from GFCC, it does not perform as well as GFCC. Also interesting is that GFCC is a compact representation of GF, but the latter performs better than GFCC, probably because GF contains more information that can be exploited by the MLP classifier. MFCC, perhaps the most widely used feature, performs reasonably well when it is processed with an ARMA filter. Among the autocorrelation-domain features, RAS-MFCC performs the best and PAC-MFCC the worst. The performance of the pitch-based feature is poor largely due to the difficulty in pitch estimation at -5 dB.



(a) Accuracy (b) HIT−FA

Figure 3.6: Median value and interquartile range of 50 test sentences for average performance on six noises. Results are shown for top four features in terms of classification accuracy and HIT−FA rate.

The proposed MRCG feature performs the best in terms of both classification accuracy and the HIT−FA rate. It is worth mentioning that GFB is also a multi-resolution feature where filters of different sizes are applied to the spectrogram. However, MRCG performs significantly better than GFB.

The differences among various features are more obvious when they are tested on the babble noise or the factory noise, which are more challenging than the other four noises. Observe that the relative performance of different features is mostly consistent from one noise to another.

In addition, we examine the performance of features separately during voiced intervals and unvoiced intervals. Unvoiced speech is more susceptible to background noise due to relatively weak energy [46]. Table 3.3 and Table 3.4 show the performance of six relatively good features during voiced intervals and unvoiced intervals respectively. Again, the MRCG feature produces the best results during both voiced intervals and unvoiced intervals.

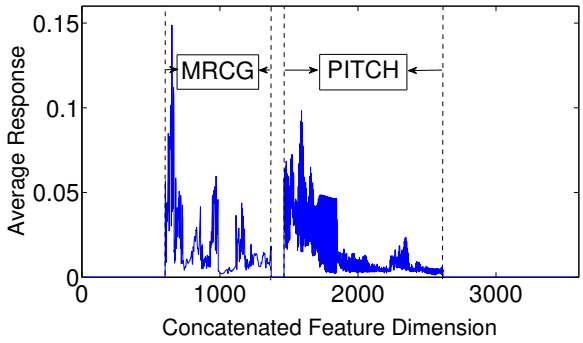


Figure 3.7: Average magnitudes of regression coefficients resulted from group Lasso for the cockpit noise.

To further validate the relative performance of features, we also evaluate three top features with different classifier — a linear SVM — that performs IBM estimation channel by channel [120]. Note that the input feature vector to each channel SVM is the same across different frequency channels. The average SVM classification accuracy for the six noises is 84.3%, 83.3%, and 82.4%, for MRCG, GF, and GFCC, respectively. The corresponding HIT−FA results are 66%, 63%. and 62%, for MRCG, GF, and GFCC, respectively. These SVM classification results show the same order of feature effectiveness as with MLP classification.

3.6.4 Feature combination results

We apply group Lasso to select complementary features for each noise type. Each feature type is appended with Δ and $\Delta\Delta$ features, as mentioned in Section 3.5.1. The group Lasso results for the cockpit noise are shown in Fig. 3.7. The average responses indicate discriminative power of a feature type. A good feature type is expected to show prominent responses. In Fig. 3.7, MRCG and PITCH have relatively high average responses while others have nearly no response, indicating that MRCG and

Table 3.5: Classification accuracy (in %) of combined feature with ARMA post-processing at −5 dB.

Feature	Factory	Babble	Engine	Cockpit	Vehicle	Tank	Average
MRCG	88.0	79.5	92.2	92.4	89.9	90.5	88.8
MRCG + PITCH (Estimated)	87.1	74.6	90.7	91.1	89.1	88.5	86.9
MRCG + PITCH (True)	90.8	85.7	92.3	93.2	90.5	90.7	90.5

Table 3.6: HIT−FA (in %) of combined feature with ARMA post-processing at −5 dB.

Feature	Factory	Babble	Engine	Cockpit	Vehicle	Tank	Average
MRCG	63	49	77	73	80	77	70
MRCG + PITCH (Estimated)	53	40	71	63	78	71	63
MRCG + PITCH (True)	70	64	77	76	81	78	74

PITCH are complementary. As for the other noise types, MRCG and PITCH are also identified by group Lasso as complementary features.

Table 3.5 and Table 3.6 show the classification accuracy and the HIT−FA rate for the combined feature (MRCG concatenated with PITCH), respectively. When we use ground truth pitch for training and estimated pitch for testing, the combined feature performs worse than MRCG alone. This is mainly because pitch estimation at −5 dB SNR is very challenging and the estimated pitch tends to be very different from the ground truth one. If we use ground truth pitch in both training and testing, the combined feature performs better than MRCG alone, especially for the factory and babble noise. If we use estimated pitch in both training and testing, the combined feature performs almost the same as MRCG alone.

3.6.5 Comparison between MRCG and a complementary feature set

In [118], it is found that AMS, RASTA-PLP, and MFCC form a complementary feature set and their combination outperforms each individual feature alone. Now we compare this complementary feature set and the MRCG feature for the aforementioned six noises at -5 dB SNR. As shown in Fig. 3.8, MRCG alone outperforms AMS+RASTA-PLP+MFCC. Such improvement mainly comes from the contextual information encoded in MRCG, which is important for separation in very low SNR conditions.

3.7 Discussion

In this study, we have evaluated an extensive list of acoustic features specifically for the classification-based speech separation at the very low SNR level of -5 dB — a condition where speech intelligibility is a main concern. In terms of classification accuracy and HIT$-$FA, we have shown that the gammatone-domain features (including GF, GFCC, MRCG) perform better than other features. The modulation-domain features (including GFMC and AMS) perform worse than most of the features likely because they do not deal with strong nonstationary noises well.

In addition, we have proposed a new feature, MRCG, which captures both local information and spectrotemporal contexts at different scales. The MRCG feature performs the best among the evaluated features. A closer look reveals that MRCG consistently produces the best results during both voiced and unvoiced intervals.

We have explored the effect of ARMA post-processing and found that the second order ARMA filtering improves most of the evaluated features by smoothing the

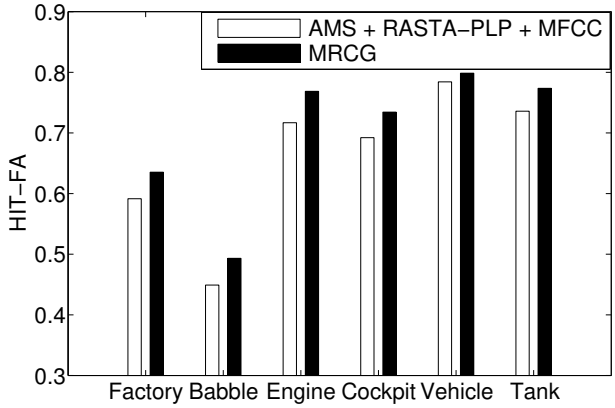


Figure 3.8: Comparison of a complementary feature set (AMS+RASTA-PLP+MFCC) and the MRCG feature in terms of HIT−FA.

temporal trajectories of feature dimensions. By employing group Lasso, we find that the MRCG feature and the pitch-based features form the best feature combination. Experimental results show that this combination yields the best performance if ground truth pitch is used. However, pitch estimation at -5 dB SNR is very difficult, and hence this insight of feature complementarity is not very useful unless pitch estimation improves substantially in very low SNR conditions. Our systematic study results in a clear recommendation: the simple MRCG feature without ARMA filtering should be considered as a benchmark in future speech separation studies, particularly at low SNR levels where human speech intelligibility is less than perfect.

It is noteworthy that PITCH and AMS features are among the first used in classification-based speech separation [56] [64]; a subsequent study combines these two [33]. Our investigation demonstrates that these are among the worst features for speech separation.

Features are of foundational importance for supervised separation. As embodied by the popularity of MFCC, progress in uncovering new and effective features often

lifts performance for a variety of tasks. Another example is GFCC which was first introduced for robust speaker identification [99] but has since been shown to be effective for robust ASR [7] and speech separation in [118] and here. Indeed a recent study found that MRCG outperforms a combination of 11 commonly used features for voice activity detection (VAD) [130]. Given the relationship between speech separation and robust ASR, we conjecture that MRCG is an effective feature for ASR in very noisy conditions. This conjecture obviously remains to be verified in future study.

Finally we emphasize that the focus of this study is on features, not classifiers. The MLP with one hidden layer unlikely represents the state-of-the-art in supervised speech separation, and DNNs with multiple hidden layers likely perform better [120]. Producing the best performing speech separation system is not the direct objective of this study, and such a system would require both effective features and effective classifiers. With that said, it is worth noting that the superior VAD performance of MRCG shown in [130] is consistently demonstrated with different DNN classifiers. In a recent study [23], MRCG is also shown to outperform many acoustic features for DNN based monaural speech separation in reverberant conditions.

CHAPTER 4

NOISE PERTURBATION FOR NOISE SEGMENT GENERALIZATION

This chapter presents noise perturbation techniques to improve segment generalization for the same noise type. The work presented in this chapter has been published in the *Proceedings of 2015 International Conference on Latent Variable Analysis and Signal Separation* [18] and *Speech Communication* [19].

4.1 Introduction

Supervised speech separation is a data-driven method where one expects a mask estimator to generalize from limited training data. However, training data only partially captures the true data distribution, thus a mask estimator can overfit training data and do a poor job in unseen scenarios. In supervised speech separation, a training set is typically created by mixing clean speech and noise. When we train and test on a nonstationary noise such as a cafeteria noise, there can be considerable mismatch between training noise segments and test noise segments, especially when the noise resource used for training is restricted. Similar problems can be seen in other supervised learning tasks such as image classification where the mismatch of training images and test images poses a great challenge. In image classification, a common

practice is to transform training images using distortions such as rotation, translation and scaling, in order to expand the training set and improve generalization of a classifier [69, 21]. We conjecture that supervised speech separation can also benefit from training data augmentation.

In this study, we aim at expanding the noise resource using noise perturbation to improve supervised speech separation. We treat noise expansion as a way to prevent a mask estimator from overfitting the training data. A recent study has shown that speech perturbation improves ASR [60]. However, our study perturbs noise instead of speech since we focus on separating target speech from highly nonstationary noises where the mismatch among noise segments is the major problem. To our knowledge, our study is the first to introduce training data augmentation to the domain of speech separation.

This chapter is organized as follows. Section 4.2 describes the system used for mask estimation. Noise perturbations are covered in Section 4.3. We present experimental results in Section 4.4. Section 4.5 concludes the chapter.

4.2 System Overview

To evaluate the effects of noise perturbation, we use a fixed system for mask estimation and compare the quality of estimated masks as well as the resynthesized speech that are derived from the masked T-F representations of noisy speech. While comparison between an estimated mask and an ideal mask reveals the spectrotemporal distribution of estimation errors, resythesized speech can be directly compared to clean speech. In this study, we use the IRM as the target of supervised learning,

which is defined as follows. The IRM is defined below [82].

$$IRM(t,f) = (\frac{S(t,f)^2}{S(t,f)^2 + N(t,f)^2})^{\beta} \tag{4.1}$$

where β is a tunable parameter. A recent study has shown that $\beta = 0.5$ is a good choice for the IRM [119]. In this case, mask estimation becomes a regression problem where the target is the IRM. Ratio masking is shown to lead to slightly better objective intelligibility results than binary masking [119]. In this study, we use the IRM with $\beta = 0.5$ as the learning target. The IRM is computed from the 64-channel cochleagrams of premixed clean speech and noise. The cochleagram is a time-frequency representation of a signal [115]. We use a 20 ms window and a 10 ms window shift to compute cochleagram in this study. We perform IRM estimation using a DNN and a set of acoustic features. Recent studies have shown that DNN is a strong classifier for ASR [78] and speech separation [120, 127]. As shown in Fig. 4.1, acoustic features are extracted from a mixture sampled at 16 kHz, and then sent to a DNN for mask prediction.

We use classification accuracy, HIT−FA rate and STOI score [105] as three criteria for measuring the quality of the estimated IRM. Since the first two criteria are defined for binary masks, we calculate them by binarizing a ratio mask to a binary one. During the mask conversion, the LC is set to be 5 dB lower than the SNR of a given mixture. While classification accuracy and HIT−FA rate evaluate estimated masks, STOI compares the the short-time envelopes of clean speech and resynthesized speech obtained from IRM masking, and it is a standard objective metric of speech intelligibility [105].

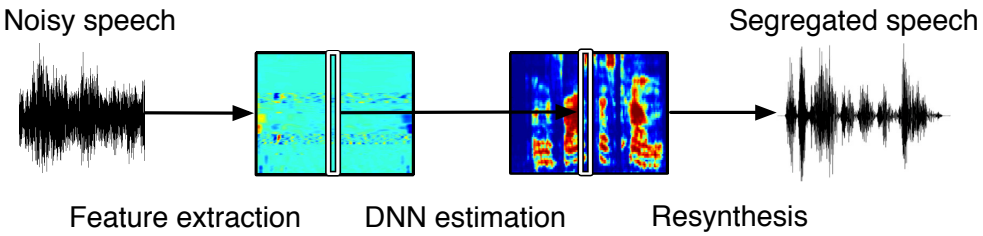


Figure 4.1: Diagram of the proposed system.

4.3 Noise Perturbation

The goal of noise perturbation is to expand noise segments to cover unseen scenarios so that the overfitting problem is mitigated in supervised speech separation. A reasonable and straightforward idea for noise expansion is to reverse each noise signal in time. We have evaluated this idea and unfortunately adding reversed noises does not improve speech separation results. We conjecture that the spectrogram of a noise segment may be a better domain to apply perturbation. A recent study has found that three perturbations on speech samples in the spectrogram domain improve ASR performance [60]. These perturbations were used to expand the speech samples so that more speech patterns are observed by a classifier. The three perturbations are introduced below. Unlike this study, we perturb noise samples instead of perturbing speech samples, as we are dealing with highly nonstationary noises.

4.3.1 Noise rate (NR) perturbation

Speech rate perturbation, a way of speeding up or slow down speech, is used to expand training utterances during the training of an ASR system. In our study, we extend the method to vary the rate of nonstationary noises. We increase or decrease noise rate by factor γ. When a noise rate is being perturbed, the value of γ is

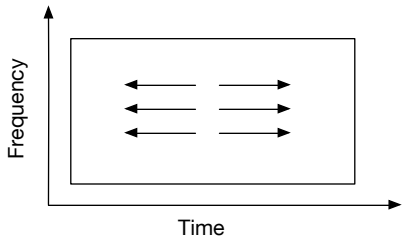


Figure 4.2: Illustration of noise rate perturbation.

randomly selected from an interval $[\gamma_{min}, 2-\gamma_{min}]$. The effect of NR perturbation on a spectrogram is shown in Fig. 4.2.

4.3.2 Vocal tract length (VTL) perturbation

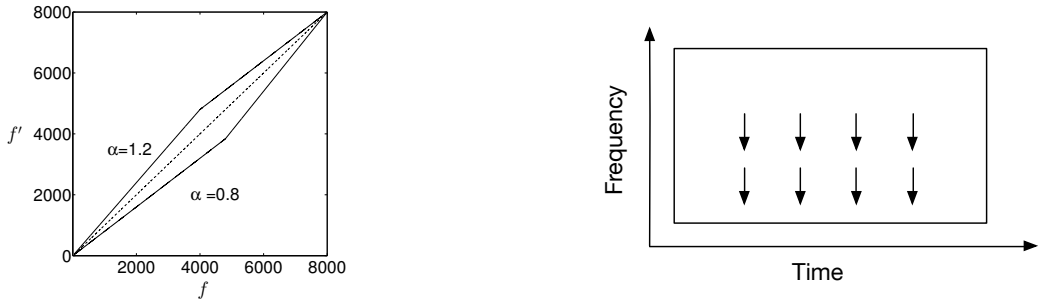
VTL perturbation has been used in ASR to cover the variation of vocal tract length among speakers. A recent study suggests that VTL perturbation improves ASR performance [54]. VTL perturbation essentially compresses or stretches the medium and low frequency components of an input signal. We use VTL perturbation as a method of perturbing a noise segment. Specifically, we follow the algorithm in [54] to perturb noise signals:

$$f' = \begin{cases} f\alpha, & \text{if } f \leq F_{hi}\frac{min(\alpha,1)}{\alpha} \\ \frac{S}{2} - \frac{\frac{S}{2}-F_{hi}min(\alpha,1)}{\frac{S}{2}-F_{hi}\frac{min(\alpha,1)}{\alpha}}(\frac{S}{2}-f), & \text{otherwise} \end{cases} \tag{4.2}$$

where α is the wrapping factor, S is the sampling rate, and F_{hi} controls the cutoff frequency. Fig. 4.3(a) shows how VTL perturbation compresses or stretches a portion of a spectrogram. The effect of VTL perturbation is visualized in Fig. 4.3(b).

4.3.3 Frequency perturbation

When frequency perturbation is applied, frequency bands of a spectrogram are randomly shifted upward or downward. We use the method described in [60] to



(a) (b)

Figure 4.3: (a) Mapping function for vocal tract length perturbation. The frequencies below a cutoff are stretched if $\alpha > 1$, and compressed if $\alpha < 1$. (b) Illustration of vocal tract length perturbation. The medium and low frequencies are compressed in this case.

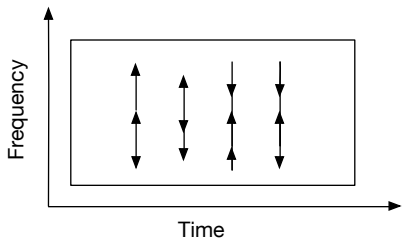


Figure 4.4: Illustration of frequency perturbation.

randomly perturb noise samples. Frequency perturbation takes three steps. First, we randomly assign a value to each T-F unit, which is drawn from a uniform distribution.

$$r(f,t) \sim U(-1,1) \tag{4.3}$$

Then we derive the perturbation factor $\delta(f,t)$ by averaging the assigned values of neighboring time-frequency units. This averaging step avoids large oscillations in spectrogram.

$$\delta(f,t) = \frac{\lambda}{(2p+1)(2q+1)} \sum_{f'=f-p}^{f+p} \sum_{t'=t-q}^{t+q} r(f',t') \tag{4.4}$$

where p and q control the smoothness of the perturbation, and λ controls the magnitude of the perturbation. These tunable parameters are decided experimentally. Finally the spectrogram is perturbed as follows.

$$\tilde{S}(f,t) = S(f + \delta(f,t), t) \tag{4.5}$$

where $S(f,t)$ represents the original spectrogram and $\tilde{S}(f,t)$ is the perturbed spectrogram. Interpolation between neighboring frequencies is used when $\delta(f,t)$ is not an integer. The effect of frequency perturbation is visualized in Fig. 4.4.

4.4 Experimental Results

4.4.1 Experimental setup

We use the IEEE corpus recorded by a male speaker [52] and six nonstationary noises from the DEMAND corpus [109] to create mixtures. All signals are sampled at 16 kHz. Note that all recordings of the DEMAND corpus are made with a 16-channel microphone array, we use only one channel of the recordings since this study is on monaural speech separation.

The DEMAND corpus has six categories of noises. We choose one noise from each category to represent distinct environments. The six nonstationary noises, each is five-minute long, are described as follows.

1. The "Street" category:

   The SCAFE noise, recorded in the terrace of a cafe at a public square.

