Deep Learning Based Array Processing for Speech Separation, Localization, and Recognition

DISSERTATION

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

By

Zhong-Qiu Wang, M.S.

Graduate Program in Computer Science and Engineering

The Ohio State University

2020

Dissertation Committee

Prof. DeLiang Wang, Advisor

Prof. Eric Fosler-Lussier

Prof. Mikhail Belkin

Abstract

Microphone arrays are widely deployed in modern speech communication systems. With multiple microphones, spatial information is available in addition to spectral cues to improve speech enhancement, speaker separation and robust automatic speech recognition (ASR) in noisy-reverberant environments. Conventionally, multi-microphone beamforming followed by monaural post-filtering is the dominant approach for multi-channel speech enhancement. This approach requires an accurate estimate of target direction, and power spectral density and covariance matrices of speech and noise. Such estimation algorithms usually cannot achieve satisfactory accuracy in noisy and reverberant conditions. Recently, riding on the development of deep neural networks (DNN), time-frequency (T-F) masking and spectral mapping based approaches have been established as the mainstream methodology for monaural (single-channel) speech separation, including speech enhancement and speaker separation. This dissertation investigates deep learning based microphone array processing and its application to speech separation and localization, and robust ASR.

We start our work by exploring various ways of integrating speech enhancement and acoustic modeling for single-channel robust ASR. We propose a training framework that jointly trains enhancement frontends, filterbanks and backend acoustic models. We also

apply sequence-discriminative training for sequence modeling and run-time unsupervised adaptation to deal with training and testing mismatches.

One essential aspect of multi-channel processing is sound localization. We utilize deep learning based T-F masking to identify T-F units dominated by target speaker and only use these T-F units for speaker localization, as they contain much cleaner phases that are informative for localization. This approach dramatically improves the robustness of conventional cross-correlation, beamforming and subspace based approaches for speaker localization in noisy-reverberant environments.

Building upon speaker localization, we next tightly integrate complementary spectral and spatial cues for deep learning based multi-channel speaker separation in reverberant environments. The key idea is to localize individual speakers and use the localization results to design spatial features that can indicate whether each T-F unit is dominated by the speech arriving from the estimated speaker direction. The spatial features are combined with spectral features in an enhancement network to extract the speaker from an estimated direction and with trained spectral structure. Strong separation performance has been observed on reverberant talker-independent speaker separation tasks.

Before addressing multi-channel speech enhancement, we explore various magnitude based phase reconstruction algorithms for monaural speaker separation. We also study complex spectral mapping based phase estimation, where we directly predict the real and imaginary components of target speech. We find that deep learning based magnitude estimates clearly benefit phase reconstruction, and complex spectral mapping leads to better phase estimation.

We then apply complex spectral mapping to multi-channel speech dereverberation and enhancement, where phase estimation is used to improve sound localization, time-invariant and time-varying beamforming, and post-filtering. State-of-the-art performance has been obtained on the enhancement and recognition tasks of the REVERB corpus and the CHiME-4 dataset.

Finally, for fixed-geometry arrays, we propose multi-microphone complex spectral mapping for speech dereverberation, where DNNs are used for time-varying non-linear beamforming. We find that concatenating multiple microphone signals for complex spectral mapping is a simple and effective way of integrating spectral and spatial information for fixed-geometry arrays.

To my twenties

Acknowledgments

When I read a dissertation, I used to jump to the acknowledgement for a glimpse of what the author has gone through, as life is way more intriguing than research. Now it comes to the end of my Ph.D. study and it is my turn to summarize my endeavor and dedication. I have been composing this section since 2016, adding and deleting content from now and then, as there are so many people to thank and so many reflections I would like to share. This section is the most valuable part of this dissertation.

I still remember that it was on August 14, 2013 when I first arrived at Ohio State, with one luggage and a dream. I once thought that doing a Ph.D. is certainly not easy, but still would be pleasant. However, what I did not expect is that it can be mentally, emotionally and physically draining, lasting for years. As the head of a leading lab, Dr. DeLiang Wang's expectation is high. He always thinks about breakthroughs and fundamental research. Many times, I was not able to make progress as progress means real-deal improvement. Even if improvement was obtained, the next thing I started to worry about was what to do next. I felt I had strength, but had no ideas where to use it. Very often I just sat in my cubicle with my brain in blank, pretending hardworking, mechanically checking the loss function after each epoch and worrying that my life would be a failure. I would like to thank my friends and officemates, Drs. Haoyu Fu, Yan Zhao, Feilong Liu and Lingyan Yin, who shared my happiness and sorrow in my darkest second and third year. I am glad

that our lifes overlap, and we learn from and grow with each other. I would also like to thank my labmates, Drs. Arun Narayanan, Kun Han, Yuxuan Wang, Donald Williamson, Jitong Chen, Yuzhou Liu and Masood Delfarah, for their help and guidance in these difficult times. I would especially like to thank my collaborators, Drs. Yan Zhao, Ke Tan, Peidong Wang and Hassan Taherian, for their valuable outputs. My thanks also go to Bobby Chen, Jonathan Lee, Xu Wei, Deserts Chang and GAI, whose music has accompanied, encouraged and inspired me along the way. Although many of my earlier studies are like permutation and combination of techniques, it is the period in which my knowledge was accumulated and momentum was built up for a productive second half of my Ph.D. study.

Thanks to Dr. DeLiang Wang's reference and Dr. Dong Yu's help, I received a summer internship offer from MSR-Redmond in Spring'16. I was so excited as interning at MSR-Asia was one of my dreams during my undergraduate years. I met so many like-minded young talents, like Drs. Yifei Huang, Yi Lu and Diyi Yang, and learned so much from them. As written in *The Ph.D. Grind*, these young talents would likely become future entrepreneurs, high-tech leaders and award-winning professors. I was so lucky to intern in the audio group led by Dr. Ivan Tashev, who is super nice, energetic and knowledgeable, and work with so many bright people, such as Drs. Amit Das, David Johnston and Shoaib Mohammed. I still remember the day when we were riding horses in Orcas island. That could be one of my happiest times in these years. I am glad that the work on speech emotion recognition culminated in two papers. It was also enjoyable to catch up with Ran Xia, my first friend at Ohio State back in 2013, and explore so many places in the lovely Evergreen state. In Summer'16, I was also collaborating with Dr. Xueliang Zhang on CHiME-3. I

thank him for introducing me to the area of beamforming, which became my later focus and led me to microphone array processing. In ICASSP'17, I was so fortunate to talk with Dr. Yuxuan Wang, who now has over 4,000 citations and is doing excellent work at ByteDance. I learned so much from him and his past experience. Things would have been quite different if I had gotten his advice earlier. Season 16/17 was an overall productive year. Ideas burst out continuingly, several algorithms were invented, papers accepted and candidacy exam also passed. With the help from Dr. DeLiang Wang, I finally managed to sail on the right track, and did not have to worry about what to do next at nights. Many times, I was so excited as I felt like I was sitting on a gold mine that could yield a series of high-quality papers, while some of them were scooped by others or did not come to fruition due to conflict of interests.

Every internship is like an escape from my plain university life, forcing me to embrace changes and explore new opportunities. In Summer'17, I was so fortunate to work in the speech group at MERL, one of the best research labs in speech and audio. My enthusiasm was as high as the temperature in Boston as the research at MERL exactly matches mine. It was a super busy, intense and pressured summer, in which my second wave of momentum was built, paving my way towards an even more productive fifth year. I learned so much from my two mentors, Drs. Jonathan Le Roux and John R. Hershey, and Dr. Shinji Watanabe. Jonathan is so caring and thoughtful, with a clear roadmap and a well-organized codebase for my internship. I learned so much from him, especially on speech separation, phase reconstruction, coding and Chainer. John is so creative with deep insights into many problems that I had never thought about. They answered many of my questions and substantially elevated my research taste. When I look back, that summer is probably the

time in which I grow fastest in my entire career, and what I learned there has been playing a fundamental role in my later research. Luckily enough, one paper with them won a Best Student Paper Award at IEEE ICASSP 2018. This is totally unexpected and exciting, as such an award only existed in my wildest dreams. I feel so lucky getting the right guidance and help at a good point. The ride on speaker separation was incredible. Thank you, Jonathan and John! It was a nice experience exploring Boston with Dr. Haoyu Fu, and the amazing Arcadia and Cape Cod with Dr. Yi Lu.