2. The "Domestic" category:

   The DLIVING noise, recorded inside a living room.

3. The "Office" category:

   The OMEETING noise, recorded in a meeting room.

4. The "Public" category:

   The PCAFETER noise, recorded in a busy office cafeteria.

5. The "Nature" category:

   The NPARK noise, recorded in a well visited city park.

6. The "Transportation" category:

   The TMETRO noise, recorded in a subway.

To create a mixture, we mix one IEEE sentence and one noise type at -5 dB SNR. This low SNR is selected with the goal of improving speech intelligibility in mind where there is not much to improve at higher SNRs [39]. The training set uses 600 IEEE sentences and randomly selected segments from the first two minutes of a noise, while the test set uses another 120 IEEE sentences and randomly selected segments from the second two minutes of a noises. Therefore, the test set has different sentences and different noise segments from the training set. We create 50 mixtures

for each training sentence by mixing it with 50 randomly selected segments from a given noise, which results in a training set containing 600×50 mixtures. The test set includes 120 mixtures. We train and test using the same noise type and SNR condition.

To perturb a noise segment, we first apply short-time Fourier transform (STFT) to derive noise spectrogram, where a frame length of 20 ms and a frame shift of 10 ms are used. Then we perturb the spectrogram and derive a new noise segment. To evaluate the three noise perturbations, we create five different training sets, each consists of 600×50 mixtures. We train a mask estimator for each training set and evaluate on a fixed test set (i.e. the 120 mixtures created from the original noises). The five training sets are described as follows.

1. Original Noise: All mixtures are created using original noises.
2. NR Perturbation: Half of the mixtures are created from NR perturbed noises, and the other half are from original noises.
3. VTL Perturbation: Half of the mixtures are created from VTL perturbed noises, and the other half are from original noises.
4. Frequency Perturbation: Half of the mixtures are created from frequency perturbed noises, and the other half are from original noises.
5. Combined: Half of the mixtures are created from applying three perturbations altogether, and the other half are from original noises.

The acoustic features we extract from mixtures are a complementary feature set (AMS + RASTAPLP + MFCC) [118] combined with gammatone filterbank (GFB)

features. To compute 15-D AMS, we derive 15 modulation spectrum amplitudes from the decimated envelope of an input signal [64]. 13-D RASTAPLP is derived by applying linear prediction analysis on the RASTA-filtered bark-scale power spectrum of an input signal [42]. We follow a standard procedure to compute 31-D MFCC. To derive GFB features, an input signal is passed to a 64-channel gammatone filterbank, the response signals are decimated to 100 Hz to form 64-D GFB features. After appending delta features, we end up with a feature set of 123×2 dimensions.

A four-hidden-layer DNN is employed to learn the mapping from acoustic features to the IRM. Each hidden layer of the DNN has 1024 rectified linear units [81]. To incorporate temporal context and obtain smooth mask estimation, we use 5 frames of features to estimate 5 frames of the IRM [119]. As we use a 246-D feature set and the 64-channel IRM, the input layer of the DNN has 246×5 units and the output layer has 64×5 sigmoidal units. Since each frame of the mask is estimated 5 times, we take the average of the 5 estimates. We use mean squared error as the cost function. Hidden-layer dropout [22] and adaptive stochastic gradient descent (AdaGrad) [25] with a mini-batch size of 1024 are used to train the DNN. We set the dropout ratio to 0.2 and the initial learning rate of AdaGrad to 0.003. We train the DNN for 80 epochs and select the best epoch by cross validation.

4.4.2 Parameters of noise perturbation

In this section, three sets of experiments are carried out to explore the parameters used in the three perturbations to get the best performance. To facilitate parameter selection, we create five smaller training sets, following the same configuration in Section 4.4.1 except that we use 480 IEEE clean sentences to create 480×20 training

mixtures. Another 120 IEEE sentences (different than the test ones in Section 4.4.1) are used to create 120 test mixtures only for the purpose of choosing parameter values (i.e. a development set). The speech separation performance is evaluated in term of STOI score.

In NR perturbation, the only adjustable parameter is the rate γ. We can slow down a noise by setting $\gamma < 1$, or speed it up using $\gamma > 1$. To capture various noise rates, we randomly draw γ from an interval $[\gamma_{min}, 2 - \gamma_{min}]$. We evaluate various intervals in term of speech separation performance. As shown in Fig. 4.5, the interval $[0.1, 1.9]$ (i.e. $\gamma_{min} = 0.1$) gives the best performance for six noises.

In VTL perturbation, there are two parameters: F_{hi} controls cutoff frequency and α the warping factor. F_{hi} is set to 4800 to roughly cover the frequency range of speech formants. We randomly draw α from an interval $[\alpha_{min}, 2 - \alpha_{min}]$ to systematically stretch or shrink the frequencies below the cutoff frequency. Fig. 4.6 shows the effects of different intervals on speech separation performance. The interval of $[0.3, 1.7]$ (i.e. $\alpha_{min} = 0.3$) leads to the best result for the majority of the noise types.

In frequency perturbation, a 161-band spectrogram derived from a noise segment is perturbed using the algorithm described in Section 4.3.3. We set $p = 50$ and $q = 100$ to avoid dramatic perturbation along time and frequency axes. We experiment with different perturbation intensity λ. As shown in Fig. 4.7, $\lambda = 1000$ achieves the best performance for the majority of the noise types.

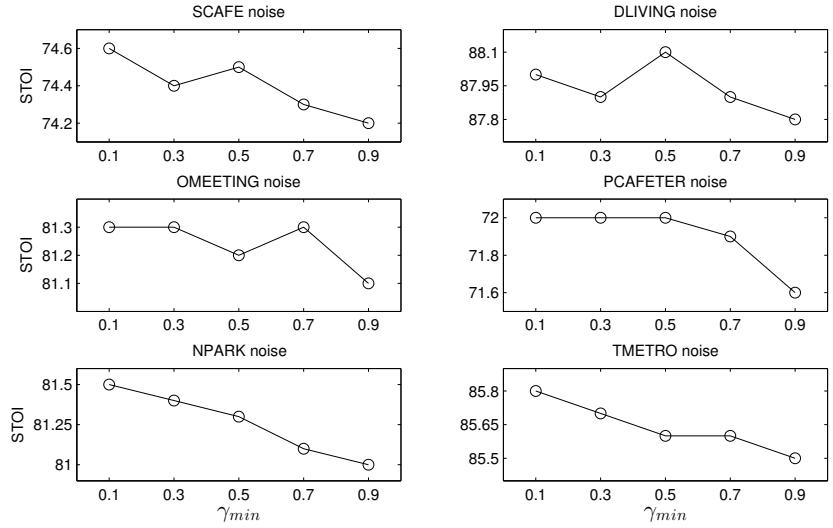


Figure 4.5: The effect of the minimum noise rate γ_{min} for NR perturbation.

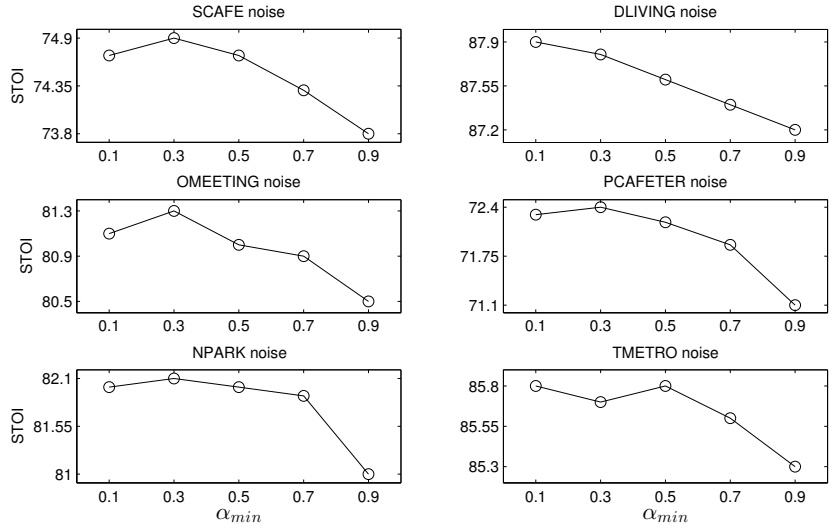


Figure 4.6: The effect of the minimum wrapping factor α_{min} for VTL perturbation.

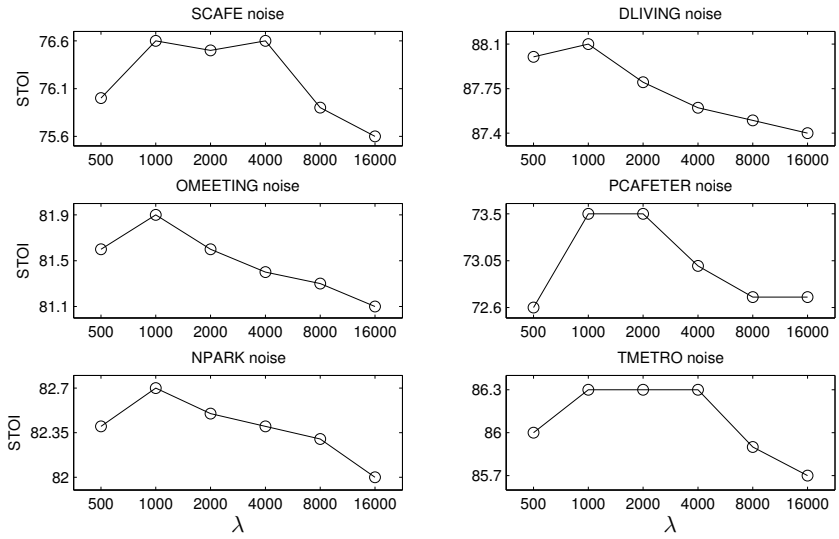


Figure 4.7: The effect of the perturbation intensity λ for frequency perturbation.

4.4.3 Evaluation results and comparisons

Before we evaluate the three perturbations, it is worth stressing that we are trying to apply noise perturbations to improve the performance of a strong baseline separation system, making further improvements harder. As described in Section 4.2, this baseline system trains a DNN to estimate the IRM. To demonstrate this, we compare our baseline system with a recently proposed supervised algorithm based on non-negative matrix factorization (NMF) [85, 79]. This algorithm is called active-set Newton algorithm (ASNA), which we denote as ASNA-NMF [112]. We select ASNA-NMF as it outperforms many variants of supervised NMF algorithms [112]. We set ASNA-NMF to use 1000 speech bases, 300 noise bases and 5 frames of magnitude spectra. For a fair comparison, we train ASNA-NMF on the first two minutes of a noise and 600 IEEE sentences, and test on the second two minutes of the noise and another 120 IEEE sentences. Table 4.1 shows the separation results of the baseline

Table 4.1: Comparison of DNN-based ratio masking (the baseline) with ASNA-NMF in terms of STOI (in %) for six noises at −5 dB.

Method \ Noise	SCAFE	DLIVING	OMEETING	PCAFETER	NPARK	TMETRO	Average
Unprocessed	64.1	79.3	67.8	62.5	67.7	77.5	69.8
ASNA-NMF	67.5	82.4	73.4	66.0	72.5	81.2	73.8
DNN-IRM	73.7	87.5	80.0	71.4	80.2	85.9	79.8

Table 4.2: Classification accuracy (in %) for six noises at −5 dB

Perturbation \ Noise	SCAFE	DLIVING	OMEETING	PCAFETER	NPARK	TMETRO	Average
Original Noise	73.0	84.0	80.0	70.3	82.7	80.3	78.4
NR Perturbation	80.2	88.5	85.3	77.9	88.5	85.1	84.2
VTL Perturbation	80.1	87.7	84.9	77.8	89.2	85.5	84.2
Frequency Perturbation	84.4	88.6	86.7	80.6	90.0	86.7	86.2
Combined	81.8	88.0	86.1	78.9	89.6	86.6	85.2

Table 4.3: HIT−FA rate (in %) for six noises at −5 dB, where FA is shown in parentheses.

Perturbation \ Noise	SCAFE	DLIVING	OMEETING	PCAFETER	NPARK	TMETRO	Average
Original Noise	55 (37)	70 (23)	65 (28)	50 (40)	69 (22)	63 (32)	62 (30)
NR Perturbation	64 (24)	77 (15)	72 (18)	60 (26)	77 (12)	72 (21)	70 (19)
VTL Perturbation	64 (24)	76 (16)	71 (19)	60 (27)	78 (10)	72 (21)	70 (20)
Frequency Perturbation	69 (17)	77 (14)	74 (15)	63 (21)	79 (9)	74 (18)	73 (16)
Combined	67 (21)	77 (15)	73 (16)	61 (25)	78 (10)	74 (18)	72 (18)

system and ASNA-NMF in terms of STOI. The DNN-based baseline produces significantly better results than ASNA-MNF for six noises at −5 dB SNR. On average, DNN-based ratio masking improves STOI by 10%, while ASNA-NMF improves STOI by 4%.

We evaluate the three perturbations with the parameter values selected in Section 4.4.2 and the five large training sets described in Section 4.4.1. The effects of noise

Table 4.4: STOI (in %) of separated speech for six noises at −5 dB, where STOI of unprocessed mixtures is shown in parentheses.

Perturbation \ Noise	SCAFE	DLIVING	OMEETING	PCAFETER	NPARK	TMETRO	Average
Original Noise	73.7 (64.1)	87.5 (79.3)	80.0 (67.8)	71.4 (62.5)	80.2 (67.7)	85.9 (77.5)	79.8 (69.8)
NR Perturbation	76.5 (64.1)	89.2 (79.3)	82.5 (67.8)	74.1 (62.5)	83.2 (67.7)	87.4 (77.5)	82.1 (69.8)
VTL Perturbation	76.1 (64.1)	88.7 (79.3)	82.2 (67.8)	74.0 (62.5)	83.6 (67.7)	87.2 (77.5)	82.0 (69.8)
Frequency Perturbation	78.2 (64.1)	89.1 (79.3)	83.3 (67.8)	75.1 (62.5)	84.1 (67.7)	87.8 (77.5)	82.9 (69.8)
Combined	77.0 (64.1)	88.6 (79.3)	82.7 (67.8)	74.7 (62.5)	83.8 (67.7)	87.6 (77.5)	82.4 (69.8)

Table 4.5: HIT−FA rate (in %) during voiced intervals, where FA is shown in parentheses.

Perturbation \ Noise	SCAFE	DLIVING	OMEETING	PCAFETER	NPARK	TMETRO	Average
Original Noise	50 (44)	70 (26)	62 (33)	48 (45)	71 (24)	55 (42)	59 (36)
NR Perturbation	60 (32)	75 (21)	69 (24)	57 (33)	79 (15)	63 (33)	67 (26)
VTL Perturbation	62 (30)	75 (21)	70 (24)	60 (31)	80 (13)	65 (31)	69 (25)
Frequency Perturbation	66 (24)	76 (20)	72 (21)	62 (27)	80 (13)	67 (29)	70 (22)
Combined	65 (27)	76 (20)	72 (21)	61 (30)	80 (13)	68 (28)	70 (23)

Table 4.6: HIT−FA rate (in %) during unvoiced intervals, where FA is shown in parentheses.

Perturbation \ Noise	SCAFE	DLIVING	OMEETING	PCAFETER	NPARK	TMETRO	Average
Original Noise	48 (33)	61 (22)	59 (25)	41 (36)	57 (20)	61 (27)	54 (27)
NR Perturbation	54 (20)	70 (11)	64 (15)	48 (22)	62 (9)	68 (16)	61 (16)
VTL Perturbation	52 (21)	68 (13)	64 (15)	45 (24)	62 (8)	68 (16)	60 (16)
Frequency Perturbation	59 (12)	68 (11)	66 (11)	48 (18)	62 (6)	70 (13)	62 (12)
Combined	55 (18)	68 (12)	64 (13)	46 (22)	62 (8)	69 (14)	61 (14)

perturbations on speech separation are shown in Table 4.2, Table 4.3 and Table 4.4, in terms of classification accuracy, HIT−FA rate and STOI score respectively. The results indicate that all three perturbations lead to better speech separation than the baseline where only the original noises are used. Frequency perturbation performs better than the other two perturbations. Compared to only using the original noises, the frequency perturbed training set on average increases classification accuracy, HIT−FA rate and STOI score by 8%, 11% and 3%, respectively. This indicates that noise perturbation is an effective technique for improving speech separation results. Combining three perturbations, however, does not lead to further improvement over frequency perturbation. We conjecture that frequency perturbation alone provides sufficient noise variations for generalization purposes. To verify this, we expand training by mixing each clean sentence with more noise segments. For the training sets using perturbed noises, we fix the number of mixtures created from original noises to 600×25, but vary the number of mixtures created from perturbed noises. Fig. 4.8 shows the average STOI results as the number is set to 600×25, 600×50, and 600×150. As the size of the training set increases, the combined method and frequency perturbation reach almost the same peak performance. We also observe that the speech separation performance does not benefit from a larger training set when no perturbation is used.

A closer look at Table 4.3 reveals that the contribution of frequency perturbation lies mainly in the large reduction in FA rate. This means that the problem of misclassifying noise-dominant T-F units as speech-dominant is mitigated. This effect can be illustrated by visualizing the masks estimated from the different training sets and the ground truth mask in Fig. 4.9 (e.g. around frame 150). When the mask estimator is

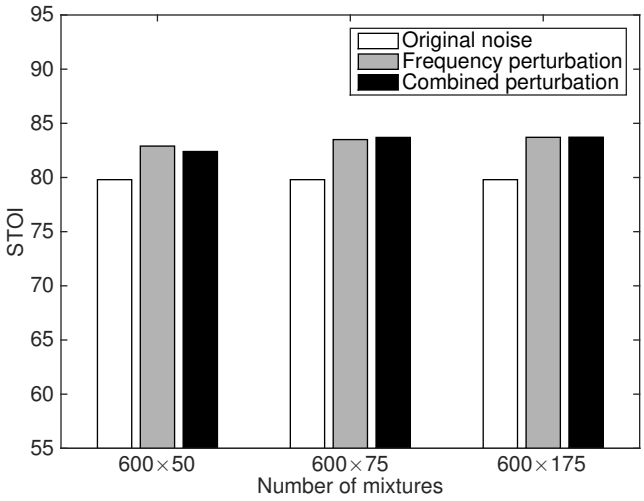


Figure 4.8: Average STOI (in %) of separated speech for six noises at −5 dB with respect to the number of training mixtures.

trained with the original noises, it mistakenly retains the regions where target speech is not present, which can be seen by comparing the top and bottom plots of Fig. 4.9. Applying frequency perturbation to noises essentially exposes the mask estimator to more noise patterns and results in a more accurate mask estimator, which is shown in the middle plot of Fig. 4.9.

In addition, we show HIT−FA rate for voiced and unvoiced intervals in Table 4.5 and Table 4.6 respectively. We find that frequency perturbation is effective for both voiced and unvoiced intervals.

While classification accuracy and HIT−FA rate evaluate the estimated binary masks, STOI directly compares clean speech and the resynthesized speech. As shown in Table 4.4, frequency perturbation yields higher average STOI scores than using original noises with no perturbation and NR and VTL perturbations.