Unlike others, I was still passionate about research even in my fifth year, waking up early in the freezing winter of Ohio. I felt like there was a fire burning in my heart, lighting my path towards a representative work. After four years' grind, I can finally produce linear and consistent output, obtaining tangible progress each day and getting closer and closer to a good study. 17/18 season was an even more productive year. Under Dr. DeLiang Wang's guidance, I finished six first-authored papers and drafted one funding proposal in two months of Autumn'17. In April'18, I finally came up with a novel perspective for phase reconstruction. This time I felt I found a diamond mine. Although the algorithms did not manage to shine in the long river of history, it is the first time I feel that I could build something from the ground up and make a real difference. It is also the first time I feel that doing research can be so exciting and produce such a strong sense of satisfaction. This feeling, I have to admit, is addictive. When performance is good, it feels like everything is blossoming, but when not good, life becomes dark and emotionally draining again.

In Summer'19, I interned in Dr. John R. Hershey's team in Google AI perception. It was a Google experience, where I learned a lot about product-level coding. It was nice working with so many smart people like Drs. Kevin Wilson, Scott Wisdom and Efthymios

Tzinis. I enjoyed the days and free meals there. Autumn'19 is a valley of my Ph.D. research, as older projects concluded and it was not easy to find a new and fundamental one. I gradually feel that doing research is more like a lifestyle, rather than fighting for sacred glory and dream, and that expectation should be lower. In the meantime, I was distracted and stopped, and learned a lesson. Overall, 19/20 season is like a gap year for me to think and sort things out. I feel so lucky that I can have another chance to grow before graduation, as there is a much bigger world outside research and there are so many things I need to improve. One thing I really recommend is to do a 5k run per day no matter what.

Day after day and year after year. On April 2, 2020, I defended my dissertation. It feels fast and never easy, so does the seven years spanning my prime twenties. I would like to express my sincere gratitude to my dissertation committee members, Drs. DeLiang Wang, Eric Fosler-Lussier and Mikhail Belkin (especially DeLiang), for their insightful comments and invaluable suggestions to my dissertation. I would also like to thank Drs. James W. Davis and Alan Ritter for serving on the committee of my candidacy exam.

Dr. DeLiang Wang, my dear advisor, uses his unique wisdom and vision to shape my writing, speaking and research abilities, carving me into an independent researcher out of a raw stone. I am so grateful that he can invest in a young man like me to grow and sprout. I appreciate the freedom he gave me in the past years and his patience when we had different views. Thank you, DeLiang!

I would also like to thank my undergraduate advisor, Dr. Yang Liu, who taught me so much about machine learning and helped me apply for Ph.D. programs. He is one of the key persons in my life to date. I feel so sorry about something I said about him. Indeed, only I can save myself.

Every Ph.D. knows the effort required to earn a Dr. before the name. I should really thank myself for pushing myself this far and never giving up. Thank you, Dr. Zhong-Qiu Wang!

In closing, I would like to thank my parents for their love and support (感谢爸爸妈妈这么多年来付出的爱，希望你们健康快乐).

The biggest two things I have learned in this journey are the meaning of *independence*, especially living independently and doing independent research, although painfully, and that life is very random. In Chongqing, my beloved hometown, people like to gather around, playing mahjong, appreciating a cup of tea and chatting about daily trivia. When I shared my stories, my uncle often talked to me “算老撒”, meaning let it go. I do not want to say yes or no on whether this effort is worthwhile, but if I were given another chance seven years ago, I would definitely do it again, but in a more hardworking, passionate, and courageous way. As what Dr. Like Hui once shared: some people, like Dr. John B. Goodenough who received the 2019 Nobel prize in chemistry, are real warriors. In their entire lifetime, they are always marching forward, defending their dreams, and fighting for glories.

Time waits for no one.

Vita

Sep. 1991	Born in Chongqing, China
2013	B.Eng. in Computer Science and Technology, Harbin Institute of Technology, Harbin, China
2017	M.Sc. in Computer Science and Engineering, The Ohio State University, Columbus, USA
2018	Best Student Paper Award at ICASSP 2018
2020	Graduate Research Award from CSE@OSU
2016	Summer research intern at Microsoft Research
2017	Summer research intern at Mitsubishi Electric Research Laboratories
2019	Summer research intern at Google AI

Publications

H. Taherian, Z.-Q. Wang, J. Chang, and D.L. Wang, "Robust Speaker Recognition Based on Single-Channel and Multi-Channel Speech Enhancement", in *IEEE/ACM Transactions on Audio, Speech, and Language Processing (IEEE/ACM T-ASLP)*, 2020.

Z.-Q. Wang and D.L. Wang, "Deep Learning Based Target Cancellation for Speech Dereverberation", in *IEEE/ACM T-ASLP*, vol. 28, pp. 941-950, 2020.

Z.-Q. Wang and D.L. Wang, "Multi-Microphone Complex Spectral Mapping for Speech Dereverberation", in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 486-490, 2020.

J. Le Roux, J. R. Hershey, Z. Wang, and G. P. Wichern, “Methods and Systems for End-to-End Speech Separation with Unfolded Iterative Phase Reconstruction”, US Patent 10,529,349, 2020.

H. Taherian, Z.-Q. Wang, and D.L. Wang, "Deep Learning Based Multi-Channel Speaker Recognition in Noisy and Reverberant Environments", in *Annual Conference of the International Speech Communication Association (Interspeech)*, pp. 4070-4074, 2019.

Z.-Q. Wang, K. Tan, and D.L. Wang, "Deep Learning Based Phase Reconstruction for Speaker Separation: A Trigonometric Perspective", in *ICASSP*, pp. 71-75, 2019.

Z.-Q. Wang and D.L. Wang, "Combining Spectral and Spatial Features for Deep Learning Based Blind Speaker Separation", in *IEEE/ACM T-ASLP*, vol. 27, pp. 457-468, 2019.

Z.-Q. Wang, X. Zhang, and D.L. Wang, "Robust Speaker Localization Guided by Deep Learning Based Time-Frequency Masking", in *IEEE/ACM T-ASLP*, vol. 27, pp. 178-188, 2019.

Y. Zhao, Z.-Q. Wang, and D.L. Wang, "Two-Stage Deep Learning for Noisy-Reverberant Speech Enhancement", in *IEEE/ACM T-ASLP*, vol. 27, pp. 53-62, 2019.

Z.-Q. Wang and D.L. Wang, "Integrating Spectral and Spatial Features for Multi-Channel Speaker Separation", in *Interspeech*, pp. 2718-2722, 2018.

Z.-Q. Wang, X. Zhang, and D.L. Wang, "Robust TDOA Estimation Based on Time-Frequency Masking and Deep Neural Networks", in *Interspeech*, pp. 322-326, 2018.

Z.-Q. Wang and D.L. Wang, "All-Neural Multi-Channel Speech Enhancement", in *Interspeech*, pp. 3234-3238, 2018.

Z.-Q. Wang, J. Le Roux, D.L. Wang, and J. R. Hershey, "End-to-End Speech Separation with Unfolded Iterative Phase Reconstruction", in *Interspeech*, pp. 2708-2712, 2018.

Z.-Q. Wang, J. Le Roux, and J. R. Hershey, "Multi-Channel Deep Clustering: Discriminative Spectral and Spatial Embeddings for Speaker-Independent Speech Separation", in *ICASSP*, pp. 1-5, 2018.

Z.-Q. Wang, J. Le Roux, and J. R. Hershey, "Alternative Objective Functions for Deep Clustering", in *ICASSP*, pp. 686-690, 2018.

Z.-Q. Wang and D.L. Wang, "On Spatial Features for Supervised Speech Separation and its Application to Beamforming and Robust ASR", in *ICASSP*, pp. 5709-5713, 2018.

Z.-Q. Wang and D.L. Wang, "Mask Weighted STFT Ratios for Relative Transfer Function Estimation and its Application to Robust ASR", in *ICASSP*, pp. 5619-5623, 2018.

I. Tashev, Z.-Q. Wang, and K. Godin, "Speech Emotion Recognition based on Gaussian Mixture Models and Deep Neural Networks", in *Information Theory and Applications Workshop*, pp. 1-4, 2017.

Y. Zhao, Z.-Q. Wang, and D.L. Wang, "A Two-stage Algorithm for Noisy and Reverberant Speech Enhancement", in *ICASSP*, pp. 5580-5584, 2017.