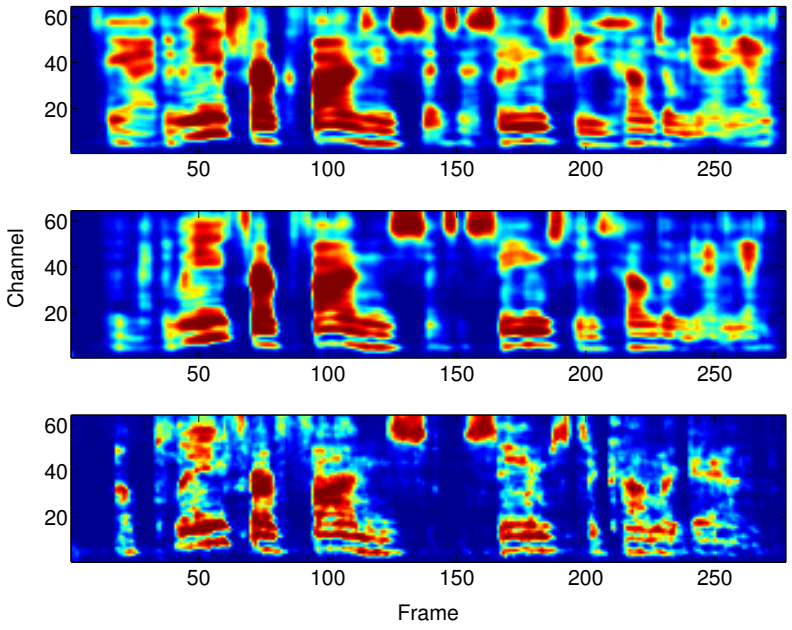


Figure 4.9: Mask comparisons. The top shows a ratio mask obtained from training on original noises, the middle shows a mask obtained from training on frequency perturbed noise, and the bottom shows the IRM.

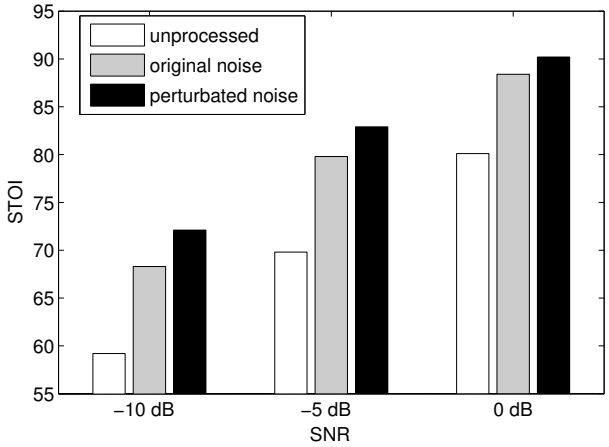


Figure 4.10: The effect of frequency perturbation in three SNR conditions. The average STOI scores (in %) across six noises are shown for unprocessed speech, separated speech by training on original noises, and separated speech by training on frequency perturbed noises.

Table 4.7: STOI (in %) of separated speech for five unmatched noises at −5 dB, where STOI of unprocessed mixtures is shown in parentheses.

Training Noise \ Test Noise	DLIVING	OMEETING	PCAFETER	NPARK	TMETRO
SCAFE	86.2 (79.3)	73.2 (67.8)	74.0 (62.5)	80.0 (67.7)	86.6 (77.5)
Matched noise	89.1 (79.3)	83.3 (67.8)	75.1 (62.5)	84.1 (67.7)	87.8 (77.5)

To evaluate the effectiveness of frequency perturbation at other SNRs, we carry out additional experiments at −10 dB and 0 dB input SNRs, where we use the same parameter values as for −5 dB SNR. Fig. 4.10 shows frequency perturbation improves speech separation in terms of STOI in each SNR condition. Also, we find that frequency perturbation remains the most effective among the three perturbations at −10 dB and 0 dB SNR.

All the above evaluations are conducted on unseen segments of the same non-stationary noises, corresponding to environment-specific speech separation [49]. Although not the focus of the present study, it is interesting to see how our mask estimator performs when evaluated on completely new noises. To get an idea, we evaluate the model trained using frequency perturbation. We use the same setting described in Section 4.4.1 except that we train on SCAFE noise and test on the other five noises. The results are shown in Table 4.7. As expected, the model does not perform as well as in the matched noise case. But it still significantly improves STOI over unprocessed mixtures.

4.5 Concluding Remarks

In this study, we have explored the effects of noise perturbation on supervised monaural speech separation at low SNR levels. As a training set is usually created

from limited speech and noise resources, a classifier likely overfits the training set and makes poor predictions on a test set, especially when background noise is highly nonstationary. We suggest to expand limited noise resources by noise perturbation.

We have evaluated three noise perturbations with six nonstationary noises recorded from daily life for speech separation. The three are noise rate, VTL, and frequency perturbations. When a DNN is trained on a data set which utilizes perturbed noises, the quality of the estimated ratio mask is improved as the classifier has been exposed to more scenarios of noise interference. In contrast, a mask estimator learned from a training set that only uses original noises tends to make more false-alarm errors (i.e. higher FA rate), which is detrimental to speech intelligibility [128]. The experimental results show that frequency perturbation, which randomly perturbs the noise spectrogram along frequency, almost uniformly gives the best speech separation results among the three perturbations examined in this study in terms of classification accuracy, HIT$-$FA rate and STOI score.

Finally, this study adds another technique to deal with the generalization problem in supervised speech separation. Previous studies use model adaptation [34] and extensive training [120] to deal with the mismatch of SNR conditions, noises and speakers between training and testing. Our study aims at situations with limited training noises, and provides an effective data augmentation method that improves generalization in nonstationary environments. The idea of signal perturbation may also be applicable to augmenting speech signals for improved generalization to different kinds of speech data, such as different speaking rates and styles.

CHAPTER 5

LARGE-SCALE TRAINING FOR NOISE TYPE GENERALIZATION AND SNR GENERALIZATION

This chapter describes noise type generalization and SNR generalization for supervised speech separation. The work presented in this chapter has been published in the *Journal of the Acoustical Society of America* [20] (see also [117]).

5.1 Introduction

A primary manifest of hearing loss, which affects roughly 10% of the population, is reduced speech intelligibility in background noises, particularly nonstationary noises [80] [24]. Compressive amplification implemented in modern hearing aids offers little help as both speech and noise are amplified. The lack of speech intelligibility improvement in noise is a main barrier to hearing aid adoption [1]. As a result, noise reduction is considered one of the biggest challenges in hearing aid design. Extensive effort has been made in speech and signal processing over the past several decades to improve speech intelligibility in background noise for hearing-impaired (HI) listeners. A main approach involves speech enhancement, which is a class of monaural speech

segregation algorithms including spectral subtraction and mean-square error estimation [72]. Speech enhancement algorithms are capable of improving SNR and speech quality, but they fail to deliver speech intelligibility benefit [74] [72].

Recently, supervised speech segregation has received increasing attention. In its simplest form, supervised segregation estimates an ideal T-F mask of a noisy mixture using a trained classifier, typically a DNN. Unlike traditional speech enhancement, supervised segregation does not make explicit statistical assumptions about the underlying speech or noise signal, but rather learns data distributions from a training set. DNN-based IBM (see Section 3.1) and IRM (see Section 4.2) estimators have been demonstrated to improve intelligibility of noisy speech by HI listeners [39] [38]. A critical issue associated with this work involves the ability to generalize to unseen noisy conditions — those not employed during training. In the context of supervised speech segregation, generalization to unseen noisy environments is key. In [64], a Gaussian mixture model based IBM classifier was trained and tested on the same brief noise segments, with very limited generalizability [76]. Healy et al. [39] used random cuts from longer-duration noise segments for training and testing in order to reduce dependency on the specific characteristics of the training conditions. However, both training and test segments were drawn from the same overall noise segments, and generalizability was still limited.

A more recent study [38] took this issue a step further by dividing 10-minute nonstationary noises into two non-overlapping time portions, with the first part used for training and the second part for testing. Using different portions of a noise for training and testing is considered an important requirement for evaluating supervised segregation algorithms [76]. With relatively long noise segments for training and a

noise perturbation technique [19] to further expand the set of training noise samples, this DNN-based IRM estimator improved speech intelligibility for HI listeners in novel noise segments. However, the mask-estimation algorithm was trained and tested using the same noise type. In addition, the SNR was the same for both training and testing, which necessitated training to be repeated at each SNR tested.

The aim of the current study was to develop and test a speech segregation algorithm that can generalize to completely new noises, as well as to untrained SNRs. As the performance of supervised learning is predicated upon the information contained in a training set, the approach employed here for broad generalization was to enlarge the training set by including various acoustic conditions (see [120]). This conceptually simple approach, often referred to as multi-condition training, is widely used in ASR and robust ASR. In the current study, large-scale multi-condition training was employed for DNN-based IRM estimation. The training set included 10,000 noises, which exposed the IRM estimator to a large variety of noisy environments. The trained DNN was then used to segregate speech from two noises not included in those used for training: multi-talker babble and cafeteria noise. Further, training was completed at a single SNR, whereas testing was completed at multiple SNRs. Finally, the performance of the algorithm was evaluated using HI and normal-hearing (NH) listeners.

5.2 Method

5.2.1 Stimuli

The stimuli included IEEE sentences [52]. They were spoken by one male talker and digitized at 44.1 kHz with 16-bit resolution. Each sentence in this corpus contained five scoring keywords. The background noises also employed by Healy et al. [38] were employed here to test algorithm performance. These included 20-talker babble (both male and female voices) and cafeteria noise, both from an Auditec CD (St. Louis, MO, www.auditec.com). The cafeteria noise consisted of three overdubbed recordings made in a hospital employee cafeteria. SNRs employed to test algorithm performance were selected to obtain scores for unprocessed sentences in noise below and above 50%. These were 0 and 5 dB for the HI subjects and -2 and -5 dB for the NH subjects. Stimuli were downsampled to 16 kHz prior to processing.

Of the total of 720 IEEE sentences, 160 were arbitrarily selected to test algorithm performance. The remaining 560 IEEE sentences were employed for algorithm training, as described in Section 5.2.2. Thus, as in previous works [39] [38], sentences employed for algorithm testing were not employed for training. Test stimuli were created by mixing each test sentence with a segment of noise randomly selected from the final two minutes of the babble or cafeteria noise recording. This method follows that of Healy et al. [38], hence facilitating detailed comparison. An unprocessed speech-in-noise condition consisted of test sentences mixed with randomly selected segments of babble or cafeteria noise at the appropriate SNR. The algorithm-processed condition employed these same test sentences, each mixed with the same randomly selected noise segment used for the unprocessed condition. Thus, the only difference between the unprocessed and segregated conditions was algorithm processing.

5.2.2 Algorithm description

In this study, we train a DNN-based IRM estimator for supervised speech segregation. The IRM is computed from the cochleagram [115] of the premixed speech and noise. The cochleagram has 64 frequency channels centered from 50 to 8000 Hz and equally spaced on the equivalent rectangular bandwidth scale. IRM estimation starts with extraction of acoustic features from noisy speech. The DNN is then trained using these features from each speech-plus-noise mixture, along with the IRM for that mixture. After training, the DNN is used to estimate the IRM when provided only the speech-plus-noise mixture, which is then applied to the noisy speech to resynthesize a segregated speech signal. It was chosen to estimate the IRM instead of the IBM because ratio masking leads to better speech quality without compromising intelligibility [119] [38].

Specifically, the IRM was computed with a 20-ms frame length and 10-ms frame shift. The power (1/15) compressed cochleagram of noisy speech was used as the only acoustic feature for IRM estimation. To incorporate temporal context, 23 frames of acoustic features were concatenated as the input to a 5-hidden-layer DNN, which simultaneously predicted 5 frames of the IRM. Since each frame of the IRM was predicted 5 times, the average was taken as the final estimate. Predicting multiple frames of training targets in this way encodes a measure of ensemble learning and yields a consistent improvement in speech segregation performance [119]. The DNN had 23×64 units in the input layer, 2048 rectified linear units [81] in each of the five hidden layers, and 5×64 sigmoidal units in the output layer. Dropout with a ratio of 0.2 was used for all hidden layers. Stochastic gradient descent with a mini-batch size of 256 and mean square error loss function was employed to train the DNN.

As discussed in Section 5.1, the approach employed currently for better generalization was to perform large-scale training to expose the DNN to a broad variety of noisy conditions. A large training set was created by mixing the 560 IEEE sentences with 10,000 non-speech sounds from a sound-effect library (Richmond Hill, Ontario, Canada, www.sound-ideas.com). The total duration of the noises was approximately 125 hours. The training set consisted of 640,000 mixtures, each of which was created by mixing a randomly selected IEEE sentence with a random segment of a randomly selected noise at the fixed SNR of -2 dB. Both random selections (sentence and noise) were done with replacement. The total duration of the training mixtures was approximately 380 hours. It is worth emphasizing that the 160 IEEE sentences and the two noises used to create test stimuli (described in Section 5.2.1) for speech intelligibility evaluation were not employed (seen) during training. To facilitate discussion, the model trained with 10,000 noises is called the 10K-noise model. In order to demonstrate the effect of the number of noises on generalization, a 100-noise model was trained using the same settings described above except that 100, rather than 10,000, nonspeech environmental sounds (Columbus, OH, www.cse.ohiostate.edu/pnl/corpus/HuCorpus.html) were used, as in [120]. Again, 640,000 mixtures were prepared using the 560 training sentences randomly paired with these 100 noises, so that total duration of the training set was the same as that for the 10K-noise model.

To put the performance of the noise-independent models (i.e., 10K-noise and 100-noise models) in perspective, the same DNN-based IRM estimator was trained and tested on the same noise type, denoted as the noise-dependent model. This model was trained on one time portion of a noise and tested on another portion of the same

noise, with no overlap between noise segments used for training and those used for testing. Specifically, the two Auditec noises (20-talker babble and cafeteria noise) were each 10 minutes long, and the noise-dependent model was trained on the first 8 minutes of each noise and tested on the remaining 2 minutes of the same noise. In addition to these Auditec noises, two other noises from the NOISEX corpus [110] were used for evaluating the noise-dependent model. These noises were factory noise and 100-talker babble noise (denoted as babble2). The NOISEX noises are each 4 minutes long, and the noise-dependent model was trained on the first 2 minutes of each noise and tested on the remaining 2 minutes of the same noise. As for the other models tested currently, the 560 IEEE training sentences and an SNR of -2 dB were employed. For each of the four noises, the training set for the noise-dependent model consisted of 560×50 mixtures, with half of the noise samples created using frequency perturbation [19] [38].

5.2.3 Subjects

A first group of subjects consisted of 10 bilateral hearing-aid wearers having a sensorineural hearing loss. These HI listeners were representative of typical audiology patients seen at The Ohio State University Speech-Language-Hearing Clinic. Ages ranged from 24 to 73 yrs (mean = 54.8), and seven were female. Hearing status was evaluated on day of test (or within one week prior to test, for 2 of 10 subjects) through otoscopy, tympanometry [3] and pure-tone audiometry [4] [5]. Pure-tone averages (PTAs, average of audiometric thresholds at 500, 1000 and 2000 Hz) ranged from 33 to 69 dB HL (average 42.2). Hearing losses therefore ranged from mild to severe and were moderate on average. Audiograms are presented in Fig. 5.1, where

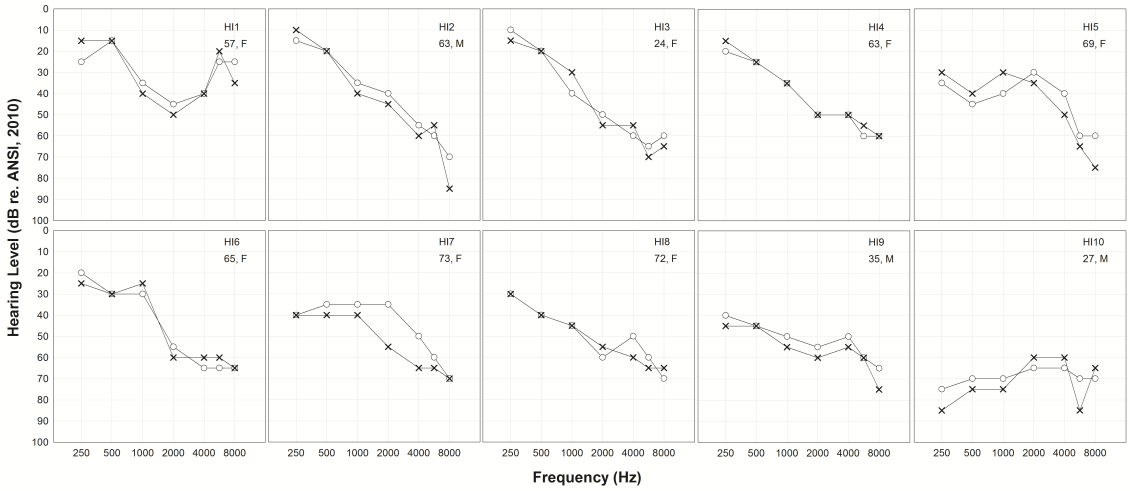


Figure 5.1: Pure-tone air-conduction audiometric thresholds for the listeners with sensorineural hearing impairment. Right ears are represented by circles and left ears are represented by Xs. Also displayed are subject number, listener age in years, and gender.

subjects are numbered and plotted in order of increasing PTA. Also provided are subject numbers, ages and genders.

A second group of subjects was composed of 10 listeners (9 female) having NH, as defined by audiometric thresholds on day of test at or below 20 dB HL at octave frequencies from 250 to 8000 Hz [4] [5]. They were recruited from undergraduate courses at The Ohio State University and had ages ranging from 19 to 41 yrs (mean = 22.9). All subjects received a monetary incentive or course credit for participating. As in our previous work on this topic [39] [38], age matching between HI and NH subjects was not performed because the goal was to assess the abilities of typical (often older) HI listeners relative to the ideal performance of young NH listeners. However, it is noteworthy that the HI and NH age groups ranged considerably and overlapped. Further, the mean age of the HI listeners tested currently was only 55 yrs.

5.2.4 Procedure

Each subject heard 20 sentences in each of eight conditions (2 noise types $\times$ 2 SNRs $\times$ 2 processing conditions). Care was taken to ensure that no subjects had prior exposure to the sentence materials and no sentence was repeated in any condition for any listener. Noise type and SNR were blocked so that unprocessed and algorithm conditions appeared juxtaposed in presentation order for each noise type and SNR. The order of conditions was balanced such that half the listeners heard unprocessed prior to algorithm for each noise type and SNR (and the other half heard the opposite order), and half of the subjects heard the babble conditions followed by the cafeteria-noise conditions (and the other half heard the opposite order). Sentence list-to-condition correspondence was pseudo-randomized for each subject.

The total RMS level of each stimulus in each condition was set to 65 dBA for NH listeners and 65 dBA plus frequency-specific gains as prescribed by the NAL-R hearing-aid fitting formula [12] for each individual HI listener. The fitting procedure employed in Healy et al. [38] was employed, including the use of a RANE (Mukilteo, WA) DEQ 60L digital equalizer to provide frequency-specific gains. Echo Digital Audio (Santa Barbara, CA) Gina 3G digital-to-analog converters were employed, as was a Mackie (Woodinville, WA) 1202-VLZ mixer to adjust overall gain, and Sennheiser (Wedemark, Germany) HD 280 Pro headphones for diotic presentation. Calibration was performed using a Larson Davis sound-level meter and flat-plate headphone coupler (models 824 and AEC 101; Depew, NY). As subject-specific hearing-aid gains were provided by the experimental apparatus, HI listeners were tested with hearing aids removed.