X. Zhang, Z.-Q. Wang, and D.L. Wang, "A Speech Enhancement Algorithm by Iterating Single- and Multi-microphone Processing and its Application to Robust ASR", in *ICASSP*, pp. 276-280, 2017.

Z.-Q. Wang and D.L. Wang, "Recurrent Deep Stacking Networks for Supervised Speech Separation", in *ICASSP*, pp. 71-75, 2017.

Z.-Q. Wang and I. Tashev, "Learning Utterance-level Representations for Speech Emotion and Age/Gender Recognition using Deep Neural Networks", in *ICASSP*, pp. 5150-5154, 2017.

Z.-Q. Wang and D.L. Wang, "Unsupervised Speaker Adaptation of Batch Normalized Acoustic Models for Robust ASR", in *ICASSP*, pp. 4890-4894, 2017.

Z.-Q. Wang and D.L. Wang, "A Joint Training Framework for Robust Automatic Speech Recognition", in *IEEE/ACM T-ASLP*, vol. 24, pp. 796-806, Apr. 2016.

Z.-Q. Wang, Y. Zhao, and D.L. Wang, "Phoneme-Specific Speech Separation", in *ICASSP*, pp. 146-150, 2016.

Z.-Q. Wang and D.L. Wang, "Robust Speech Recognition from Ratio Masks", in *ICASSP*, pp. 5720-5724, 2016.

D. Bagchi, M. Mandel, Z. Wang, Y. He, A. Plummer, and E. Fosler-Lussier, "Combining Spectral Feature Mapping and Multi-channel Model-based Source Separation for Noise-robust Automatic Speech Recognition", in *IEEE Automatic Speech Recognition and Understanding Workshop*, pp. 496-503, 2015.

Z.-Q. Wang and D.L. Wang, "Joint Training of Speech Separation, Filterbank and Acoustic Model for Robust Automatic Speech Recognition", in *Interspeech*, pp. 2839-2843, 2015.

Fields of Study

Major Field: Computer Science and Engineering

Table of Contents

List of Tables

Table **Page**

List of Figures

Figure **Page**

Chapter 1. Introduction

1.1. Motivation

Recent years have witnessed a dramatic demand in voice-based interfaces for speech communication, thanks in part to the wide adoption of deep learning. Amazon Echo and Google Home, which feature an intelligent voice-controlled assistant, have been sold to tens of millions of customers over the last five years. As such devices are deployed in homes and offices, major technical challenges arise including how to reliably localize and enhance a target speaker, separate competing speakers, and recognize their speech in everyday environments with room reverberation and environmental noises. Far-field ASR, for instance, is a widely acknowledged difficulty due to reverberation and noise.

Driven by Moore's law in the past decades, modern electronic devices have gained more and more computing capability. It is nowadays very common for a modern smart device to have more than one microphone. For example, Amazon Echo features seven microphones, Google Home two, and iPhone-7 has four microphones. An array of microphones produces multiple recordings at the same time. Similar to the human auditory system, spatial origins of the underlying sound sources can be computed from these recordings as, for each source, the signal arrives at each microphone at a different time. Such time difference of arrival (TDOA) information provides an informative cue

complementary to spectral (monaural) information for speech enhancement and separating multiple speakers; for example, one can enhance or maintain signals from a particular direction and suppress signals arriving from other angles.

Classical methods for multi-channel speech enhancement are mainly focused on using beamforming to combine multiple signals and utilizing post-filtering for further noise and reverberation reduction [40]. The beamforming approach designs a linear filter to boost or maintain the signal from the target direction, while attenuate interferences from other directions [157], [83], [40]. It requires accurate direction of arrival (DOA) estimation, and speech and noise covariance matrix estimation. However, conventional DOA algorithms such as generalized cross correlation with phase transform (GCC-PHAT) [80] and multiple signal classification (MUSIC) [134] localize sound sources based on signal energy, and are not robust to noise and reverberation. In addition, spatial covariance matrices are computed based on silence intervals detected by conventional voice activity detectors. Such voice activity detectors make strong stationarity assumptions on noise and usually fail to produce satisfactory performance in real-world conditions where a variety of highly non-stationary intrusions occur.

Multi-talker separation has been an active research area in the past two decades. Earlier research efforts were mainly focused on multi-channel separation, as it was considered a very difficult problem separating multiple speakers based on only spectral information. The major cue exploited in multi-channel multi-talker separation is inter-channel phase patterns, as they naturally form clusters within each frequency for spatially separated directional sources with different time delays to the array [124]. This observation leads to the popular narrow-band and wideband spatial clustering algorithms [70], [102], [131], and

independent component analysis based methods [78]. However, these algorithms only utilize spatial information and do not offer a clear and promising mechanism to leverage spectral information.

In recent years, DNNs [133] have been firmly established as the state-of-the-art approach for single-channel speech enhancement [165], [161]. In this approach, a DNN is typically trained to estimate a real-valued T-F mask to attenuate T-F units dominated by reverberation and noise. Build upon the first DNN study on speech enhancement [165], a subsequent study [55] found that DNN based monaural speech enhancement algorithms led to, for the first time, substantial speech intelligibility improvements for hearing-impaired listeners. Breakthroughs have also been made in single-channel talker-independent speaker separation in [57] and [206], where novel neural network training mechanisms are introduced to solve the label-permutation problem. These studies suggest that magnitude estimation can be substantially improved using deep learning based T-F masking, and point to new directions for single-channel speech enhancement and speaker separation.

These studies also reveal new opportunities for multi-channel processing, since the mask or magnitude estimation provides a powerful means for multi-channel tasks such as acoustic beamforming, sound source localization and post-filtering. If a mask value at a T-F unit is close to one, the phase at that unit is little contaminated, meaning that the inter-channel phase patterns are relatively well manifested. Such T-F units can be utilized to extract reliable spatial information for multi-channel speech enhancement and speaker separation.

The rest of this chapter is organized as follows. Section 1.2 gives a more detailed review of the technical background, defines the objectives of this dissertation, and introduces the roadmap to achieve the objectives. Section 1.3 presents the organization of this dissertation.

1.2. Background, Objectives and Roadmap

Given a P-microphone time-domain mixture signal $\boldsymbol{y}[n] = \left[y_1[n], \dots, y_P[n]\right]^{\mathrm{T}} \in \mathbb{R}^{P\times 1}$ recorded in a reverberant and noisy enclosure, the physical model in the short-time Fourier transform (STFT) domain is formulated as

$$\boldsymbol{Y}(t,f) = \boldsymbol{S}(t,f) + \boldsymbol{N}(t,f) = \boldsymbol{c}(t,f;q)S_q(t,f) + \boldsymbol{N}(t,f), \tag{1.1}$$

where $S_q(t,f) \in \mathbb{C}$ is the complex STFT coefficient of the direct-path signal of the target speaker captured by a reference microphone q at time t and frequency f, and $\boldsymbol{c}(t,f;q) \in \mathbb{C}^{P\times 1}$ is the relative transfer function with the q^{th} element being one. $\boldsymbol{S}(t,f) = \boldsymbol{c}(t,f;q)S_q(t,f)$, $\boldsymbol{N}(t,f)$ and $\boldsymbol{Y}(t,f) \in \mathbb{C}^{P\times 1}$, respectively, represent the STFT vectors of the direct-path signal of a target source (i.e. target speech), non-target signals, and received mixture at a T-F unit. Note that $\boldsymbol{N}$ denotes any non-target signals we aim to remove, such as reverberation, noise or competing speakers.