Familiarization at the start of testing involved five IEEE sentences not employed for formal testing, first in quiet, followed by five sentences in the unprocessed noisy condition, then five in the algorithm condition. Babble or cafeteria noise was used, corresponding to whichever noise the subject was to receive first, and the SNR matched the least favorable employed during testing. This familiarization was repeated half way through the experiment using the other noise type, prior to switching noise types. The HI subjects were asked after presentation of the initial sentences if the stimuli were comfortable in level. The overall presentation level was reduced by 5 dB for the one subject who indicated that the stimuli sounded loud. This individual judged this reduced level to be comfortable. The overall presentation level was 96 dBA or below for all subjects. The experimenter was seated with the subject in a double-walled audiometric booth, and instructed the listeners to repeat back as much of each sentence as possible, controlled the presentation of each sentence, and scored responses.

5.3 Results and Discussion

5.3.1 Predicted intelligibility results

Before presenting intelligibility results from HI and NH listeners, predicted intelligibility scores using an acoustic metric are provided. Specifically, the STOI metric [105] was employed, as it is a standard speech intelligibility predictor involving a comparison between the envelopes of segregated speech and clean speech. STOI evaluation provides an opportunity to compare predicted and actual intelligibility scores and an objective benchmark for future algorithm comparisons.

Table 5.1: Speech segregation results, for four test noises and their average, at −2 dB SNR measured in short-time objective intelligibility (STOI) values.

	Babble	Cafeteria	Factory	Babble2	Average
Unprocessed	0.612	0.596	0.611	0.611	0.608
100-noise model	0.683	0.704	0.750	0.688	0.706
10K-noise model	0.792	0.783	0.807	0.786	0.792
Noise-dependent model	0.833	0.770	0.802	0.762	0.792

Table 5.1 shows the STOI results for the unprocessed mixtures, the two noise-independent models, and the noise-dependent model. The mean STOI scores were computed for the 160 test sentences in each test-noise condition. Values are shown for each of the test noises, and for the average across noises. Apparent is that all models improved STOI scores relative to unprocessed speech in noise. The noise-independent model trained with 100 noises performed substantially poorer than that trained with 10,000 noises, even though the two were trained using the same number of mixtures (640,000). Therefore, it is the increase in the amount of distinct noise samples rather than the size of the training set that determines generalization ability. On the other hand, the 10K-noise model provided identical performance on average to the noise-dependent model. This indicates that, with 10,000 noises, the noise-independent model has been exposed to an adequate variety of noisy environments. It is highly encouraging that the STOI scores for the noise-independent model match those for the noise-dependent model (see Wang et al. [117], for additional STOI results).

Figure 5.2 visualizes the first 100 learned filters taken from the first hidden layer of the 10K-noise model. Each panel in Fig. 5.2 corresponds to a hidden unit, showing the weights coming from the input layer in two dimensions: the abscissa represents time (23 frames) and the ordinate represents frequency (64 channels). Apparent is that

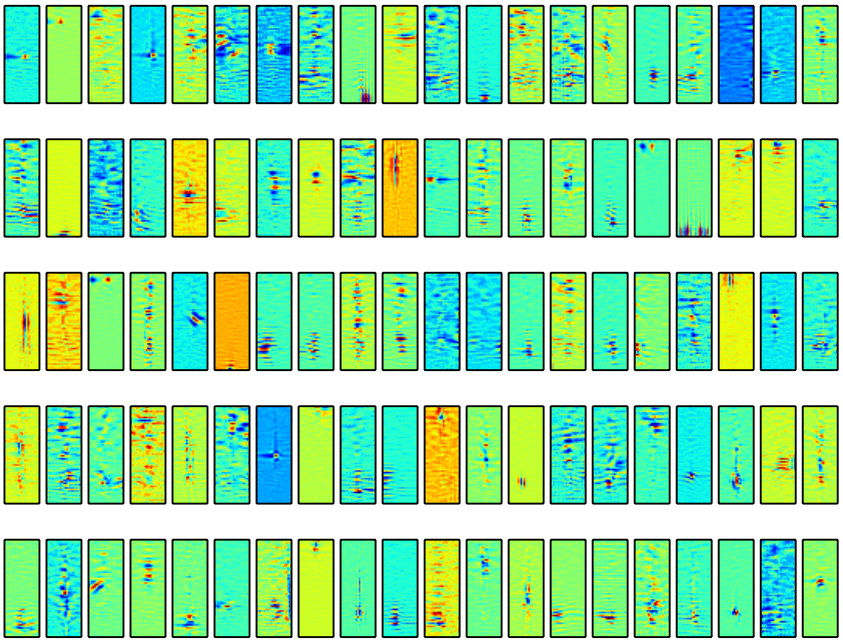


Figure 5.2: Visualization of 100 filters learned by the bottom hidden layer of a DNN trained on mixtures created using 10,000 noises. Each filter is shown in two dimensions: the abscissa represents time (23 frames) and the ordinate represents frequency (64 channels).

the network learns what appear to be speech-specific feature detectors. For example, some filters resemble harmonic detectors (e.g., the 10th filter in the last row), while some others seem to capture feature transitions (e.g., the 5th filter in the third row). These speech-specific feature detectors appear to encode fundamental characteristics of the speech signal, enabling the model to be noise independent. Although the 10K-noise model was trained on 640,000 mixtures created at -2 dB SNR, it is able to generalize to different SNRs. To demonstrate this, a second 10K-noise model was trained on 640,000 new random mixtures created at -5 dB, and both models were evaluated on both the -5 dB and -2 dB test sets. Cafeteria noise was employed. As shown in Fig. 5.3, the STOI difference between the matched and mismatched SNR conditions is negligible at both test SNR levels. This is likely because the model had seen sufficient local (i.e., frame level) SNR variations even with a fixed utterance-level SNR in training. Therefore, the 10K-noise model trained at -2 dB was used to produce the algorithm-processed stimuli for all SNR conditions employed for human-subject testing.

Figure 5.4 illustrates the results of using the 10K-noise model trained at -2 dB to perform speech segregation on a mixture of an IEEE sentence and cafeteria noise at 0 dB SNR. The cochleagrams of clean speech, speech-plus-noise, and segregated speech are shown in Fig. 5.4(a), Fig. 5.4(b) and Fig. 5.4(e), respectively. The IRM is given in Fig. 5.4(c) and the estimated IRM in Fig. 5.4(d). It is clear that the target speech is well separated from the cafeteria noise despite that the test noise and test SNR were not used during the training stage.

Table 5.2 lists the STOI scores for the same test conditions used in the human-subjects listening tests presented in the next subsection. Again, the mean STOI scores

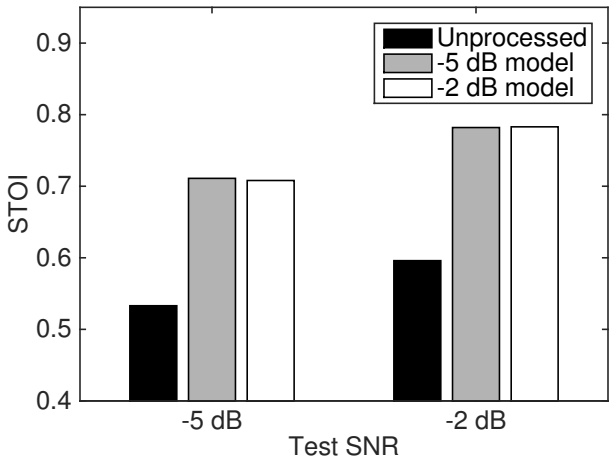


Figure 5.3: Short-time objective intelligibility (STOI) predictions for a noise-independent model trained and tested in matched and mismatched SNR conditions.

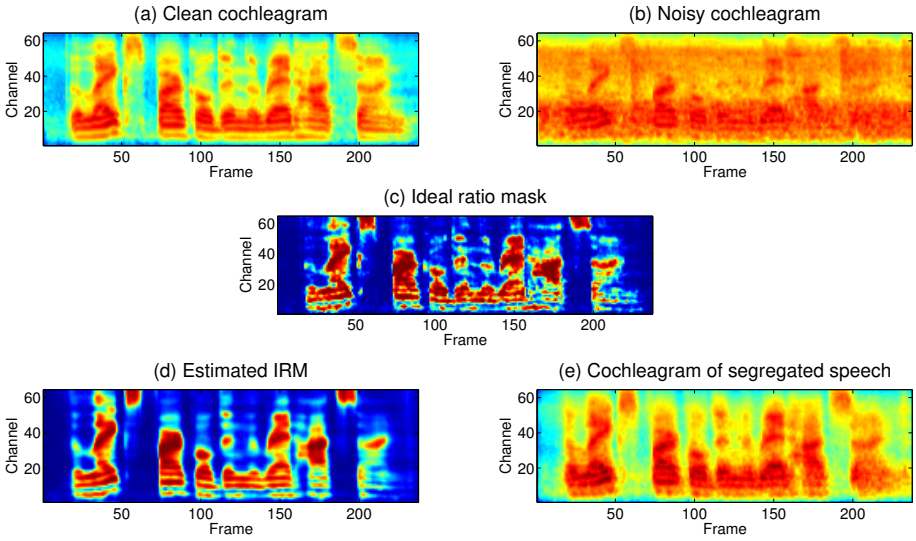


Figure 5.4: Segregation of an IEEE sentence (The lake sparkled in the red hot sun) from cafeteria noise at 0 dB SNR; (a) cochleagram of the utterance in quiet; (b) cochleagram of the utterance in noise; (c) IRM for this mixture; (d) estimated IRM for this mixture; and (e) cochleagram of the segregated utterance by applying the estimated IRM to the noisy utterance.

Table 5.2: STOI values for speech mixed with (unprocessed), and segregated from (processed), babble and cafeteria noise at the SNRs indicated.

	Babble noise		Cafeteria noise	
	Unprocessed	Processed	Unprocessed	Processed
5 dB	0.784	0.904	0.760	0.893
0 dB	0.663	0.834	0.642	0.823
−2 dB	0.612	0.792	0.596	0.783
−5 dB	0.541	0.707	0.533	0.708

were computed for the 160 test sentences in each test-noise condition. As shown in the table, the 10K-noise model substantially improves STOI values over unprocessed mixtures at all SNRs. For each SNR, similar STOI improvement was observed for the two noises, which was to be expected as the DNN was trained using a large number of noises, decreasing the likelihood of overfitting one specific noise.

5.3.2 Actual intelligibility results

Figure 5.5 shows intelligibility based on percentage of keywords reported by individual human listeners in each condition. Individual HI listeners are represented by filled symbols and NH listeners by open symbols. Scores on unprocessed speech in noise are represented by circles and those on algorithm-processed speech are represented by triangles. Algorithm benefit is therefore represented by the height of the line connecting these two symbols. As in Fig. 5.1, HI subjects are numbered and plotted in order of increasing PTA.

In the babble background, all but one HI subject received some benefit at the less favorable SNR. Benefit in this condition was 45 percentage points or greater for 4 of the 10 HI listeners and was 20 points or greater for 7 of the 10 HI listeners. At the more favorable babble SNR, 7 of 10 HI subjects received some benefit. Benefit in

this condition was reduced in magnitude compared to the less favorable SNR case, as most unprocessed scores were high. However, the HI listener with the lowest unprocessed score received a benefit of 42 percentage points. With regard to the NH listeners in babble noise, the majority also received some benefit (6 of 10 subjects at the less favorable SNR and 7 of 10 at the more favorable SNR). As in our previous work [39] [38], the benefit for the NH listeners was smaller than that obtained for the HI listeners.

In the cafeteria-noise background, all but one HI listener received some benefit at the less favorable SNR. Benefit in this condition was 20 percentage points or greater for 8 of the 10 HI listeners. At the more favorable cafeteria-noise SNR, 7 of 10 HI subjects received some benefit. The HI listener with the lowest unprocessed score in this condition received a benefit of 41 percentage points. For the NH listeners in cafeteria noise, the majority also received some benefit (9 of 10 subjects at the less favorable SNR and 6 of 10 at the more favorable SNR).

Group-mean intelligibility scores in each condition are displayed in Fig. 5.6. In babble, the average benefit from algorithm processing was 11.6 and 27.0 percentage points for the HI listeners at 5 and 0 dB SNR, and 10.3 and 8.1 percentage points for the NH listeners at -2 and -5 dB SNR, respectively. A series of planned comparisons (paired, uncorrected t tests) between unprocessed and processed scores in each panel of Fig. 5.6 indicated that algorithm processing produced significant increases in intelligibility for both HI and NH listeners at all babble SNRs $[t(9) \geq 1.8, p < 0.05]$.

In cafeteria noise, the average benefit from algorithm processing was 13.3 and 22.6 percentage points for the HI listeners at 5 and 0 dB SNR, and 4.3 and 10.3 percentage points for the NH listeners at -2 and -5 dB SNR, respectively. Planned comparisons

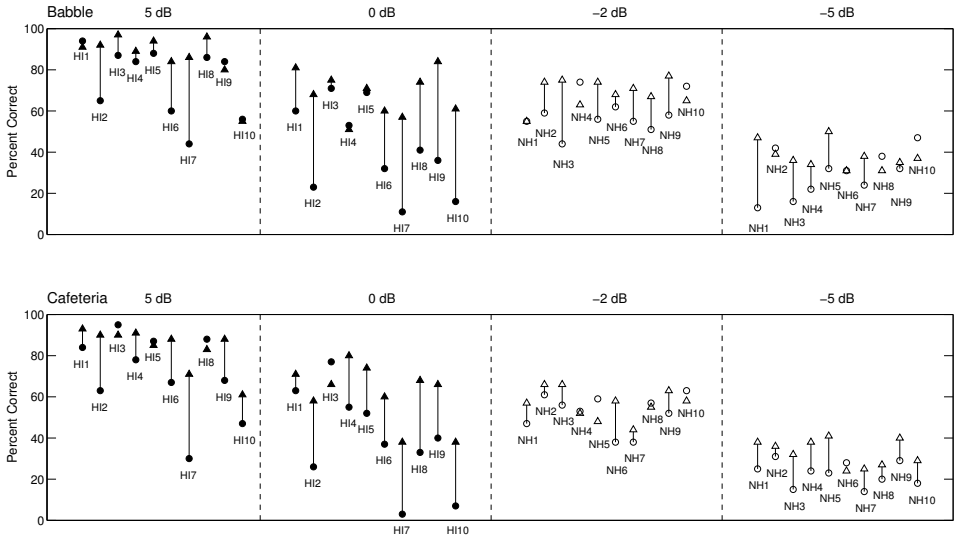


Figure 5.5: Intelligibility of IEEE sentences based on percentage of keywords reported. The top panels represent scores in, or segregated from, babble noise, and the bottom panels represent scores in, or segregated from, cafeteria noise, all at the SNRs indicated. Individual HI listeners are represented by filled symbols and individual NH listeners are represented by open symbols. Scores for unprocessed speech in noise are represented by circles and scores for algorithm-processed noisy speech are represented by triangles. Algorithm benefit is represented by the height of the line connecting these symbols.

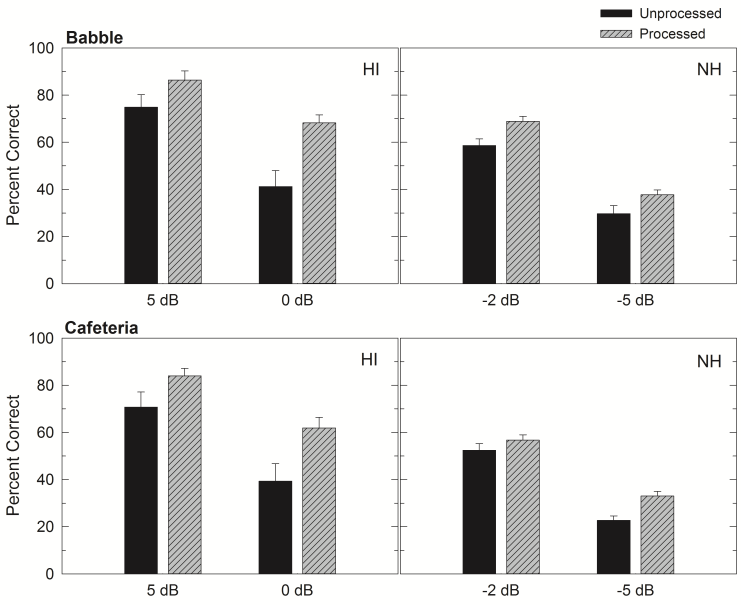


Figure 5.6: Group-mean intelligibility scores and standard errors for HI and NH listeners hearing unprocessed IEEE sentences in noise and sentences following algorithm processing. The top panels show scores for a babble background and the bottom panels show scores for a cafeteria-noise background, at the SNRs indicated.

indicated that algorithm processing produced significant increases in intelligibility for the HI listeners at both cafeteria-noise SNRs $[t(9) \geq 2.8, p \leq 0.01]$, and a significant increase in intelligibility for the NH listeners at the less favorable cafeteria-noise SNR $[t(9) = 5.1, p < 0.01]$.

5.4 General Discussion

It is worth emphasizing that, in the current study, two aspects of generalization have been addressed. First, the noise-independent algorithm trained on a large set of noises that did not include the noises employed for testing, and therefore it had to generalize to entirely novel noises. Second, the algorithm was trained at a single SNR, and it had to generalize to untrained SNRs during the test stage. These issues

represent some of the most difficult challenges that must be overcome for an algorithm to have direct translational significance. Obviously, the ability to generalize to noisy environments unseen during training is a requirement for an algorithm to be useful in real-world hearing technology. Despite these challenging demands, the current model produced substantial improvements in human intelligibility. A new group of NH listeners was tested currently in the unprocessed conditions that were identical to those in Healy et al. [38], which provides an opportunity for comparison. These conditions involve the same speech materials, noise recordings, and SNRs, just different random selections of noise for each noisy sentence. Despite the use of different NH subjects, scores across the two studies were within 1 percentage point on average across the four unprocessed conditions, and no condition differed by more than two percentage points across studies.

The current benefits for HI subjects observed from algorithm processing are somewhat smaller than when the algorithm was tested on novel segments of the same noise type used in training [38], rather than on new noises. However, much of this reduction in benefit can be attributed to the high unprocessed scores produced by the HI subjects employed in the current study. In accord with this generally better performance on unprocessed speech in noise, the PTAs of the current HI subjects are lower on average (reflecting less hearing loss) by 8.3 dB relative to their counterparts who participated in Healy et al. [38]. Despite the reduction in mean benefit due to high unprocessed scores, several aspects of the current results are particularly encouraging. First, those HI subjects having the lowest scores on unprocessed speech in noise received considerable benefit, in general accord with that observed when novel segments of the same noise recording were employed [38]. Second, the intelligibility

scores following the current algorithm processing were higher than the corresponding results in Healy et al. [38], particularly for the cafeteria noise.

A third aspect that may be considered encouraging is that no decrement in performance was observed even for those subjects who displayed very high unprocessed speech-in-noise scores. This ability to avoid decrements in performance when unprocessed intelligibility is high is almost as important as the ability to improve performance when unprocessed intelligibility is low. Even when considering only the current conditions in which HI performance on unprocessed speech was 85% or above (see Fig. 5.5) a benefit of 1.6 percentage points was still observed. This result is consistent with our previous results on this topic [39] [38], and suggests that the algorithm did not produce distortions that might potentially hinder better users. A result that has been seen consistently [39] [38] is that benefit demonstrated by HI listeners is greater than that displayed by NH listeners. This may be understood in terms of the different noise tolerances of the two groups. Hearing-impaired listeners are largely intolerant of noise and benefit considerably from algorithmic reduction of background noise. In contrast, NH listeners perceive speech in noise with considerable ease. Because they perform this task effectively in the unprocessed conditions, they benefit less from automatic speech segregation.

One result that differs from Healy et al. [38] involves the benefit demonstrated by the NH listeners. In the 2015 report, the NH listeners demonstrated a sizeable benefit when listening to speech extracted from babble noise, but not for speech extracted from cafeteria noise. In the current study, the NH listeners received similar degrees of benefit in both noise types. Interestingly, the STOI improvements in Table 5.2 are also similar for both noise types. This similarity in benefit across the two

noise types may be understood in terms of the current algorithm training procedure. Because neither of the test noises were employed during algorithm training, the generalization to them was similar and the algorithm produced similar degrees of STOI improvement. This differs from the 2015 training procedure in which generalization was tested on unseen segments of the same noise recording. In those 2015 conditions, it is apparent that generalization to a novel segment of cafeteria noise was more challenging algorithmically than generalization to a novel segment of babble, reflected by considerably larger STOI improvements for the babble noise (see Table I in Healy et al. [38]). This is likely because the cafeteria noise is more dynamic, with more transient bursts than the babble noise.

Finally, human robustness to noise may have much to do with our extensive exposure to a wide range of noisy conditions. It is documented that children show elevated speech reception thresholds relative to adults when recognizing noisy speech [32] [57]. Musicians exhibit higher intelligibility in speech-in-noise conditions compared to non-musicians [87], presumably because of their more extensive experience in listening to polyphonic signals. Bilingual speakers have a deficit in speech perception in noisy conditions compared to monolingual speakers, even though these two groups show similar performance in quiet [106]. All these effects are consistent with the idea that extensive training (or experience) is crucial for the remarkable noise robustness of the normal auditory system.

5.5 Conclusion

A DNN-based supervised speech segregation system with large-scale training was presented and shown to generalize to untrained noises and SNR conditions. Speech

intelligibility benefits were observed for HI listeners in both test noises and at both test SNRs. Normal-hearing listeners displayed a benefit at both test SNRs for multitalker babble noise, and at the less favorable SNR for the cafeteria noise. The current results represent a stride toward using supervised speech segregation in real-world environments.

CHAPTER 6

MODELING TEMPORAL DYNAMICS FOR SPEAKER GENERALIZATION

This chapter presents an approach to improve speaker generalization of noise-independent models. The work presented in this chapter has been published in the *Proceedings of 2016 Annual Conference of International Speech Communication Association* [14]. A journal version of this paper is under review by the *Journal of the Acoustical Society of America*.

6.1 Introduction

DNNs have been very successful in supervised separation [120] [127] [50]. Recent listening tests demonstrate that IRM estimation using a DNN substantially improves speech intelligibility of hearing-impaired and normal hearing listeners [39] [20]. For supervised learning tasks, generalizing to unseen conditions is a critical issue. Noise generalization and speaker generalization are two important aspects for supervised speech separation. The first aspect has been investigated in Chapter 4 and Chapter 5. With noise expansion through frequency perturbation, a model trained on one noisy type performs well with unseen segments of the same noise type [19] [38]. A DNN-based IRM estimator, when trained with a large variety of noises but a fixed

speaker, generalizes to unseen noises and unseen SNRs, and leads to clear speech intelligibility improvement [20]. However, it remains unknown how well such a model generalizes to unseen speakers and unseen noises at the same time.

In this study, we investigate speaker generalization of noise-independent models. To illustrate the problem, we first evaluate a speaker-dependent DNN on both seen and unseen speakers. A five-hidden-layer DNN is trained on 320,000 mixtures created using 67 utterances of a female speaker and 10,000 noises. A test set is created from another 25 utterances of the same female speaker and an unseen babble noise at -5 dB SNR. Then, we create another two test sets with an unseen female speaker and an unseen male speaker. Figure 6.1 shows the performance of the speaker-dependent DNN on seen and unseen speakers in terms of the STOI score [105]. As expected, the speaker-dependent DNN significantly improves STOI for the seen speaker. However, for both unseen speakers, the STOI scores of processed speech do not improve over those of unprocessed speech; They are actually lower. A DNN trained on a single speaker seems ineffective in separating a new speaker from background noise.

A straightforward approach for speaker generalization is to train a DNN-based IRM estimator on a large number of speakers and noises. Our experiments (see Section 6.4) indicate that, unfortunately, a DNN does not appear to be effective in modeling many speakers. Even with a large number of training speakers, a DNN still performs rather poorly on unseen speakers. A recent study [67] also shows performance degradation of a speaker-generic model compared to a speaker-specific model. A less challenging setting, which we call speaker-set-dependent, is to train a model with a closed set of speakers and test it on the same speakers. Our experimental results show that the performance of a speaker-set-dependent DNN on seen speakers

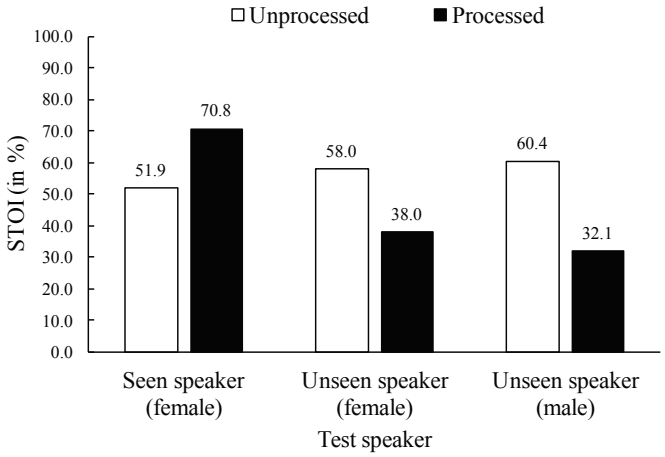


Figure 6.1: Performance of a speaker-dependent DNN on seen and unseen speakers with a babble noise in terms of STOI (in %) at -5 dB SNR.

degrades as the number of training speakers increases. Unlike a DNN trained on a single speaker, a speaker-set-dependent DNN is exposed to many speakers during training and therefore learns to detect speech patterns for many different speakers. While a speaker-dependent DNN focuses on separating one speaker from background noise, a set-dependent DNN has to search for many potential speakers. When the background noise contains speech components (e.g. babble noise), a speaker-set-dependent DNN is likely to mistake interfering speech for target speech since the patterns of interfering speech may resemble those of some training speakers.

A strategy to resolve the confusability of target speech and noise is for a speaker-set-dependent model to detect and focus on a target speaker. One such method is to train many speaker-dependent models and use speaker identification for model selection. However, this method has several potential limitations. First, the performance on seen speakers depends on the accuracy of speaker identification, which is known to be challenging in noisy environments [132]. Second, it is limited to the closed

set of trained speakers; For an unseen speaker, it needs to find a way to align the speaker to a similar trained speaker, which can also be difficult. A related method based on non-negative matrix factorization (NMF) learns a dictionary for each training speaker, and identifies a few speakers to approximate an unseen speaker during testing [102]. However, selecting appropriate speaker dictionaries can be challenging with nonstationary noises.

A supervised mask estimator typically uses a window of consecutive time frames to extract features to provide a useful context for improved mask estimation at a current frame. In other words, each mask frame is estimated independently given a context window containing limited temporal information about a target speaker. However, even with a long context window, the information beyond the window is not utilized. Mask estimation at a current frame can potentially benefit if a model utilizes earlier observations to characterize the target speaker. Therefore, supervised speech separation may be better formulated as a sequence-to-sequence mapping where a sequence of mask frames is predicted from a sequence of acoustic features.

In this study, we propose a model to separate unseen speakers from unseen noises. Our model is based on an RNN and accounts for temporal dynamics of speech. An RNN has self connections to feed back previous hidden activations, unlike a DNN which is a feedforward network. For a multilayer RNN, both low-level and high-level features of the previous time step are carried forward to facilitate learning of long-term dependencies. Given an incoming stream of noisy speech, our model analyzes and separates a target speaker from noise. The model learns from previous frames to focus on the target speaker for better speaker generalization.

This chapter is organized as follows. Section 6.2 describes the proposed model in detail. Experimental setup is discussed in Section 6.3. We present and analyze experimental results in Section 6.4. Section 6.5 concludes the chapter.

6.2 System Description

For speaker-independent speech separation, effectively modeling a target speaker is crucial. Given that characterizing a target speaker likely requires long-term observations, we propose to use RNNs to account for temporal dynamics of speech. A traditional DNN-based model only utilizes a window of features to capture temporal dynamics, which appears insufficient for speaker characterization for the sake of speech separation. In contrast, an RNN makes each mask prediction using information extracted from many previous frames.

To model temporal dependencies, an RNN is typically trained with back propagation through time (BPTT). A standard RNN suffers from the exploding and vanishing gradients during BPTT [8] [88]. While the exploding gradient problem can be mitigated using gradient clipping, the vanishing gradient problem prematurely stops an RNN from learning long-term dependencies. LSTM [43], a variant of RNN, mitigates the vanishing gradient problem by introducing a memory cell that facilitates the information flow over time. LSTM has been successful in modeling long temporal dependencies in many recent applications such as language modeling [104] [103], acoustic modeling [31] [93] and video classification [84]. While recent studies explored LSTM for speech enhancement [122] [27], our study focuses on speaker- and noise-independent speech separation. Figure 6.2 shows an LSTM block, which depicts a memory cell and three gates where the forget gate controls how much previous

information should be erased from the cell and the input gate controls how much information should be added to the cell. In this study, we use LSTM defined by the following equations [29]:

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \tag{6.1}$$

$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f) \tag{6.2}$$

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \tag{6.3}$$

$$z_t = g(W_{zx}x_t + W_{zh}h_{t-1} + b_z) \tag{6.4}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot z_t \tag{6.5}$$

$$h_t = o_t \odot g(c_t) \tag{6.6}$$

$$\sigma(s) = \frac{1}{1 + e^{-s}} \tag{6.7}$$

$$g(s) = \frac{e^s - e^{-s}}{e^s + e^{-s}} \tag{6.8}$$

where x_t, z_t, c_t, h_t represent input, block input, memory cell and hidden activation at time t, respectively. Input gate, forget gate and output gate are denoted as i_t, f_t and o_t, respectively. W's and b's denote weights and biases, respectively. $\odot$ represents element-wise multiplication or the gating operation. While the three gates are bounded to $[0, 1]$ by the function $\sigma(s)$, the output of an LSTM block is bounded to $[-1, 1]$ by both $\sigma(s)$ and $g(s)$. Note that the input gate i_t and the forget gate f_t are dependent on the current lower-layer input x_t and the previous hidden activation h_{t-1}. This dependency makes the updating of the memory cell context-sensitive, and therefore enables the modeling of complex temporal dynamics. With training by BPTT, LSTM supposedly learns to store task-relevant and context-sensitive information in its memory cells.

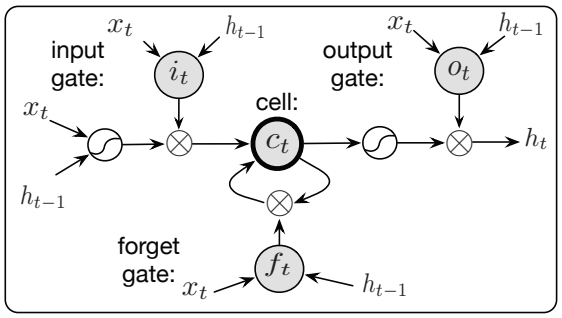


Figure 6.2: Diagram of an LSTM block with three gates and a memory cell.

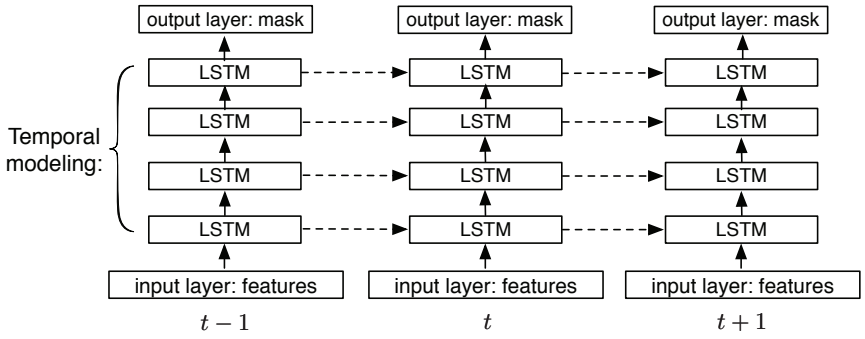


Figure 6.3: Diagram of the proposed system. Four stacked LSTM layers are used to model temporal dynamics of speech. Three time steps are shown here.

In supervised speech separation, we trained LSTM to maintain the speaker-sensitive information extracted from many previous frames to improve mask estimation for a current frame. The proposed system is illustrated in Fig. 6.3. We use four stacked hidden LSTM layers for temporal modeling and one output layer for mask estimation. We describe the system using the following equations:

$$y_t = \sigma(W_{out} h_t^{(L)} + b_{out}) \tag{6.9}$$

$$x_t^{(l+1)} = h_t^{(l)}, \text{ for } L > l \geq 1 \tag{6.10}$$

$$x_t^{(1)} = f_t \tag{6.11}$$

where f_t denotes acoustic features at time t. $x_t^{(l)}$ and $h_t^{(l)}$ represent the input and output of the LSTM block at layer l and time t, respectively. The estimated mask at time t is denoted as y_t. W_{out} and b_{out} represent the weight and bias of the output layer, respectively. While the bottom LSTM layer directly receives acoustic features, the other LSTM layers take the hidden activation from the LSTM layer below. The output layer takes the hidden activation $h_t^{(L)}$, $L = 4$, of the top LSTM layer, and estimates the IRM.

As shown in Fig. 6.3, compared to a DNN-based system which only passes information from the input layer to the output layer successively, an LSTM-based system adds multiple information pathways in the time dimension, where different pathways carry forward features at different levels of abstraction.

In this study, we use a feature window of 23 frames (11 to the left, 11 to the right) to estimate one frame of the IRM, which is defined on a 64-channel cochleagram with a 20-ms frame length and a 10-ms frame shift [115]. The estimated IRM is used to weight sub-band signals from a 64-channel gammatone filterbank. The weighted sub-band signals are summed to derive separated speech. The input features are 64-dimensional gammatone filterbank energies [20] extracted from noisy speech. From the input layer to the output layer, the proposed network has 23×64, 1024, 1024, 1024, 1024 and 64 units, respectively. In our evaluations, we compare the proposed RNN with a DNN baseline, which has five hidden layers with rectified linear units (ReLUs) [81] and one sigmoidal output layer. From the input layer to the output layer, the DNN has 23×64, 2048, 2048, 2048, 2048, 2048 and 64 units, respectively. Compared to the LSTM, this DNN is deeper and wider aside from no recurrent connections, and it provides a strong baseline.

6.3 Experimental Setup

6.3.1 Data preparation

We create large training sets with different numbers of training speakers to investigate speaker generalization of noise-independent LSTMs and DNNs. The trained models are tested on 6 seen speakers and 6 unseen speakers, both with unseen noises. Testing on multiple seen speakers is expected to be less challenging than testing on unseen speakers, and it serves as an intermediate step towards to speaker generalization.

In our experiments, we use 7138 utterances (83 speakers, about 86 utterances per speaker) from the WSJ0 SI-84 training set [90], which is widely used for speech separation and recognition evaluation. To create noisy speech, we use 10,000 training noises from a sound effect library (available at http://www.sound-ideas.com), and two highly-nonstationary test noises (babble and cafeteria) from an Auditec CD (available at http://www.auditec.com). Among the 83 speakers, all utterances of the 6 unseen speakers and the test utterances of 6 seen speakers are excluded from training. Since we investigate speaker generalization of noise-independent models, the two test noises are never used during training. We create the following two test sets:

- Test Set 1: 150 mixtures are created from 25×6 utterances of 6 seen speakers (3 males and 3 females) and random segments of the babble noise at −5 dB SNR.
- Test Set 2: 150 mixtures are created from 25×6 utterances of 6 unseen speakers (3 males and 3 females) and random segments of the babble noise at −5 dB SNR.

We create each training mixture by mixing an utterance with a random segment drawn from the 10,000 noises at a random SNR drawn from $\{-5, -4, -3, -2, -1, 0\}$ dB. To investigate the impact of the number of training speakers on speaker generalization, we evaluate three categories of models:

- Speaker-dependent models:

  For each speaker in Test Set 1 and Test Set 2, we train and test on the same speaker. Each training set has 320,000 mixtures and the total duration is about 500 hours.

- Speaker-set-dependent model:

  Five models are trained with {6, 10, 20, 40, 77} speakers including the 6 speakers of Test Set 1 and evaluated with Test Set 1. Each training set has 3,200,000 mixtures (about 5000 hours).

- Speaker-independent models:

  Five models are trained with {6, 10, 20, 40, 77} speakers and tested on the 6 unseen speakers of Test Set 2. Each training set includes 3,200,000 mixtures (about 5000 hours).

6.3.2 Optimization methods

We train the DNN and LSTM with the mean square error (MSE) cost function and the Adam optimizer [65] whose adaptive learning rates lead to faster convergence than standard stochastic gradient descent. The initial global learning rate is set to 0.001 and reduced by half every epoch. The best model is selected by cross validation. We use a mini-batch size of 256 for speaker-dependent DNNs. A mini-batch size of

4096 is used for speaker-set-dependent DNNs as we find a larger batch size slightly improves optimization. All LSTMs are trained with a mini-batch size of 256 and with truncated BPTT [123] of 250 time steps. For all LSTMs, we add 1 to the bias in Equation 6.4 to facilitate gradient flow and encourage learning of long-term dependencies in the beginning of training [58]:

$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f + 1) \tag{6.12}$$

6.4 Experimental Results and Analysis

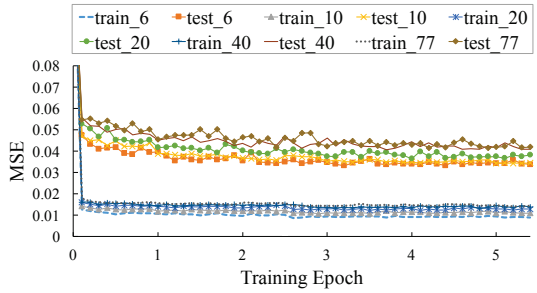
To evaluate the generalizability of the DNN and LSTM, we use three metrics including the MSE of the estimated mask, STOI and HIT−FA rate [64]. The latter compares an estimated binary mask with the IBM. Since we use the IRM as the learning target, we binarize it to compute HIT−FA. During binarization, the local criterion (LC) in the IBM definition is set to be 5 dB lower than the test SNR. Both the STOI and HIT−FA rate have been shown to correlate with human speech intelligibility well [39, 64].