One popular approach for multi-channel speech enhancement is multi-channel Wiener filtering (MCWF) [40], which computes a linear filter per T-F unit to project the mixture STFT vector to target speech by minimizing the following error function

$$\mathcal{L}\left(\boldsymbol{w}^{(\mathrm{mcwf})}(t,f)\right) = E\left[\left|\boldsymbol{w}^{(\mathrm{mcwf})}(t,f)^H\boldsymbol{Y}(t,f) - S_q(t,f)\right|^2\right], \tag{1.2}$$

where $\boldsymbol{w}^{(\mathrm{mcwf})}(t,f) \in \mathbb{C}^{P\times 1}$ denotes the oracle linear filter, $S_q(t,f) \in \mathbb{C}$ represents the STFT coefficient of the target speech captured by a reference microphone q at time t and frequency f, $(\cdot)^H$ computes conjugate transpose, and $|\cdot|$ extracts magnitude. The expectation operation is performed by assuming that $\boldsymbol{N}(t,f)$ and $\boldsymbol{S}(t,f)$ respectively follow a zero-mean complex Gaussian distribution. The closed-form solution of this optimization problem is

$$\begin{aligned}\boldsymbol{w}^{(\mathrm{mcwf})}(t,f) &= \left(\Phi^{(y)}(t,f)\right)^{-1}\Phi^{(\mathrm{s})}(t,f)\boldsymbol{u}_q \\ &= \left(\Phi^{(\mathrm{s})}(t,f)+\Phi^{(v)}(t,f)\right)^{-1}\Phi^{(\mathrm{s})}(t,f)\boldsymbol{u}_q \\ &= \left(\left|S_q(t,f)\right|^2\boldsymbol{c}(t,f;q)\boldsymbol{c}(t,f;q)^H+\Phi^{(v)}(t,f)\right)^{-1}\Phi^{(s)}(t,f)\boldsymbol{u}_q,\end{aligned} \tag{1.3}$$

where $\Phi^{(s)}(t,f)$, $\Phi^{(v)}(t,f)$, and $\Phi^{(y)}(t,f)=\Phi^{(s)}(t,f)+\Phi^{(v)}(t,f) \in \mathbb{C}^{P\times P}$ respectively denote the speech, noise and mixture spatial covariance matrices, respectively, and $\boldsymbol{u}_q$ is a one-hot vector with the q^{th} element being one. Since the target speaker is directional (i.e. from a specific direction), the speech covariance matrix can be computed as $\Phi^{(s)}(t,f)=\left|S_q(t,f)\right|^2\boldsymbol{c}(t,f;q)\boldsymbol{c}(t,f;q)^H$, where $\left|S_q(t,f)\right|^2 \in \mathbb{R}$ denotes power spectral density.

Under Woodbury matrix identity, Eq. (1.3) can be formulated as a product of a minimum variance distortion-less response (MVDR) beamformer [40] and a Wiener filter based real-valued post-filter

$$\boldsymbol{w}^{(\mathrm{mcwf})}(t,f)=\boldsymbol{w}^{(\mathrm{mvdr})}(t,f)PF(t,f) \tag{1.4}$$

$$\boldsymbol{w}^{(\mathrm{mvdr})}(t,f) = \frac{\Phi^{(v)}(t,f)^{-1}\boldsymbol{c}(t,f;q)}{\boldsymbol{c}(t,f;q)^{H}\Phi^{(v)}(t,f)^{-1}\boldsymbol{c}(t,f;q)} \tag{1.5}$$

$$PF(t,f) = \sqrt{\frac{\boldsymbol{w}^{(\mathrm{mvdr})}(t,f)^{H}\Phi^{(s)}(t,f)\boldsymbol{w}^{(\mathrm{mvdr})}(t,f)}{\boldsymbol{w}^{(\mathrm{mvdr})}(t,f)^{H}\Phi^{(s)}(t,f)\boldsymbol{w}^{(\mathrm{mvdr})}(t,f) + \boldsymbol{w}^{(\mathrm{mvdr})}(t,f)^{H}\Phi^{(v)}(t,f)\boldsymbol{w}^{(\mathrm{mvdr})}(t,f)}} \tag{1.6}$$

The post-filter $PF(t,f)$ can be considered as a Wiener filter based on the energy of beamformed speech and the energy of beamformed noise. The classic MVDR beamforming results from solving the following constrained quadratic optimization problem

$$\begin{gathered}\boldsymbol{w}^{(\mathrm{mvdr})}(t,f) = \mathrm{argmin}_{\boldsymbol{w}(t,f)}\ \boldsymbol{w}(t,f)^{H}\Phi^{(v)}(t,f)\boldsymbol{w}(t,f) \\ \text{subject to}\ \ \boldsymbol{w}(t,f)^{H}\boldsymbol{c}(t,f;q) = 1\end{gathered} \tag{1.7}$$

The idea is to find a linear filter by minimizing noise energy while maintaining the signal from the target direction.

The meaning of Eq. (1.4) is that the MVDR beamformer points a beam towards the target speaker of interest and constructively combines multiple signals into a single one so that the target speech is maintained distortionlessly while non-target signals from other directions are suppressed. The post-filter is necessary to further reduce the residual noise or reverberation in the beamformed signal, as linear beamforming is fundamentally limited when room reverberation is strong, when speech and noise sources are spatially close, or when the number of microphones is small.

In practical systems, all the statistics including $\Phi^{(s)}(t,f)$, $\Phi^{(v)}(t,f)$, $\left|S_q(t,f)\right|^2$ and $\boldsymbol{c}(t,f;q)$ need to be estimated based on the multi-channel mixture input $\boldsymbol{Y}$.

The relative transfer function $\boldsymbol{c}(t,f;q)$, also known as the steering vector, is traditionally computed based on sound localization algorithms such as GCC-PHAT [80], steered-response power with phase transform (SRP-PHAT) [28], and MUSIC [134]. These algorithms are originally designed for narrow-band antenna arrays and are not robust when dealing with wideband speaker localization in noisy and reverberant environments.

The speech and noise covariance matrices, $\Phi^{(s)}(t,f)$ and $\Phi^{(v)}(t,f)$, are conventionally computed using voice activity detection (VAD), where a voice activity detector is utilized to identify noise-only segments for noise covariance matrix computation, or simply using the beginning and ending silence intervals of the mixture signal for estimation [40]. However, VAD algorithms usually assume that environmental noise is stationary, which is unrealistic as real-world noises are typically non-stationary.

The post-filter $PF(t,f)$ is usually computed based on multi-channel signal statistics as in Eq. (1.6), conventional single-channel speech enhancement algorithms [93], [40], or spatial filters computed using phase information [118], [136], [149], [40]. These algorithms usually cannot achieve high-quality noise reduction in reverberant multi-source environments.

Recently, deep learning based T-F masking has substantially advanced monaural speech separation [161]. The key idea is to train a DNN to estimate the ideal binary mask (IBM) [162] or the ideal ratio mask (IRM) [113] for enhancement. Deep learning dramatically improves mask (or magnitude) estimation, and the separated speech exhibits large speech intelligibility and quality improvements over conventional enhancement methods [55], [166].

In this context, we investigate deep learning for microphone array processing and its application to speech separation and localization, and robust ASR. Motivated by the formulation of multi-channel Wiener filtering, this dissertation addresses the following issues in multi-channel processing.

- *Robust speaker localization*. Localization determines the direction of the target speech. Better localization leads to better estimation of the relative transfer function $\boldsymbol{c}(t,f;q)$. Our study performs robust speaker localization by using DNN based T-F masking to identify T-F units dominated by a single source, and only utilizing these T-F units for localization;
- *Acoustic beamforming.* Similar to localization, we utilize DNN based T-F masking to identify T-F units dominated by speech and noise to compute speech and noise covariance matrices, $\Phi^{(s)}(t,f)$ and $\Phi^{(v)}(t,f)$. We also use enhanced speech and noise complex spectra to compute the covariance matrices. Better covariance matrix estimation leads to better beamforming;
- *Post-filtering.* $PF(t,f)$ in Eq. (1.6) is a real-valued mask bounded in the range $[0,1]$. It can be readily improved using deep learning based T-F masking. In addition, based on localization results, we explore spatial features, which can indicate whether the dominant source at each T-F unit is from the estimated direction, and combine them with spectral features to extract the target speech from a particular direction and with specific spectral structure;
- *Phase estimation.* Better phase estimation can lead to better covariance matrix estimation for beamforming and better phase difference estimation for sound localization. It can also help post-filtering to improve the phase produced by linear

beamforming. Our study proposes multiple magnitude based phase reconstruction algorithms. We also investigate complex-domain ratio masking and mapping for phase estimation, following [39], [146], [192];

- *Non-linear time-varying beamforming.* Conventional beamforming techniques are linear and based on second-order statistics. Based on a fixed-geometry array, we investigate DNN based multi-microphone modeling to exploit non-linear spatial information contained in multi-channel inputs for non-linear time-varying beamforming;
- *Multi-channel speech dereverberation, enhancement and speaker separation.* We apply the above ideas to enhance target speech in noisy and reverberant conditions where only a single speaker is assumed active, and also to multi-talker separation tasks where all the speakers need to be separated and enhanced;
- *Single- and multi-channel robust ASR.* A key application of speech enhancement and source separation is to improve modern DNN based ASR systems. This dissertation addresses not only single- but also multi-channel robust ASR in noisy-reverberant conditions, based on deep learning based T-F masking and multi-channel processing.

It is highly desirable to make trained models directly applicable to microphone arrays with various numbers of microphones arranged in diverse layouts. This is especially useful for cloud-based services, where client setup can vary significantly in terms of microphone array configuration. This demand poses challenges to supervised separation, which requires fixed input and output dimensions, and has potentially limited generalization capability to novel array geometries. On the other hand, modern electronic devices such as

Amazon echo and Google Home use a fixed array geometry. It is therefore of interest to develop algorithms for a fixed geometry.

1.3. Dissertation Organization

The rest of this dissertation is organized as follows.

Chapter 2 explores ways of integrating speech enhancement frontends and ASR backends for single-channel robust ASR in noisy-reverberant conditions. We propose a joint training approach that jointly trains frontends, filterbanks and acoustic models. We also apply sequence-discriminative training and unsupervised adaptation to further improve the performance on the CHiME-2 dataset.

Chapter 3 studies robust speaker localization, a key step towards multi-channel speech enhancement and source separation. The idea is to utilize a DNN to identify T-F units dominated by direct sound and only use these T-F units for sound localization. This approach dramatically improves the robustness of conventional cross-correlation, beamforming and subspace based approaches for speaker localization in noisy-reverberant environments.

Chapter 4 integrates complementary spectral and spatial features for deep learning based multi-channel speaker separation in reverberant environments. The main idea is to localize individual speakers so that an enhancement DNN can be trained on spatial as well as spectral features to extract the speaker from an estimated direction and with specific spectral structure. To determine the direction of the speaker of interest, we identify T-F units dominated by that speaker and only use them for direction estimation. The T-F unit level speaker dominance is determined by a two-channel separation network, which

integrates spectral and inter-channel phase patterns at the input feature level. In addition, T-F masking based beamforming is tightly integrated in the system by leveraging the magnitudes and phases produced by beamforming.

Chapter 5 investigates STFT-domain monaural magnitude-based phase reconstruction for talker-independent speaker separation. For a two-source mixture, with the magnitude of each source accurately estimated and under a geometric constraint, the absolute phase difference between each source and the mixture can be uniquely determined. In addition, the source phases at each T-F unit can be confined to only two candidates. In order to pick the correct candidate, we propose three algorithms based on iterative phase reconstruction, group delay estimation, and phase-difference sign prediction. State-of-the-art results are obtained on the publicly available wsj0-2mix and 3mix corpus at the time of publication.

Chapter 6 leverages a complex spectral mapping approach for phase estimation and proposes a target cancellation algorithm for multi-channel speech dereverberation. For single-channel processing, we extend magnitude-domain masking and mapping based dereverberation to complex-domain mapping, where DNNs are trained to predict the real and imaginary (RI) components of the direct-path signal from reverberant (and noisy) ones. For multi-channel processing, we first compute a beamformer to cancel the direct-path signal, and then feed the RI components of the cancelled signal, corresponding to a filtered version of non-target signals, as additional features to perform dereverberation. Our models outperform other state-of-the-art models on the test set of the REVERB challenge in terms of speech dereverberation and recognition performance.

Chapter 7 applies complex spectral mapping to multi-channel speech enhancement, building upon Chapter 6. A novel time-varying beamforming algorithm is proposed to deal

with highly nonstationary environmental noise. State-of-the-art robust ASR performance is obtained on the CHiME-4 corpus.

Chapter 8 combines the RI components of multiple microphones for DNN training. The proposed approach essentially amounts to non-linear time-varying beamforming. It is evaluated on multi-channel dereverberation and robust ASR, and contrasted with single-microphone modeling and conventional dereverberation algorithms.

Chapter 9 concludes this dissertation and discusses future directions.

Chapter 2. Single-Channel Speech Enhancement and Robust ASR

This chapter investigates the integration of deep learning based single-channel speech enhancement and acoustic modeling, which lays a foundation for later multi-channel robust ASR. The key idea is to jointly train enhancement frontends with backend ASR models. This work has been published in Interspeech 2015 [171] and IEEE/ACM T-ASLP in 2016 [172].

2.1. Introduction

DNN-HMM hybrid methods [65] have become the dominant approach in ASR, producing large improvements over conventional GMM-HMM methods. Although a lot of progress has been made in ASR on clean speech, the performance drops sharply in the presence of reverberation, mismatched noises and low SNR conditions. Improving the robustness of ASR systems in such environments remains a challenge.

Although DNN based acoustic models are robust to noisy input with small variations [207], speech separation algorithms are able to significantly improve recognition performance even when DNNs are used for acoustic modeling [25]. Recently, different DNN based speech separation methods, such as T-F masking [167], [168], [165] and

spectral mapping [6], [52], [202], are developed and shown to perform surprisingly well even in highly adverse environments.

When incorporating speech separation into ASR, there are three commonly used strategies. The first one is to conduct acoustic modeling on clean speech, and at run time, a separation frontend is used to enhance noisy speech before recognition [114], [31]. A disadvantage would occur when the separation frontend introduces distortions unseen by the acoustic model trained on clean speech [114]. The second strategy alleviates the distortion problem to some extent by using a separation frontend to enhance both training and test set, and conducts acoustic modeling on the enhanced training set. It may be able to improve the recognition performance since the features may become cleaner after enhancement. The third strategy performs acoustic modeling on noisy speech and at the test stage, noisy or enhanced features are fed to the acoustic model for decoding. The resulting multi-condition training strategy is shown to be very effective [159] but gives unimpressive performance in matched conditions [89]. Clearly, different strategies have their own advantages and disadvantages. Which strategy to adopt highly depends on the situation.

Speech separation and recognition are not two independent tasks and they can clearly benefit from each other. Previous studies [42], [43], [171], proposed to integrate speech separation and acoustic modeling via joint adaptive training. This chapter further develops this approach and proposes various techniques to elevate the performance. The present work makes the following four contributions. First, we concatenate a DNN based speech separation frontend, a trainable mel-filterbank and a DNN based acoustic model together to build a larger and deeper DNN, and jointly adjust the weights in each module via the

back-propagation algorithm. With joint training, the separation frontend and filterbank are able to provide enhanced features expected by the acoustic model. In addition, the linguistic information contained in the acoustic model is allowed to flow back to influence the separation frontend and filterbank. Furthermore, the filterbank can be trained according to the separation frontend and acoustic model [128]. Second, concatenating the separation frontend and acoustic model to form a bigger DNN naturally leads us to sequence-discriminative training applied to the jointly trained DNN for further improvement. This way, at the training stage, the information from language models can be flowed back to influence not only the acoustic model but also the separation frontend by optimizing sequence-discriminative criterion. Third, utterance-level unsupervised adaptation is performed at run time to adapt the jointly trained DNN to potentially mismatched conditions or new speakers. Fourth, we find that adding additional features, which are robust to noise and reverberation, for acoustic modeling significantly improves the robustness.

The proposed sequence-discriminative jointly-trained models trained with additional robust features achieves 10.63% average WER on the test set of the noisy and reverberant CHiME-2 dataset (task-2) [159]. This represented the best result on this dataset at the time of publication.

The rest of this chapter is organized as follows. We describe our joint training approach in Chapter 2.2, followed by experiments and evaluations in Chapter 2.3 and conclusions in Chapter 2.4.