Table 6.1: Comparison of the DNN and LSTM trained with 77 speakers in terms of the HIT−FA rate on the 6 seen speakers and unseen babble noise at −5 dB SNR.

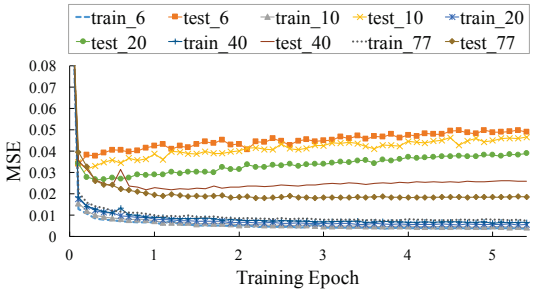
Model	HIT	FA	HIT−FA
DNN	83%	23%	60%
LSTM	89%	11%	78%

6.4.1 Performance trend on seen test speakers

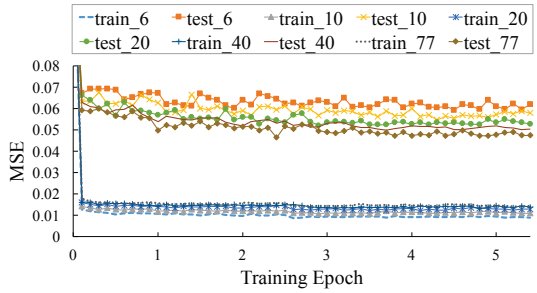
We evaluate the DNN and LSTM with 6 seen speakers. First, we train with the same 6 speakers. Figure 6.4 compares the training and test errors of the DNN and



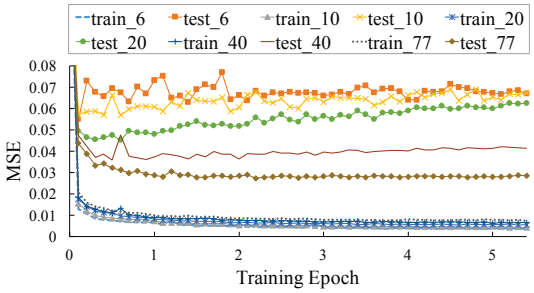
(a) DNN training/test errors on seen speakers



(b) LSTM training/test errors on seen speakers

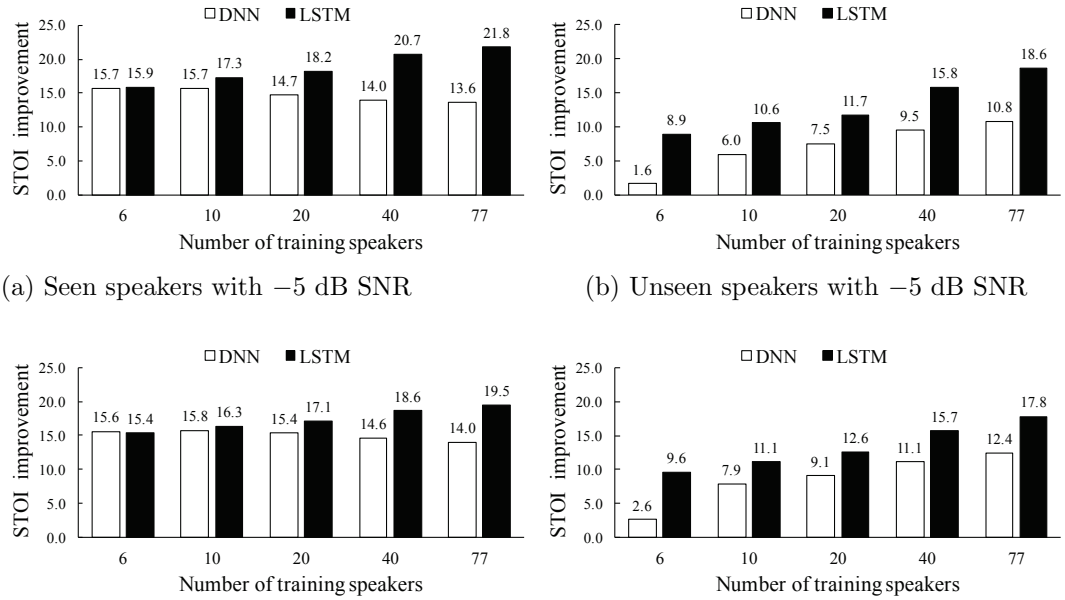


(c) DNN training/test errors on unseen speakers



(d) LSTM training/test errors on unseen speakers

Figure 6.4: Training and test errors of the DNN and LSTM as the number of training speakers increases. All models are evaluated with a test set of 6 seen speakers and a test set of 6 unseen speakers. Training mixtures are created with {6, 10, 20, 40, 77} speakers and 10,000 noises. The two test sets are created with the unseen babble noise at -5 dB SNR. All models are noise-independent. (a) Performance of the DNN on the 6 seen speakers. (b) Performance of LSTM on the 6 seen speakers. (c) Performance of the DNN on the 6 unseen speakers. (d) Performance of LSTM on the 6 unseen speakers.



(a) Seen speakers with −5 dB SNR

(b) Unseen speakers with −5 dB SNR

(c) Seen speakers with −2 dB SNR

(d) Unseen speakers with −2 dB SNR

Figure 6.5: Comparison of the DNN and LSTM in terms of STOI improvement (in %) with the unseen babble noise. (a) Performance of the DNN and LSTM on 6 seen speakers at −5 dB SNR. (b) Performance of the DNN and LSTM on 6 unseen speakers at −5 dB SNR. (c) Performance of the DNN and LSTM on 6 seen speakers at −2 dB SNR. (d) Performance of the DNN and LSTM on 6 unseen speakers at −2 dB SNR.

LSTM over training epochs. Figure 6.4(a) and Figure 6.4(b) show that the training errors of the DNN and LSTM drop significantly in the first epoch since each training set contains a very large number of training samples (about 5000 hours). Compared to the DNN, LSTM converges faster and then appears to overfit the training utterances of the 6 speakers. This is expected since LSTM models utterances as sequences and better fits training utterances. Indeed, LSTM reaches a lower training error than the DNN in all conditions. With a fixed training set size but an increasing number of training speakers, we observe performance degradation for the DNN but substantial performance boost for LSTM. The opposite trends for the DNN and LSTM reveal the capacity of LSTM in modeling a large number of speakers. Without utilizing the long-term context, the DNN treats all segments of training utterances as if they come from a single speaker. As the DNN is exposed to more training speakers, it becomes more challenging to separate a target speaker from the babble noise, whose local spectral-temporal patterns resemble those of speech. Table 6.1 shows the HIT$-$FA rates for the DNN and LSTM with the unseen babble noise at -5 dB SNR. Indeed, the DNN has a much lower HIT$-$FA rate than LSTM, and the DNN produces more than twice FA errors, implying that the DNN is more likely to mistake background noise as target speech. In contrast, with a large number of training speakers, LSTM appears to learn speech dynamics that are shared among speakers. Figure 6.5 compares the DNN and LSTM in terms of STOI improvement. Figure 6.5(a) shows that LSTM substantially outperforms the DNN when a large number of training speakers is used. With an increasing number of training speakers, the STOI improvement decreases for the DNN but increases for LSTM. In addition, we evaluate the models with a -2

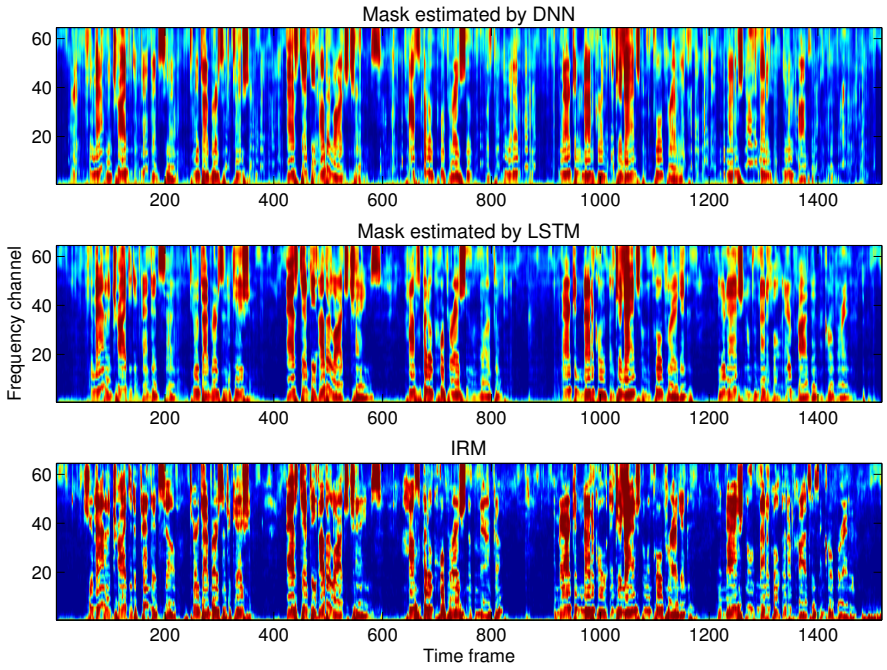


Figure 6.6: Visualization of the estimated masks by the DNN (top) and LSTM (middle) and the IRM (bottom). The mixture is created by mixing an unseen male speaker with the unseen babble noise at -5 dB SNR.

dB test set and observe consistent improvement of LSTM over the DNN, as shown in Fig. 6.5(c).

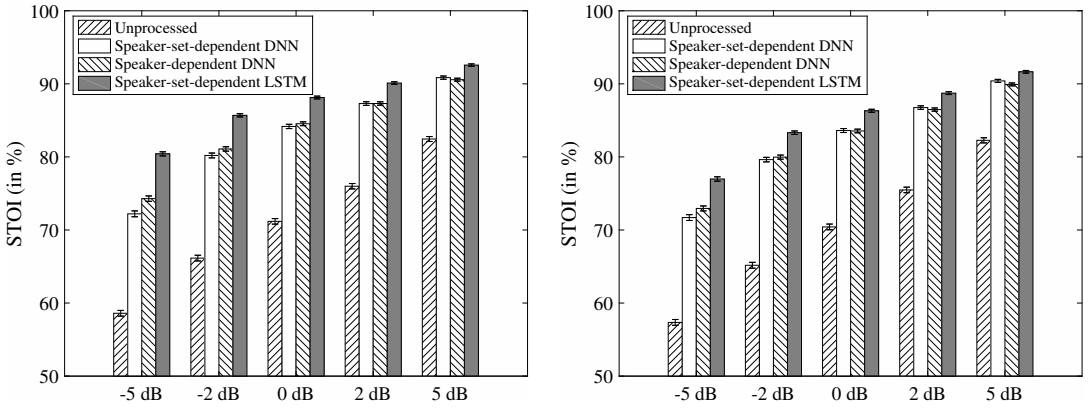
6.4.2 Performance trend on unseen test speakers

For the 6 unseen test speakers, Fig. 6.4(c), Fig. 6.4(d), Fig. 6.5(b) and Fig. 6.5(d) show that both the DNN and LSTM improve as the number of training speakers increases. Although the speaker-independent DNN benefits from more training speakers, the benefit diminishes quickly as the number of training speakers increases. Unable to utilize the long-term dependencies, the speaker-independent DNN appears

to only learn a generic speaker model from training speakers. As a result, the performance of the speaker-set-dependent DNN degrades somewhat as additional training speakers are added to the 6 seen speakers as it becomes more difficult to find a generic model to represent more speakers.

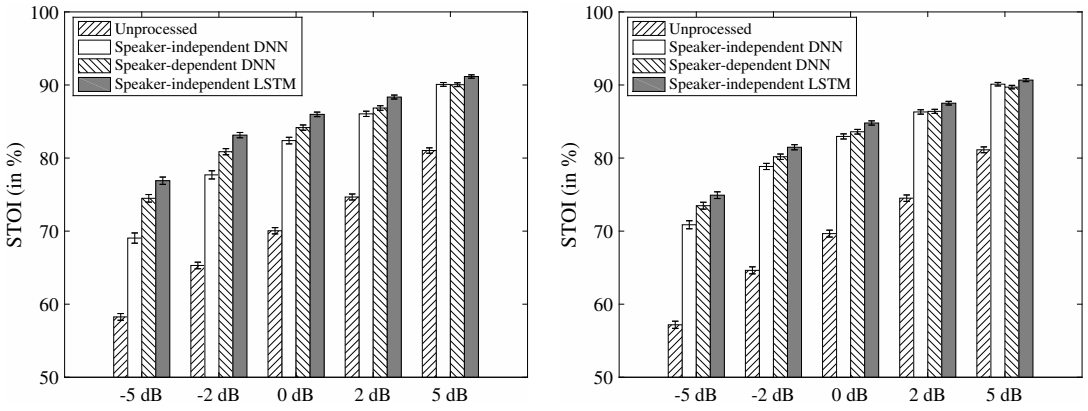
Compared to the speaker-independent DNN, the speaker-independent LSTM substantially improves the performance in terms of the MSE and the STOI improvement. The STOI improvement of LSTM is 7.8% higher than the DNN with the unseen babble noise at -5 dB SNR. This clearly indicates that LSTM achieves better speaker generalization than the DNN. We visualize estimated masks by the DNN and LSTM in Fig. 6.6, and observe that LSTM reduces the error of mistaking the background noise for target speech (e.g. around frame 850) and better preserves target speech (e.g. around frame 1425)

6.4.3 Model comparisons



(a) babble (b) Cafeteria

Figure 6.7: Comparison of speaker-set-dependent models (trained on 77 speakers and tested on 6 seen speakers) and speaker-dependent models in terms of STOI. Group means and standard errors are shown.



(a) babble (b) Cafeteria

Figure 6.8: Comparison of speaker-independent models (trained on 77 speakers and tested on 6 unseen speakers) and speaker-dependent models in terms of STOI. Group means and standard errors are shown.

We evaluate speaker-dependent, speaker-set-dependent and speaker-independent models with the babble and cafeteria noise at $\{-5, -2, 0, 2, 5\}$ dB SNRs. Fig. 6.7 compares speaker-set-dependent DNN, speaker-set-dependent LSTM and speaker-dependent DNN. The speaker-independent DNN, speaker-independent LSTM and speaker-dependent DNN are compared in Fig. 6.8. On the one hand, Fig. 6.7 show that speaker-set-dependent LSTM with 77 training speakers outperforms both speaker-dependent and speaker-set-dependent DNNs, indicating that LSTM learns from other speakers to improve the performance on the 6 seen speakers. On the other hand, as shown in Fig. 6.8, speaker-independent LSTM outperforms both speaker-dependent and speaker-independent DNNs on the 6 unseen speakers, especially at the very low SNR of -5 dB. LSTM also performs well at the unseen SNRs of 2 dB and 5 dB, demonstrating that LSTM generalizes to unseen noises, unseen speakers and unseen SNRs. We apply paired t-tests with a significance level of 0.01 and find

that the improvement of the LSTM over the DNN is statistically significant for both seen and unseen speakers at every test SNR.

In addition to the babble and cafeteria noise, we have tested speaker-independent DNN and LSTM on two other unseen noises, namely the factory noise and the speech shape noise (SSN). For the factory noise, LSTM improves the processed STOI over DNN by 3.7% and 2.0% at -5 dB and -2 dB, respectively. For SSN, LSTM improves by 5.0% and 2.0% at -5 dB and -2 dB, respectively.

6.4.4 Analysis of LSTM internal representations

As we discussed in Section 6.2, LSTM is supposed to memorize long-term contexts to help mask estimation at a current frame. We analyze what LSTM has learned by visualizing the memory cells c_t in Equation 6.5 across time frames. Since different memory cells have different dynamic ranges, we map the value range of each memory cell to $[0, 1]$ for better visualization:

$$c = \frac{c_t - c_{min}}{c_{max} - c_{min}} \tag{6.13}$$

where c_{min} and c_{max} denote the minimum and maximum values of a memory cell according to a long-term observation, respectively. Although the internal representations of LSTM are usually distributed and not intuitive, we find a few memory cells that exhibit interesting temporal patterns. We select three memory cells in the third LSTM layer and depict them in Fig. 6.9. As shown in the bottom three plots of Fig. 6.9, the first memory cell is excited by male speech and inhibited by female speech. The second cell is activated by female speech. The third one detects a silent interval following target speech after a few frames of delay. These patterns suggest that memory cells encode speech contexts.

Besides memory cells, LSTM also takes previous hidden activations as input. Therefore, the total information from previous time steps is encoded by both c_{t-1} and h_{t-1}. Since our proposed model has four LSTM layers, the past information can be represented as the concatenation of eight vectors:

$$v_{state} = \begin{bmatrix} {c_{t-1}^{(1)}}^T & {h_{t-1}^{(1)}}^T & \cdots & {c_{t-1}^{(4)}}^T & {h_{t-1}^{(4)}}^T \end{bmatrix}^T \tag{6.14}$$

To verify if v_{state} carries useful information, we reset it to a zero vector to erase past information at different time steps and examine the impact on subsequent mask estimation. We separately reset v_{state} in speech-dominant and noise-dominant intervals and visualize the resulting estimated masks in Fig. 6.10. The 6th and 9th plots of Fig. 6.10 show that resetting v_{state} during speech-dominant intervals does not make much difference as LSTM appears to quickly recapture the target speaker after observing strong target speech patterns in a few subsequent time steps. However, resetting v_{state} during noise-dominant intervals may degrade mask estimation for a considerable duration, as shown in the 7th and 8th plots of Fig. 6.10. LSTM is likely distracted by interfering speech contained in the background and focuses on wrong patterns until strong target-speech patterns are observed. In other words, LSTM seems to be context-aware and keep track of a target speaker for better mask estimation at a current frame.