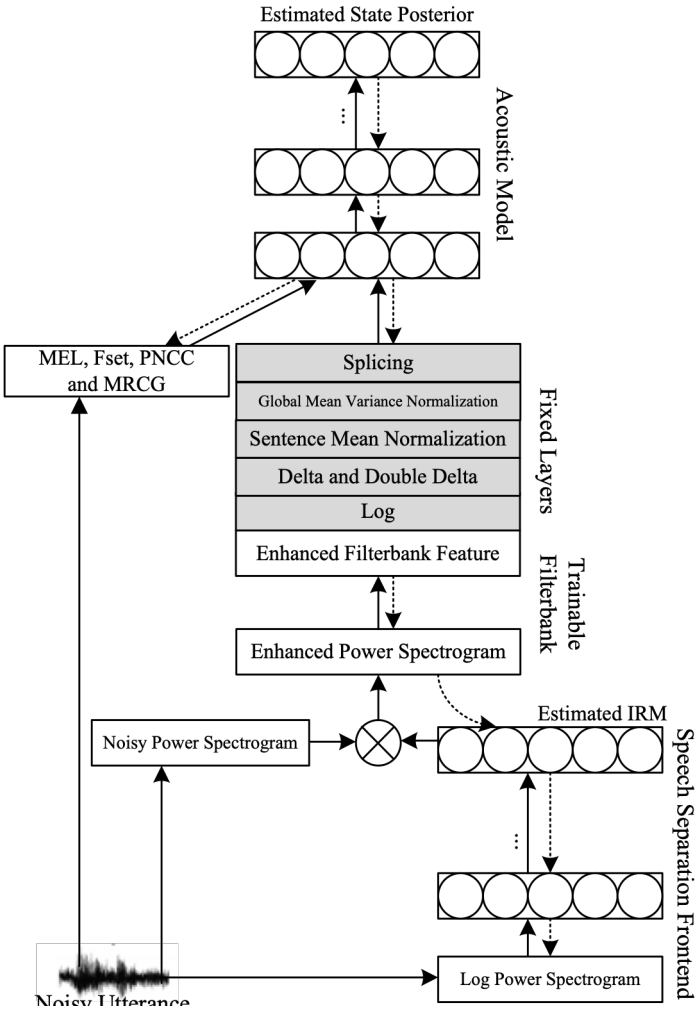


Figure 2-1. Schematic diagram of the proposed joint training framework. The layer shown in gray means that the weights or operations of that layer are fixed. Solid and dotted arrows respectively indicate the directions of forward pass and backward pass. See text for more details.

2.2. System Description

Our system is built in a step-by-step way. We first train a separation frontend and an acoustic model separately, both using DNNs. Then we concatenate the separation frontend, mel-filterbank and acoustic model together to construct a deeper and larger DNN, and

jointly adjust the weights in all modules. After that, we replace the cross-entropy criterion used at the joint training stage with sequence-discriminative criterion for sequence training. Finally, we perform utterance-level unsupervised adaptation at run time. The overall framework of our system is shown in Figure 2-1.

2.2.1. Deep Learning Based T-F Masking

Originated in computational auditory scene analysis (CASA) [163], T-F masking has shown considerable potential for removing additive noise in noisy speech. The key idea is to estimate the IBM [162] that identifies speech dominant and noise dominant T-F units, or the IRM [113], which represents the ratio of speech energy to the sum of speech energy and noise energy within each T-F unit. This framework formulates speech separation as a supervised mask estimation problem. Recently, DNN is employed for mask estimation, and achieves very promising separation performance in both matched and un-matched test conditions [165]. Recent listening tests show that DNN based IBM estimation produces substantial speech intelligibility improvements of noisy utterances for both hearing-impaired and normal-hearing listeners [55]. In addition, different training targets are carefully analyzed recently [166], and it is suggested that the IRM is likely to be a better training target for supervised speech separation. Therefore, we utilize DNNs to estimate the IRM in this study.

The ideal mask can be defined in different T-F representation domains. In line with later joint training, the IRM in this study is defined in the power spectrogram domain [166]

$$M(t,f) = \frac{|S(t,f)|^2}{|S(t,f)|^2 + |N(t,f)|^2}, \tag{2.1}$$

where M is the IRM of a noisy signal created by mixing a noise-free utterance with a noise signal at a specific SNR level, and $|S(t,f)|^2$ and $|N(t,f)|^2$ respectively denote the power spectrograms of the noise-free utterance and the noise signal at time t and frequency f.

At run time, the IRM must be estimated from noisy utterances. We employ a DNN as the learning machine for IRM estimation. The DNN has four hidden layers each with 1,024 rectified linear units (ReLUs) [43]. There are 161 sigmoidal units in the output layer, corresponding to the dimension of each frame in the power spectrogram. Starting from random initialization, the network is trained to minimize the cross-entropy loss function within each T-F unit. The loss function is

$$\mathcal{L}(\widehat{M}) = -\frac{1}{T}\sum_{t,f}\Big[M(t,f)\log\widehat{M}(t,f) + \big(1-M(t,f)\big)\log\Big(1-\widehat{M}(t,f)\Big)\Big], \qquad (2.2)$$

where $\widehat{M}$ is the estimated mask.

The feature used for mask estimation is log-compressed power spectrogram. We splice a large context window of 19 frames centered at the current frame as the input to DNN. The frame length is 20 ms and frame shift 10 ms. For a signal with 16 kHz sampling rate, the input dimension corresponding to one frame is 3,059 (161×19). The log power spectrogram feature is globally mean-variance normalized before splicing.

At run time, we multiply $\widehat{M}$ point-wisely with the power spectrogram of noisy speech to get the enhanced power spectrogram

$$\hat{X} = \widehat{M} \otimes |X|^2, \qquad (2.3)$$

where $\hat{X}$ is the resulting enhanced power spectrogram, $|X|^2$ denotes the noisy power spectrogram, and $\otimes$ represents point-wise matrix multiplication.

2.2.2. Acoustic Modeling

The DNN-HMM hybrid approach is dominant in ASR today. We utilize a DNN with 7 hidden layers each with 2,048 ReLUs for acoustic modeling. The DNN is trained to estimate the posterior probability of each senone state by minimizing the cross-entropy criterion.

Log mel-spectrogram is widely used as the only feature for acoustic modeling. However, mel-spectrogram itself is not robust to noise and reverberation. We incorporate robust features for acoustic modeling as different features contain different and perhaps complementary information for senone state discrimination. We consider a subset of the following features.

- 40-dimensional log mel-spectrogram together with its delta and double deltas (MEL). We perform sentence level mean normalization before splicing an 11-frame context window;
- 256-dimensional multi-resolution cochleagram (MRCG) [17] with its delta and double deltas. This feature is shown to be relatively robust to additive noise for mask estimation;
- 31-dimensional power-normalized cepstral coefficients (PNCC) [74] together with their deltas and double deltas. Sentence level mean normalization is performed before splicing an 11-frame context window. The PNCC feature is found to be robust to reverberation and additive noise;
- 13-dimensional RASTA-PLP [56]. The context window is set to 7;
- 15-dimensional amplitude modulation spectrogram (AMS) [82] extracted from each of 26 channels;

- 31-dimensional narrowband mel-frequency cepstral coefficients (MFCC) with the analysis window of 20 ms. The context window is set to 7;
- 31-dimensional wideband MFCC with the analysis window of 200 ms. The context window size is 7.

The last four features, denoted as Fset, are shown to have complementary power for mask estimation [169]. This study directly uses Fset features for acoustic modeling. With the features mentioned above, the input dimension is 4,026 (40×3×11+256×3+31×3×11 +13×7+15×26+31×7+31×7). They are globally mean-variance normalized before DNN training. To facilitate comparison, we always include MEL for acoustic modeling.

2.2.3. Joint Training

As illustrated in Figure 2-1, the key idea of joint training is to concatenate an acoustic model DNN and a speech separation DNN to form a larger and deeper neural network, and jointly adjust the weights in all modules. The link for concatenating the separation frontend and the acoustic model is a trainable filterbank layer and a set of layers with fixed operations, which represent the extraction of the enhanced MEL features (with delta and double deltas and an 11-frame context window) (see also [115], [116], [171]). More specifically, after obtaining the estimated IRM from the separation frontend based on the log power spectrogram of a noisy utterance, we multiply it point-wisely with the noisy power spectrogram to get the enhanced power spectrogram. The enhanced power spectrogram is then fed into the trainable filterbank layer to get the enhanced filterbank feature. Afterwards, we compress it logarithmically, add delta and double deltas, perform sentence-level mean normalization, conduct global mean-variance normalization, and splice 11 frames to yield the enhanced MEL features. The enhanced MEL features, together

with other robust features, are finally passed to the acoustic model to estimate state posterior probabilities. The joint training framework can be performed in a single neural network because the point-wise multiplication, filtering, sentence- and global-level normalization, adding delta and double deltas are all linear transformations. Therefore, we can flow the error signal from the acoustic model back to the filterbank layer and the separation frontend, and jointly train all modules using back-propagation.

A similar frontend and backend joint training approach was presented by Gao *et al.* [41], where feature mapping is employed as the frontend. It has been suggested that masking is likely a better approach than mapping for speech separation [166]. In addition, the output dimension of their frontend is equal to the input dimension, which consists of many consecutive frames and is large. In contrast, we obtain enhancement results per single frame. Furthermore, their frontend obtains enhanced MEL features by direct mapping instead of using a trainable filterbank layer and fixed layers to transform the enhanced power spectrogram.

Parameter initialization is critical before joint training. Here we use the weights in a separately trained acoustic model and a separately trained separation frontend to initialize the corresponding parts of the DNN before joint training. Following [128], we initialize the parameters in the trainable filterbank (FB) layer using

$$W^{FB} = \exp(W^*), \tag{2.4}$$

where W^* is initialized to

$$W^* = \log\left(\max\left(Mel\_FB, eps\right)\right) \tag{2.5}$$

Here Mel_FB denotes the standard 40-dimensional mel-filterbank and eps is a small constant (10^{-3} in this study). With Eq. (2.4), every time W^* is updated, all the parameters in the filterbank are ensured to be non-negative.

The whole network is trained to minimize the cross-entropy criterion from the acoustic model alone. We tried to put a weight between the loss of the acoustic model and the loss of the separation frontend. However, no clear improvement on the ASR performance was observed. The sentence-level mean of each utterance and global mean and variance are updated at the beginning of each epoch in the forward pass.

2.2.4. Sequence-Discriminative Training

The previous sections describe how the DNN-based acoustic models are trained to minimize the cross-entropy criterion at the frame level. As ASR is a sequence classification problem, it is sensible to optimize sequence-discriminative criterion to better capture the essence of this problem. It is widely known that sequence training is helpful for GMM-HMM systems. In recent studies, sequence training is also found to be useful for DNN-HMM hybrid systems [158], [116]. Here, we investigate the effectiveness of sequence training criterion on the joint training system. We replace the frame-wise cross-entropy criterion with the state-level minimum Bayes risk (sMBR) [75] and back-propagate the error signal from this criterion to adjust the weights in the acoustic model, filterbank and separation frontend. This method is expected to improve recognition performance. We believe that this method may also benefit mask estimation since the error signal from the sequence training criterion contains information from language models, which is rarely exploited in speech separation research.

2.2.5. Unsupervised Adaptation

Adaptation is commonly performed on well-trained acoustic models to compensate the differences between training and test conditions. It can be supervised or unsupervised, depending on whether the labels of adaptation data are available. Many adaptation methods have been proposed for DNN based acoustic models, such as linear transformation [135], [114], conservative training [208], and subspace based methods [129]. In [105], it is suggested that the linear input network (LIN) and linear hidden network based approaches are better than linear output network, factorization and KL-divergence based adaptation.

We perform unsupervised adaptation to our jointly trained acoustic models following the LIN approach. At run time, given a single test utterance, we first use the un-adapted jointly-trained sequence-discriminative model to generate initial decoding results. The first-pass decoded state sequence is then used as the labels for learning a linear transformation of the input features of the separation frontend by minimizing the cross-entropy criterion calculated from the acoustic model, with all the other parameters fixed. The linear transformation is defined as follows:

$$\hat{x}_{t,f} = w_f x_{t,f} + b_f, \tag{2.6}$$

where $x_{t,f}$ denotes the globally mean-variance normalized log power spectrogram, corresponding to the un-adapted input of the separation frontend, $\hat{x}_{t,f}$ denotes the adapted features, and w_f and b_f are the parameters to be learned. For a test utterance, the number of parameters to learn is 322 (161+161), which is approximately in the same range of the number of frames in the test utterance.

For each utterance, the adaptation process is run for 20 epochs with a mini-batch size equal to the length of the utterance. We simply adopt the learned parameters at the last epoch due to the lack of a development set. After obtaining all the linear transformation for each test utterance, we re-generate the likelihood and run a second-pass decoding to obtain the final results.

A similar adaptation method was proposed in [114]. One key difference is that we perform adaptation on the input of the separation frontend rather than on the output of the separation frontend. We think that our strategy is better since, if we perform adaptation on the input of the separation frontend, the enhancement results would be changed in a highly non-linear way rather than in a simple linear fashion.

This unsupervised adaptation technique with the learned linear transformation can also adapt a well-trained separation frontend to new test environments to some extent.

2.3. Experimental Setup

We evaluate the proposed algorithms on the reverberant and noisy CHiME-2 dataset (task-2) [159]. The CHiME-2 dataset is created by first convolving clean utterances in the WSJ0-5k dataset with time-varying binaural room impulse responses (BRIRs) and then mixing with reverberant noises at six SNR levels equally spaced from -6 to 9 dB. The BRIRs and reverberant noises are recorded with the same microphone and living room setup. The recorded noises contain major noise sources in a typical kitchen or living room, such as competing speakers, electronic devices, footsteps, laughter, and distant noises. The multi-conditional training set (*si_tr_s*) contains 7,138 utterances (~14.5h), the development set (*si_dt_05*) contains 409 utterances at each SNR level (~4.5h), and the test set (*si_et_05*)

contains 330 utterances at each SNR level (~4h). The CHiME-2 dataset provides reverberant noises, and reverberant noise-free utterances corresponding to the multi-conditional training set. With the noises, clean speech, reverberant noise-free utterances, and noisy-reverberant utterances available, we can readily evaluate the recognition performance together with speech separation performance of our system.

Our system is monaural in nature. We simply average the signals from the left and right channel before extracting features. This technique shows better performance than only using one of these two channels. A GMM-HMM system is built using the Kaldi toolkit [121] on the clean utterances in the WSJ0-5k to get the senone state for each frame of the corresponding noisy-reverberant utterances. Following the common pipeline in the Kaldi toolkit, the GMM-HMM system is first built using the MFCC feature. Then we concatenate 13-dimensional MFCC feature within a 7-frame context window, and utilize linear discriminant analysis (LDA) to compress the concatenated feature to 40 dimensions. After that, we de-correlate it via maximum likelihood linear transform (MLLT) and use feature-space maximum likelihood linear regression (fMLLR) to reduce speaker variance, which is estimated by speaker adaptive training. The resulting cross-word tied-state tri-phone GMM-HMM system contains 1,965 senone states. The initial clean alignments are obtained by performing forced alignment on the clean utterances. To refine the initial clean alignments, we further train a DNN-based acoustic model using the MEL features of the clean utterances, and re-generate clean alignments. Such clean alignments are used as the labels for training all the acoustic models in this study. Note that the DNN-HMM hybrid system built on the clean utterances is a powerful recognizer. It achieves 2.15% word error rates (WER) on the clean test set of the WSJ0-5k dataset. Therefore, we believe that these

high-quality labels can guide the DNN-based acoustic model to perform well on discriminating different senone states even when the input features are very noisy and the input SNR very low. We use the CMU pronunciation dictionary and the official 5k close-vocabulary tri-gram language model in our experiments. This language model is used for decoding at run time and generating the lattices of the training utterances at the sequence training stage.

The training data for mask estimation is obtained from parallel noisy-reverberant and reverberant noise-free data. The mixed noise signals can be obtained by direct subtraction. With these datasets, we train a separation frontend to remove additive noise in noisy-reverberant utterances. The noisy-reverberant dataset, i.e. the multi-conditional training data, is used for both mask estimation and acoustic modeling.

Our experiments are done in an incremental manner. We first build our acoustic models using feature subsets selected according to the performance on the development set. Then we jointly train the acoustic models with the separation frontend. Afterwards, we perform sequence training on the jointly trained DNN. Finally, we perform unsupervised adaptation to the sequence-discriminative jointly-trained DNN at run time.

2.3.1. Expanded Feature Set for Acoustic Modeling

We first report the results of incorporating robust features for acoustic modeling. In this experiment, no speech enhancement or separation is performed. We simply train acoustic models multi-conditionally by adding robust features and do not tune the network structure or training recipes for each feature set. To push up the baselines, we perform sequence training on the multi-conditionally trained acoustic models, followed by run-time unsupervised adaptation. The WER results are presented in Table 2-1.

Table 2-1. Performance (%WER) using multi-condition training with robust features for acoustic modeling.