6.4.5 Impact of future frames

In the above experiments, we use 23 time frames, including 11 future frames, of acoustic features for both the DNN and LSTM. Incorporating future frames improves mask estimation but impedes real-time implementation. To investigate the impact of future frames, we evaluate the models with different asymmetric windows on 6 unseen

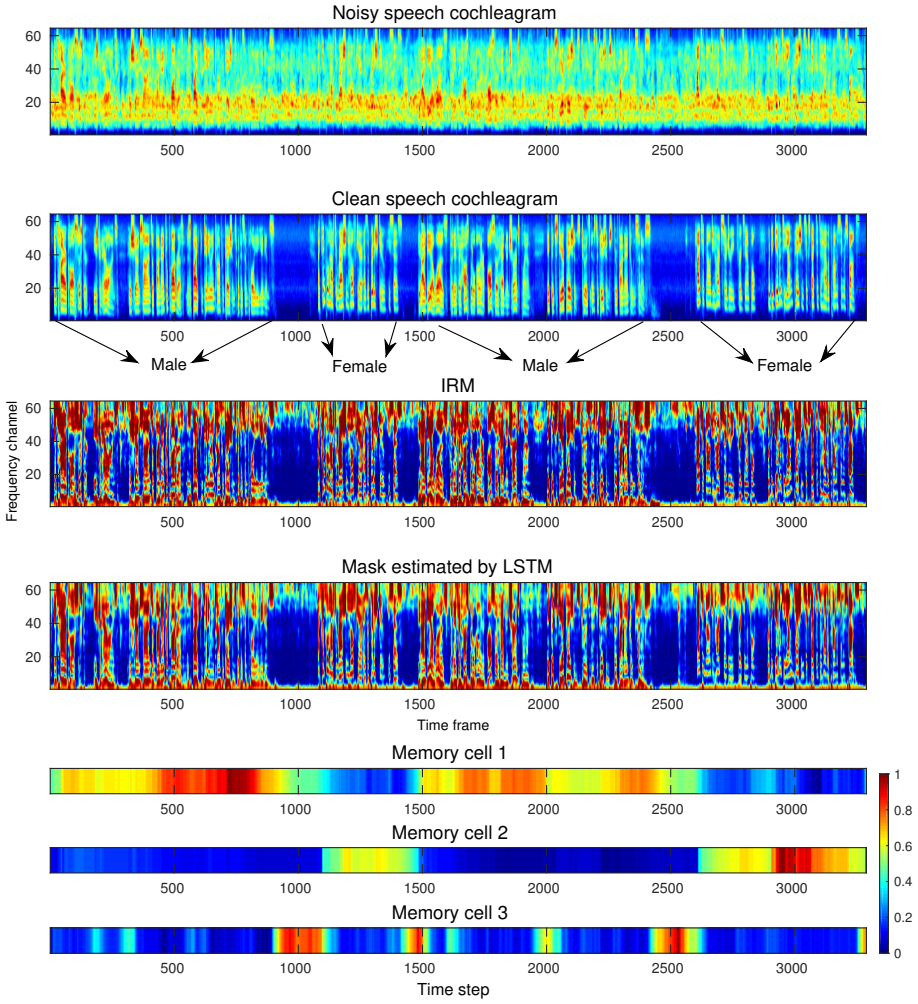


Figure 6.9: Visualization of speech patterns and memory cell values. Four utterances of two unseen speakers (male and female) are concatenated and mixed with the unseen babble noise at 0 dB SNR. The top four plots depict noisy speech cochleagram, clean speech cochleagram, the IRM and the estimated mask by LSTM, respectively. The bottom three plots show values of three different cells across time, respectively.

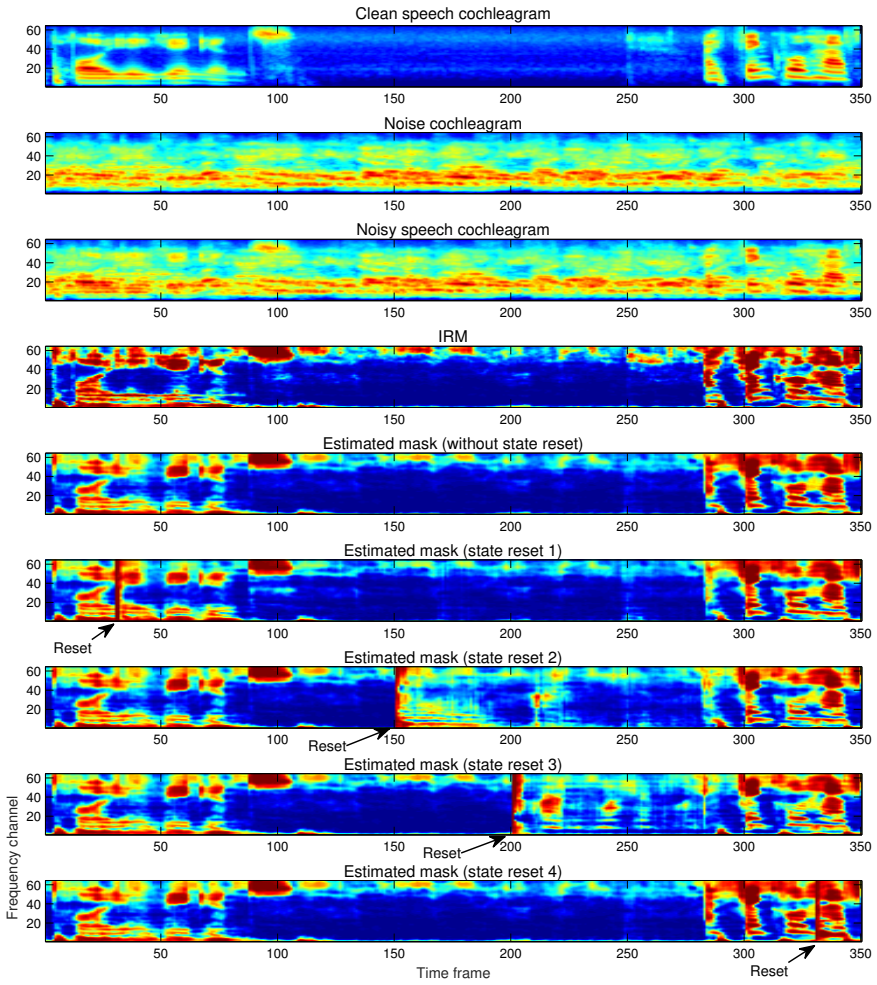
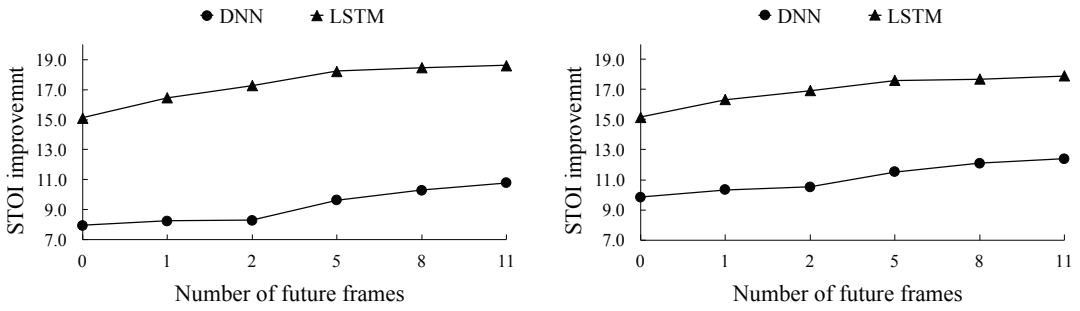


Figure 6.10: Impact of resetting the internal states of LSTM. The top five plots show the clean speech cochleagram, noise cochleagram, noisy speech cochleagram, the IRM and the estimated mask by LSTM, respectively. The 6th and 9th plots show the estimated masks when LSTM internal states are reset during speech-dominant intervals. The 7th and 8th plots show the estimated masks when LSTM internal states are reset during noise-dominant intervals.



(a) −5 dB SNR (b) −2 dB SNR

Figure 6.11: Impact of future frames on the performance of the DNN and LSTM in terms of STOI improvement (in %). The input contains 11 past frames, a current frame and {0, 1, 2, 5, 8, 11} future frames. The models are evaluated with 6 unseen speakers and the unseen babble noise. (a) Performance of the DNN and LSTM at −5 dB SNR. (b) Performance of the DNN and LSTM at −2 dB SNR.

speakers and the unseen babble noise at −5 dB and −2 dB SNRs. Each asymmetric window contains 11 past frames, a current frame and a different number of future frames. We do not decrease the past frames as they facilitate learning and do not violate causality. Figure 6.11 compares the impact of future frames on the DNN and LSTM. As shown in Fig. 6.11(a) and 6.11(b), LSTM substantially outperforms the DNN in all conditions. It is worth noting that LSTM without future frames still outperforms the DNN with 11 future frames, and gives about 15% STOI improvement over unprocessed speech in both SNR conditions.

6.5 Discussion

In this study, we have investigated speaker generalization of noise-independent models for supervised speech separation. Our previous investigation, which is discussed in Chapter 5, has demonstrated that a DNN, when trained with a large variety of noises but a fixed speaker, generalizes to unseen noises and unseen SNRs. However,

real world applications desire a model to perform well with both unseen speakers and unseen noises. Our experimental results show that training of a DNN with many speakers does not perform well on both seen and unseen speakers. This reveals the inefficiency of DNN in modeling a large number of speakers. As a DNN is exposed to more training speakers, the performance on seen speakers drops, suggesting that it fails to focus on a target speaker. A DNN makes independent mask estimation given a window of acoustic features, which appear insufficient to characterize a target speaker for the sake of speech separation.

We have proposed a separation model based on LSTM to improve speaker generalization. The proposed model treats mask estimation as a sequence-to-sequence mapping problem. By modeling temporal dynamics of speech, LSTM utilizes previous inputs to characterize and memorize a target speaker. Therefore mask estimation depends on both the current input and LSTM internal states. By visualizing the temporal patterns of LSTM memory cells, we find that the cell values correlate with speech patterns. Those memory cells capture different contexts to improve mask estimation at a current frame. By resetting LSTM internal states in both speech-dominant and noise-dominant intervals, we find that LSTM appears to detect and focus on a target speaker to help resolve the confusability of speech and noise patterns.

The proposed model substantially outperforms an already strong DNN baseline on both seen and unseen speakers. Interestingly, with more training speakers, the DNN performance on seen speakers degrades, while LSTM improves the results on seen speakers. This reveals the capacity of LSTM in modeling individual speakers. In addition, we have evaluated the dependency of DNN and LSTM on future frames for separation. Our experimental results show that LSTM without future frames

still significantly outperforms the DNN with 11 future frames. The proposed model represents a major step towards speaker- and noise-independent speech separation.

CHAPTER 7

CONTRIBUTIONS AND FUTURE WORK

7.1 Contributions

Since the formulation of supervised speech separation, generalization has been a critical issue. In this dissertation, we have identified and addressed several generalization aspects, including noise generalization, SNR generalization and speaker generalization. Specifically, we have explored acoustic features, noise augmentation, large-scale training and learning machines. The model proposed in Chapter 6 has been shown to generalize to unseen noises, unseen SNRs and unseen speakers, which represents a major stride in improving robustness of supervised speech separation.

In Chapter 3, we systematically evaluate an extensive list of acoustic features for supervised speech separation in low SNR conditions. The feature list includes two mel-domain features (MFCC and DSCC), two linear prediction features (PLP and RASTA-PLP), three gammatone-domain features (GF, GFCC and GFMC), one zero-crossing feature (ZCPA), three autocorrelation features (RAS-MFCC, AC-MFCC and PAC-MFCC), two medium-term filtering features (PNCC and SSF), two modulation features (GFB and AMS) and a set of pitch-based features. We find that gammatone domain features outperform other features. Besides the existing features, we have

proposed the MRCG feature, which incorporates both local information and broader spectrotemporal contexts. Among all evaluated features, the proposed MRCG feature performs the best for IBM estimation.

For noise-dependent speech separation, noise segment generalization is desirable. Our noise augmentation techniques presented in Chapter 4 improve noise segment generalization. First, we identify the issue that a DNN trained with limited noise samples generalizes poorly to unseen segments for nonstationary noises, such as the cafeteria noise. Then, three noise perturbation techniques, including noise rate perturbation, VTL perturbation and frequency perturbation, are investigated for noise augmentation. By evaluating on 6 environmental noises, we demonstrate that these three noise perturbation techniques improve noise segment generalization, with frequency perturbation performing the best. Specifically, frequency perturbation reduces the false-alarm error in mask estimation.

In Chapter 5, we propose large-scale training for noise type generalization, which leads to a noise-independent model. Compared to noise-dependent speech separation, noise-independent speech separation has three advantages. First, the performance of noise-dependent speech separation significantly degrades with unseen noises. Second, noise-dependent speech separation requires accurate noise or environment classification. Third, from the practical perspective, noise-independent speech separation is easier to deploy as it only requires one model for inference. We study how the amount of training noises impacts the generalizability of a DNN to unseen noises. Our experimental results suggest that a large DNN trained with 10,000 noise substantially outperforms the one trained on 100 noises, and matches the performance of noise-dependent models. It is the large variety of distinct noise samples rather than the

sheer size of training set that improves noise generalization. Besides noise generalization, we also demonstrate that a DNN generalizes to unseen SNRs. With subject tests, we demonstrate, for the first time, that supervised speech separation improves speech intelligibility for hearing-impaired listeners in unseen noises with unseen SNRs. This study represents a major step towards general supervised speech separation.

Besides noise generalization, real-world applications also require speaker generalization since the target speaker is usually unknown during training. In Chapter 6, we investigate speaker generalization for noise-independent models. Our first attempt for speaker generalization is training with many speakers. However, we observe poor generalization of a DNN with unseen speakers. Even with seen speakers, the performance of the DNN degrades as additional training speakers are added. Through error analysis, we find that a DNN trained on many speakers tends to make false-alarm errors, where the model mistakes interfering speech fragments for target speech. This reveals the inefficiency of a DNN for speaker generalization. To better resolve the confusability of target speech and background noise, we propose a speech separation model based on RNN with LSTM, which is expected to model temporal dynamics of speech. Our experimental results show that the proposed LSTM substantially outperforms the DNN, and that it generalizes to unseen speakers, unseen noises and unseen SNRs. Further experiments with LSTM internal states reveal that the trained LSTM encodes long-term context to better focus on a target speaker. Finally, we evaluate the impact of future time frames on the performance of the DNN and LSTM, and find that LSTM without future information outperforms the DNN with future information. As far as we know, the proposed LSTM clearly represents the state-of-the-art for speaker and noise generalization in supervised speech separation.

7.2 Future Work

In this dissertation, we have demonstrated that supervised speech separation is capable of generalizing to unseen noises, unseen SNRs and unseen speakers. As a result, supervised speech separation has come to a point where real-world applications are conceivable. In consideration of deploying supervised speech separation in real environments, we identify the following issues for future research:

- *Channel generalization.* As a data-driven approach, supervised speech separation learns patterns for speech and noise from training data. The speech corpora released in the research community are usually recorded with a single microphone or very few microphones. Since the recorded speech is shaped by a recording device, we must consider channel variations. In the ideal case, the same microphone is used for training and testing. However, it is often necessary to deploy a channel-independent model that performs on unseen devices since data collection and training for a new device is time-consuming and costly. One possible approach to deal with channel variations is to augment training data by simulating many microphones. The characteristics of a microphone can be approximately captured by its impulse response. Therefore, incorporating multiple microphone impulse responses for training will likely improve channel generalization. Compared to recording speech for a new device, measuring its impulse response in an anechoic chamber is more convenient. Another possible approach for dealing with channel variations is to perform feature transformation or model adaption with a small amount of data. Feature transformation learns a mapping from a training microphone to a test microphone, while model

adaptation adjusts a trained model to better accommodate the feature distribution of a new device. It is interesting to carry out experiments to evaluate these two approaches.

- *Quantitative evaluation of supervised speech separation with real recordings.* supervised speech separation has been systematically evaluated with artificially mixed noisy speech in terms of objective speech intelligibility measures and subject tests. To qualitatively evaluate its performance on real recordings, one can listen to unprocessed and processed recordings, and judge if speech intelligibility or quality improves. Indeed, our empirical evaluation suggests that a model trained on artificially mixed noisy speech works well for real recordings. However, it is difficult to compute objective speech intelligibility measures for real recordings where underlying clean speech is unavailable. Quantifying speech intelligibility with objective measures is useful for model development since listening tests are time-consuming and expensive. It would be useful to design a setting where both noisy speech and its underlying speech are recorded. One possible approach is to record clean speech and play it through high-fidelity speakers in a noisy environment for a second recording. A systematic study is needed for quantitative evaluation of supervised speech separation with real recordings.

- *Model Compression.* To deploy supervised speech separation in portable devices, we need to consider computational complexity of a model. DNNs and LSTMs used for speech separation typically contain large weight matrices, which are

slow to manipulate and do not fit low-memory or low-power devices. Therefore, it is necessary to study model compression. Model compression has been applied to image classification [36] [35] and machine translation [96]. To reduce the redundant information contained in large neural networks, two commonly used techniques are weight quantization and weight pruning. It is worth studying these techniques for low-computation and low-memory supervised speech separation.

- *High-fidelity speech separation.* Most of supervised speech separation algorithms operate in spectral-magnitude domain or cochlear domain. The separated speech can be highly intelligible but of low quality. With masking in the spectral-magnitude domain, the phase in not enhanced. With masking in the cochlear domain, speech distortion is introduced. A recent study estimates the complex ideal ratio mask (cIRM) to improve speech quality [124]. However, high-fidelity speech separation remains a challenge. One possible approach is to incorporate prior knowledge about clean speech. For example, we can learn a generative model for speech and use it to further enhance the separated speech produced by a current pipeline. High-fidelity speech separation is especially appealing for applications like enhanced telecommunication, where speech quality, not intelligibility, is the focus.

This dissertation has demonstrated the power of supervised speech separation in dealing with unseen conditions. The generalization capability is substantially improved by the proposed methods. With further advances on high-fidelity and low-complexity models, supervised speech separation is expected to elevate the performance of many human-centered speech processing applications such as hearing aids and telecommunication over mobile devices [114].

BIBLIOGRAPHY

[1] H. Abrams and J. Kihm. An introduction to MarkeTrak IX: A new baseline for the hearing aid market. *Hearing Review*, 22:16, 2015.

[2] M. Ahmadi, V. L. Gross, and D. G. Sinex. Perceptual learning for speech in noise after application of binary time-frequency masks. *J. Acoust. Soc. Am.*, 133:1687–1692, 2013.

[3] ANSI. *ANSI S3. 39 (R2012), American national standards institute specifications for instruments to measure aural acoustic impedance and admittance (aural acoustic immittance).* New York: ANSI, 1987.

[4] ANSI. *ANSI S3.21 (R2009), Methods for manual pure-tone threshold audiometry.* New York: ANSI, 2004.

[5] ANSI. *ANSI S3.6, American national standard specification for audiometers.* New York: ANSI, 2010.

[6] F. R. Bach and M. I. Jordan. Blind one-microphone speech separation: A spectral learning approach. In *Proc. NIPS*, pages 65–72, 2004.

[7] J. Barker, E. Vincent, N. Ma, H. Christensen, and P. Green. The PASCAL CHiME speech separation and recognition challenge. *Computer Speech & Language*, 27:621–633, 2013.

[8] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Networks*, 5:157–166, 1994.

[9] S. F. Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Trans. Acoust. Speech Signal Process.*, 27:113–120, 1979.

[10] A. S. Bregman. *Auditory scene analysis: The perceptual organization of sound.* Cambridge MA: MIT Press, 1990.

[11] D. S. Brungart, P. S. Chang, B. D. Simpson, and D. L. Wang. Isolating the energetic component of speech-on-speech masking with ideal time-frequency segregation. *J. Acoust. Soc. Am.*, 120:4007–4018, 2006.

[12] D. Byrne and H. Dillon. The National Acoustic Laboratories'(NAL) new procedure for selecting the gain and frequency response of a hearing aid. *Ear and Hear.*, 7:257–265, 1986.

[13] C. Chen and J. A. Bilmes. MVA processing of speech features. *IEEE Trans. Audio, Speech, Lang. Process.*, 15:257–270, 2007.

[14] J. Chen and D. L. Wang. Long short-term memory for speaker generalization in supervised speech separation. In *Proc. INTERSPEECH*, pages 3314–3318, 2016.

[15] J. Chen and D. L. Wang. DNN based mask estimation for supervised speech separation. In S. Makino, editor, *Audio Source Separation*. Berlin Heidelberg: Springer, to appear.

[16] J. Chen, Y. Wang, and D. Wang. A feature study for classification-based speech separation at low signal-to-noise ratios. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 22:1993–2002, 2014.

[17] J. Chen, Y. Wang, and D. L. Wang. A feature study for classification-based speech separation at very low signal-to-noise ratio. In *Proc. ICASSP*, pages 7039–7043, 2014.

[18] J. Chen, Y. Wang, and D. L. Wang. Noise perturbation improves supervised speech separation. In *Proc. LVA/ICA*, pages 83–90, 2015.

[19] J. Chen, Y. Wang, and D. L. Wang. Noise perturbation for supervised speech separation. *Speech Communication*, 78:1–10, 2016.

[20] J. Chen, Y. Wang, S. E. Yoho, D. L. Wang, and E. W. Healy. Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises. *J. Acoust. Soc. Am.*, 139:2604–2612, 2016.

[21] D. Ciresan, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. In *Proc. CVPR*, pages 3642–3649, 2012.

[22] G. E. Dahl, T. N. Sainath, and G. E. Hinton. Improving deep neural networks for LVCSR using rectified linear units and dropout. In *Proc. ICASSP*, pages 8609–8613, 2013.

[23] M. Delfarah and D. L. Wang. Features for masking-based monaural speech separation in reverberant conditions. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, to appear.

[24] H. Dillon. *Hearing aids, 2nd ed.* Turramurra Australia: Boomerang, 2012.

[25] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, 12:2121–2159, 2011.

[26] Y. Ephraim and D. Malah. Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. *IEEE Trans. Acoust., Speech, Sig. Process.*, 32:1109–1121, 1984.

[27] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux. Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks. In *Proc. ICASSP*, pages 708–712, 2015.

[28] J. S. Erkelens, R. C. Hendriks, R. Heusdens, and J. Jensen. Minimum mean-square error estimation of discrete fourier coefficients with generalized gamma priors. *IEEE Trans. Audio, Speech, Lang. Process.*, 15:1741–1752, 2007.

[29] F. A. Gers, J. Schmidhuber, and F. Cummins. Learning to forget: Continual prediction with LSTM. *Neural Computation*, 12:2451–2471, 2000.

[30] S. Gonzalez and M. Brookes. A pitch estimation filter robust to high levels of noise (PEFAC). In *Proc. Euro. Sig. Process. Conf.*, pages 451–455, 2011.

[31] A. Graves, A. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *Proc. ICASSP*, pages 6645–6649, 2013.

[32] J. W. Hall III, J. H. Grose, E. Buss, and M. B. Dev. Spondee recognition in a two-talker masker and a speech-shaped noise masker in adults and children. *Ear and Hear.*, 23:159–165, 2002.

[33] K. Han and D. L. Wang. A classification based approach to speech segregation. *J. Acoust. Soc. Am.*, 132:3475–3483, 2012.

[34] K. Han and D. L. Wang. Towards generalizing classification based speech separation. *IEEE Trans. Audio, Speech, Lang. Process.*, 21:168–177, 2013.

[35] S. Han, H. Mao, and W. J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015.

[36] S. Han, J. Pool, J. Tran, and W. Dally. Learning both weights and connections for efficient neural network. In *Proc. NIPS*, pages 1135–1143, 2015.

[37] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proc. CVPR*, pages 770–778, 2016.

[38] E. W. Healy, S. E. Yoho, J. Chen, Y. Wang, and D. L. Wang. An algorithm to increase speech intelligibility for hearing-impaired listeners in novel segments of the same noise type. *J. Acoust. Soc. Am.*, 138:1660–1669, 2015.

[39] E. W. Healy, S. E. Yoho, Y. Wang, and D. L. Wang. An algorithm to improve speech recognition in noise for hearing-impaired listeners. *J. Acoust. Soc. Am.*, 134:3029–3038, 2013.

[40] R. C. Hendriks, R. Heusdens, and J. Jensen. MMSE based noise PSD tracking with low complexity. In *Proc. ICASSP*, pages 4266–4269, 2010.

[41] H. Hermansky. Perceptual linear predictive (PLP) analysis of speech. *J. Acoust. Soc. Am.*, 87:1738–1752, 1990.

[42] H. Hermansky and N. Morgan. RASTA processing of speech. *IEEE Trans. Speech, Audio Process.*, 2:578–589, 1994.

[43] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9:1735–1780, 1997.

[44] G. Hu and D. L. Wang. Speech segregation based on pitch tracking and amplitude modulation. In *Proc. WASPAA*, pages 79–82, 2001.

[45] G. Hu and D. L. Wang. Auditory segmentation based on onset and offset analysis. *IEEE Trans. Audio, Speech, Lang. Process.*, 15:396–405, 2007.

[46] G. Hu and D. L. Wang. Segregation of unvoiced speech from nonspeech interference. *J. Acoust. Soc. Am.*, 124:1306–1319, 2008.

[47] G. Hu and D. L. Wang. A tandem algorithm for pitch estimation and voiced speech segregation. *IEEE Trans. Audio, Speech, Lang. Process.*, 18:2067–2079, 2010.

[48] Y. Hu and P. C. Loizou. A comparative intelligibility study of single-microphone noise reduction algorithms. *J. Acoust. Soc. Am.*, 122:1777–1786, 2007.

[49] Y. Hu and P. C. Loizou. Environment-specific noise suppression for improved speech intelligibility by cochlear implant users. *J. Acoust. Soc. Am.*, 127:3689–3695, 2010.

[50] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis. Joint optimization of masks and deep recurrent neural networks for monaural source separation. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 23:2136–2147, 2015.

[51] C. Hummersone, T. Stokes, and T. Brookes. On the ideal ratio mask as the goal of computational auditory scene analysis. In G. R. Naik and W. Wang, editors, *Blind source separation*, pages 349–368. Berlin Heidelberg: Springer, 2014.

[52] IEEE. IEEE recommended practice for speech quality measurements. *IEEE Trans. Audio Electroacoust.*, 17:225–246, 1969.

[53] S. Ikbal, H. Misra, and H. Bourlard. Phase autocorrelation (PAC) derived robust speech features. In *Proc. ICASSP*, pages 133–136, 2003.

[54] N. Jaitly and G. E. Hinton. Vocal Tract Length Perturbation (VTLP) improves speech recognition. In *Proc. ICML Workshop on Deep Learning for Audio, Speech and Lang. Process.*, 2013.

[55] J. Jensen and R. C. Hendriks. Spectral magnitude minimum mean-square error estimation using binary and continuous gain functions. *IEEE Trans. Audio, Speech, Lang. Process.*, 20:92–102, 2012.

[56] Z. Jin and D. L. Wang. A supervised learning approach to monaural segregation of reverberant speech. *IEEE Trans. Audio, Speech, Lang. Process.*, 17:625–638, 2009.

[57] P. M. Johnstone and R. Y. Litovsky. Effect of masker type and age on speech intelligibility and spatial release from masking in children and adults. *J. Acoust. Soc. Am.*, 120:2177–2189, 2006.

[58] R. Jozefowicz, W. Zaremba, and I. Sutskever. An empirical exploration of recurrent network architectures. In *Proc. ICML*, pages 2342–2350, 2015.

[59] S. Kamath and P. Loizou. A multi-band spectral subtraction method for enhancing speech corrupted by colored noise. In *Proc. ICASSP*, pages 44164–44164, 2002.

[60] N. Kanda, R. Takeda, and Y. Obuchi. Elastic spectral distortion for low resource speech recognition with deep neural networks. In *Proc. ASRU*, pages 309–314, 2013.

[61] C. Kim and R. Stern. Power-normalized cepstral coefficients (PNCC) for robust speech recognition. In *Proc. ICASSP*, pages 4101–4104, 2012.

[62] C. Kim and R. M. Stern. Nonlinear enhancement of onset for robust speech recognition. In *Proc. INTERSPEECH*, pages 2058–2061, 2010.

[63] 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. on Speech and Audio Process.*, 7:55–69, 1999.

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

[65] D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proc. ICLR*, 2015.

[66] U. Kjems, J. B. Boldt, M. S. Pedersen, T. Lunner, and D. L. Wang. Role of mask pattern in intelligibility of ideal binary-masked noisy speech. *J. Acoust. Soc. Am.*, 126:1415–1426, 2009.

[67] M. Kolbæk, Z.-H. Tan, and J. Jensen. Speech intelligibility potential of general and specialized deep neural network based speech enhancement systems. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 25:153–167, 2017.

[68] K. Kumar, C. Kim, and R. M. Stern. Delta-spectral cepstral coefficients for robust speech recognition. In *Proc. ICASSP*, pages 4784–4787, 2011.

[69] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. of the IEEE*, 86:2278–2324, 1998.

[70] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.

[71] N. Li and P. C. Loizou. Factors influencing intelligibility of ideal binary-masked speech: Implications for noise reduction. *J. Acoust. Soc. Am.*, 123:1673–1682, 2008.

[72] P. C. Loizou. *Speech enhancement: Theory and practice, 2nd ed.* Boca Raton FL: CRC press, 2013.

[73] Y. Lu and P. C. Loizou. A geometric approach to spectral subtraction. *Speech communication*, 50:453–466, 2008.

[74] H. Luts, K. Eneman, J. Wouters, M. Schulte, M. Vormann, M. Buechler, N. Dillier, R. Houben, W. A. Dreschler, M. Froehlich, et al. Multicenter evaluation of signal enhancement algorithms for hearing aids. *J. Acoust. Soc. Am.*, 127:1491–1505, 2010.

[75] H. K. Maganti and M. Matassoni. An auditory based modulation spectral feature for reverberant speech recognition. In *Proc. INTERSPEECH*, pages 570–573, 2010.

[76] T. May and T. Dau. Requirements for the evaluation of computational speech segregation systems. *J. Acoust. Soc. Am.*, 136:EL398–EL404, 2014.

[77] L. Meier, S. V. D. Geer, and P. Bühlmann. The group lasso for logistic regression. *J. Roy. Stat. Soc. Ser. B*, 70:53–71, 2008.

[78] A. Mohamed, G. E. Dahl, and G. E. Hinton. Acoustic modeling using deep belief networks. *IEEE Trans. Audio, Speech, Lang. Process.*, 20:14–22, 2012.

[79] N. Mohammadiha, P. Smaragdis, and A. Leijon. Supervised and unsupervised speech enhancement using nonnegative matrix factorization. *IEEE Trans. Audio, Speech, Lang. Process.*, 21:2140–2151, 2013.

[80] B. C. Moore. *Cochlear hearing loss: physiological, psychological and technical issues*. West Sussex England: John Wiley & Sons, 2007.

[81] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proc. ICML*, pages 807–814, 2010.

[82] A. Narayanan and D. L. Wang. Ideal ratio mask estimation using deep neural networks for robust speech recognition. In *Proc. ICASSP*, pages 7092–7096, 2013.

[83] S. K. Nemala, K. Patil, and M. Elhilali. A multistream feature frameork based on bandpass modulation filtering for robust speech recognition. *IEEE Trans. Audio, Speech, Lang. Process.*, 21:416–426, 2013.

[84] J. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In *Proc. CVPR*, pages 4694–4702, 2015.

[85] A. Ozerov, E. Vincent, and F. Bimbot. A general flexible framework for the handling of prior information in audio source separation. *IEEE Trans. Audio, Speech, Lang. Process.*, 20:1118–1133, 2012.

[86] K. K. Paliwal and L. D. Alsteris. On the usefulness of STFT phase spectrum in human listening tests. *Speech Communication*, 45:153–170, 2005.

[87] A. Parbery-Clark, E. Skoe, C. Lam, and N. Kraus. Musician enhancement for speech-in-noise. *Ear and hear.*, 30:653–661, 2009.

[88] R. Pascanu, T. Mikolov, and Y. Bengio. On the difficulty of training recurrent neural networks. In *Proc. ICML*, pages 1310–1318, 2013.

[89] R. Patterson, I. Nimmo-Smith, J. Holdsworth, and P. Rice. An efficient auditory filterbank based on the gammatone function. *Applied Psychology Unit Report 2341*, 1988.

[90] D. B. Paul and J. M. Baker. The design for the Wall Street Journal-based CSR corpus. In *Proc. of the workshop on Speech and Natural Language*, pages 357–362, 1992.

[91] N. Roman, D. Wang, and G. J. Brown. Speech segregation based on sound localization. *J. Acoust. Soc. Am.*, 114:2236–2252, 2003.

[92] S. T. Roweis. One microphone source separation. In *Proc. NIPS*, pages 793–799, 2000.

[93] H. Sak, A. W. Senior, and F. Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *Proc. INTERSPEECH*, pages 338–342, 2014.

[94] P. Scalart et al. Speech enhancement based on a priori signal to noise estimation. In *Proc. ICASSP*, pages 629–632, 1996.

[95] M. R. Schädler, B. T. Meyer, and B. Kollmeier. Spectro-temporal modulation subspace-spanning filter bank features for robust automatic speech recognition. *J. Acoust. Soc. Am.*, 131:4134–4151, 2012.

[96] A. See, M.-T. Luong, and C. D. Manning. Compression of neural machine translation models via pruning. *Proc. CoNLL*, pages 291–301, 2016.

[97] M. L. Seltzer, B. Raj, and R. M. Stern. A Bayesian classifier for spectrographic mask estimation for missing feature speech recognition. *Speech Communication*, 43:379–393, 2004.

[98] B. J. Shannon and K. K. Paliwal. Feature extraction from higher-lag autocorrelation coefficients for robust speech recognition. *Speech Communication*, 48:1458–1485, 2006.

[99] Y. Shao and D. L. Wang. Robust speaker identification using auditory features and computational auditory scene analysis. In *Proc. ICASSP*, pages 1589–1592, 2008.

[100] P. Smaragdis. Convolutive speech bases and their application to supervised speech separation. *IEEE Trans. Audio, Speech, Lang. Process.*, 15:1–12, 2007.

[101] S. Srinivasan, N. Roman, and D. Wang. Binary and ratio time-frequency masks for robust speech recognition. *Speech Communication*, 48:1486–1501, 2006.

[102] D. L. Sun and G. J. Mysore. Universal speech models for speaker independent single channel source separation. In *Proc. ICASSP*, pages 141–145, 2013.

[103] M. Sundermeyer, H. Ney, and R. Schlüter. From feedforward to recurrent LSTM neural networks for language modeling. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 23:517–529, 2015.

[104] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Proc. NIPS*, pages 3104–3112, 2014.

[105] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen. An algorithm for intelligibility prediction of time-frequency weighted noisy speech. *IEEE Trans. Audio, Speech, Lang. Process.*, 19:2125–2136, 2011.

[106] D. Tabri, K. M. S. A. Chacra, and T. Pring. Speech perception in noise by monolingual, bilingual and trilingual listeners. *Int. J. of Lang. & Comm. Disord.*, 46:1–12, 2015.

[107] S. Tamura. An analysis of a noise reduction neural network. In *Proc. ICASSP*, pages 2001–2004, 1989.

[108] S. Tamura and A. Waibel. Noise reduction using connectionist models. In *Proc. ICASSP*, pages 553–556, 1988.

[109] J. Thiemann, N. Ito, and E. Vincent. The diverse environments multi-channel acoustic noise database: A database of multichannel environmental noise recordings. *J. Acoust. Soc. Am.*, 133:3591, 2013.

[110] A. Varga and H. J. Steeneken. Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems. *Speech Communication*, 12:247–251, 1993.

[111] T. Virtanen. Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria. *IEEE Trans. Audio, Speech, Lang. Process.*, 15:1066–1074, 2007.

[112] T. Virtanen, J. F. Gemmeke, and B. Raj. Active-set Newton algorithm for overcomplete non-negative representations of audio. *IEEE Trans. Audio, Speech, Lang. Process.*, 21:2277–2289, 2013.

[113] D. L. Wang. On ideal binary mask as the computational goal of auditory scene analysis. In P. Divenyi, editor, *Speech separation by humans and machines*, pages 181–197. Boston MA: Kluwer Academic Pub., 2005.

[114] D. L. Wang. Deep learning reinvents the hearing aid. *IEEE Spectrum*, 54:32–37, 2017.

[115] D. L. Wang and G. J. Brown, editors. *Computational auditory scene analysis: Principles, algorithms and applications*. Hoboken NJ: Wiley-IEEE Press, 2006.

[116] D. L. Wang, U. Kjems, M. S. Pedersen, J. B. Boldt, and T. Lunner. Speech intelligibility in background noise with ideal binary time-frequency masking. *J. Acoust. Soc. Am.*, 125:2336–2347, 2009.

[117] Y. Wang, J. Chen, and D. L. Wang. Deep neural network based supervised speech segregation generalizes to novel noises through large-scale training. In *Technical Report OSU-CISRC-3/15-TR02*. OSU Department of Computer Science and Engineering, 2015.

[118] Y. Wang, K. Han, and D. L. Wang. Exploring monaural features for classification-based speech segregation. *IEEE Trans. Audio, Speech, Lang. Process.*, 21:270–279, 2013.

[119] Y. Wang, A. Narayanan, and D. L. Wang. On training targets for supervised speech separation. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 22:1849–1858, 2014.

[120] Y. Wang and D. L. Wang. Towards scaling up classification-based speech separation. *IEEE Trans. Audio, Speech, Lang. Process.*, 21:1381–1390, 2013.

[121] M. Weiss, E. Aschkenasy, and T. Parsons. Study and the development of the INTEL technique for improving speech intelligibility. Technical Report NSC-FR/4023, Nicolet Scientific Corporation, 1974.

[122] F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. Le Roux, J. R. Hershey, and B. Schuller. Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR. In *Proc. LVA/ICA*, pages 91–99, 2015.

[123] R. J. Williams and J. Peng. An efficient gradient-based algorithm for online training of recurrent network trajectories. *Neural Computation*, 2:490–501, 1990.

[124] D. S. Williamson, Y. Wang, and D. Wang. Complex ratio masking for monaural speech separation. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 24:483–492, 2016.

[125] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

[126] F. Xie and D. Van Compernolle. A family of MLP based nonlinear spectral estimators for noise reduction. In *Proc. ICASSP*, pages 53–56, 1994.

[127] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee. An experimental study on speech enhancement based on deep neural networks. *IEEE Signal Process. Lett.*, 21:65–68, 2014.

[128] C. Yu, K. K. Wójcicki, P. C. Loizou, J. H. Hansen, and M. T. Johnson. Evaluation of the importance of time-frequency contributions to speech intelligibility in noise. *J. Acoust. Soc. Am.*, 135:3007–3016, 2014.

[129] K. Yuo and H. Wang. Robust features for noisy speech recognition based on temporal trajectory filtering of short-time autocorrelation sequences. *Speech Communication*, 28:13–24, 1999.

[130] X.-L. Zhang and D. Wang. Boosting contextual information for deep neural network based voice activity detection. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 24:252–264, 2016.

[131] X. Zhao, Y. Shao, and D. L. Wang. CASA-based robust speaker identification. *IEEE Trans. Audio, Speech, Lang. Process.*, 20:1608–1616, 2012.

[132] X. Zhao, Y. Wang, and D. Wang. Robust speaker identification in noisy and reverberant conditions. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 22:836–845, 2014.