Features for Acoustic Modeling	Dev. Set	Test Set						
	Average	-6dB	-3dB	0dB	3dB	6dB	9dB	Average
MEL	19.40	26.77	20.49	16.14	12.80	10.67	10.11	16.16
+sMBR	17.24	23.87	17.35	13.64	11.30	9.10	8.28	13.92
+adaptation	**16.81**	**22.64**	**16.85**	**12.78**	**10.44**	**8.69**	**7.79**	**13.20**
MEL+PNCC	18.54	25.13	18.57	14.94	11.73	9.51	8.57	14.74
+sMBR	16.52	23.22	16.59	12.46	10.52	8.24	7.49	13.09
+adaptation	16.10	22.03	16.33	12.22	10.29	7.66	7.36	12.65
MEL+PNCC+MRCG	17.99	23.33	17.92	14.20	11.36	8.95	8.05	13.97
+sMBR	15.97	22.01	15.62	12.18	10.59	8.18	7.12	12.62
+adaptation	15.57	21.17	15.21	11.83	10.55	7.77	6.80	12.22
MEL+PNCC+MRCG+Fset	17.93	23.09	17.17	13.32	10.41	8.71	8.07	13.46
+sMBR	15.63	21.17	14.96	12.24	9.83	7.68	7.14	12.17
+adaptation	**15.48**	**20.51**	**14.68**	**11.77**	**9.70**	**7.49**	**7.02**	**11.86**

If we only train our acoustic models using the cross-entropy criterion, with the commonly used MEL features alone, we obtain 16.16% average WER on the test set. Note that if we just use the default DNN code for the CHiME-2 dataset in the Kaldi toolkit, we only obtain 17.49% average WER on the test set. This is consistent with the results obtained in [53]. The major differences are that we use ReLUs, dropout and Adagrad for training, while the default DNN code uses sigmoidal units, pre-training and stochastic gradient descent. By adding PNCC, the average WER can be reduced to 14.74%. After appending MRCG, the WER is brought down to 13.97%. The performance is further pushed to 13.46% average WER after we add Fset. Note that this result is already better than our previous best result [171] using the same set of features on this dataset, mainly because better clean alignments are generated using the Kaldi toolkit.

We then apply sequence training to the multi-conditionally trained acoustic models. We observe that sequence training leads to large improvement for all the input features, and the relative improvement becomes smaller if more features are used for acoustic modeling.

Finally, we apply utterance-level unsupervised adaptation to the sequence-discriminative acoustic models. Similar to Chapter 2.2.5, given a test utterance, we first decode it to obtain a hypothesized state sequence, from which we learn a linear transformation of the input features. To reduce the number of parameters to learn and make a fair comparison with later experiments, we only learn a linear transformation for the MEL features. Learning linear transformations for other features may decrease the performance, simply because too many parameters are learned. Thus, the total number of parameters to be learned is 240 ($40\times3+40\times3$) for each test utterance. From Table 2-1, we see that unsupervised adaptation leads to consistent improvement, while the relative improvement for acoustic models with more features becomes smaller as well.

Compared with only using the MEL features, adding all the extra robust features for acoustic modeling reduces the average WER by 2.7% (16.16% to 13.46%), 1.75% (13.92% to 12.17%), and 1.34% (13.20% to 11.86%) without sequence training or adaptation, with sequence training but no adaptation, and with sequence training and adaptation, respectively. These considerable improvements occur probably because features are extracted from different domains using different filterbanks, compression operations and environmental compensations, and therefore they likely complement each other for acoustic modeling on multi-conditional data. This suggests that relying on the DNN to learn optimal non-linear features from relatively raw input, e.g. the MEL features, may not be the optimal strategy for robust ASR. Combining the feature learning ability of DNNs and domain knowledge may be a better way for improving the robustness of ASR systems.

As shown in Table 2-1, the average WER on the development set keeps decreasing as we add more and more features. Therefore, in the following experiments, we add the

Table 2-2. Performance (%WER) comparison of proposed approach without extra robust features

Approaches	Acoustic Model	dev. set	test set						
		Average	-6dB	-3dB	0dB	3dB	6dB	9dB	Average
Plug-and-Play	MEL	18.22	23.58	18.53	14.85	12.42	9.68	9.56	14.77
	+sMBR	16.63	22.72	16.12	13.81	10.84	8.61	8.39	13.42
	+adaptation	16.05	21.18	15.82	12.16	10.54	8.14	7.88	12.62
Re-training	Enhanced MEL	18.67	25.85	19.20	15.93	12.52	9.96	9.21	15.45
	+sMBR	17.08	24.38	17.19	13.66	11.10	8.69	8.20	13.87
	+adaptation	16.59	23.54	16.40	12.76	10.55	8.37	7.66	13.21
Re-training	Enhanced MEL + MEL	18.31	25.31	18.83	15.69	11.94	9.23	8.89	14.98
	+sMBR	16.50	24.10	16.68	14.18	10.42	8.63	7.88	13.65
	+adaptation	16.07	22.70	16.14	13.32	9.96	7.88	7.40	12.9
Jointly training frontend, AM and filterbank	Jointly enhanced MEL	17.63	22.55	17.65	14.42	11.36	9.23	8.74	13.99
	+sMBR	15.28	20.44	14.66	12.39	9.81	7.73	7.38	12.07
	+adaptation	**14.56**	**18.72**	**13.77**	**11.36**	**9.32**	**7.32**	**6.86**	**11.23**
Jointly training frontend and AM	Jointly enhanced MEL	17.62	23.15	17.69	14.72	11.38	9.30	9.15	14.23
	+sMBR	15.30	20.61	14.89	12.48	9.81	7.85	7.49	12.19
	+adaptation	14.60	19.13	13.67	11.40	9.19	7.51	7.08	11.33
Directly training a large DNN	Log power spectrogram + MEL	19.06	24.88	18.91	15.15	12.57	10.44	9.25	15.2

PNCC, MRCG and Fset features for acoustic modeling. Note that we do not perform any kind of enhancement on these extra features since they are considered to be inherently robust. To facilitate comparisons, we also report the results based on the MEL features alone.

2.3.2. Plug-and-Play and Re-Training Approaches

Before presenting the results of the joint training approach, we explore two alternative strategies when incorporating speech separation into ASR systems.

The first strategy, denoted as *plug-and-play*, is to train our acoustic models using the MEL features alone or the MEL+PNCC+MRCG+Fset features. At run time, we use the trained separation frontend to get the enhanced power spectrogram which is then passed to the mel-filterbank to get the enhanced MEL features. Finally, together with other robust features, the enhanced MEL features are passed to the acoustic model for decoding. As

shown in the first entry of Table 2-2, if we only use the MEL features for acoustic modeling, the frontend leads to 1.39% (16.16% to 14.77%), 0.5% (13.92% to 13.42%), and 0.58% (13.20% to 12.62%) absolute improvement without sequence training or adaptation, with sequence training but no adaptation, and with sequence training and adaptation, respectively. We can see that the relative improvement of using our frontend becomes much smaller if the acoustic model has been sequence-trained. Note that for unsupervised adaptation, we learn a linear transformation of the enhanced MEL features. The first-pass decoding results for adaptation are obtained by applying the plug-and-play approach to the sequence-discriminative acoustic model. Again, the number of parameters to be learned is 240 (40×3+40×3). Performing unsupervised adaptation on the enhanced MEL features leads to 0.8% (13.42% to 12.62%) average WER reduction. Similar observations can be found in the first entry of Table 2-3, in which we use the MEL+PNCC+MRCG+Fset features for acoustic modeling.

The second alternative, denoted as *re-training*, is to train our acoustic models using the enhanced MEL features alone or the enhanced MEL+PNCC+MRCG+Fset features. At run time, after obtaining the enhanced MEL features, together with other robust features, we feed all of them to the acoustic model for decoding. Note that, again, Fset, MRCG and PNCC are directly extracted from the original noisy-reverberant utterances. The results are shown in the second entries of Table 2-2 and Table 2-3, respectively. Motivated by deep stacking [27], [191], the unenhanced MEL features are additionally incorporated for acoustic modeling. The results are reported in the third entry of Table 2-2 and Table 2-3, without and with extra robust features, respectively. We can see that adding the unenhanced MEL features for acoustic modeling brings some gains for the re-training approach.

Table 2-3. Performance (%WER) comparison of proposed approach with extra robust features.

Approaches	Acoustic Model	dev. set	test set						
		Average	-6dB	-3dB	0dB	3dB	6dB	9dB	Average
Plug-and-Play	MEL+PNCC+MRCG+Fset	16.90	21.32	15.26	12.52	10.11	7.83	7.44	12.41
	+sMBR	15.34	20.04	13.64	11.56	9.56	7.64	7.08	11.59
	+adaptation	14.98	19.65	13.49	11.32	9.30	7.34	6.91	11.34
Re-training	Enhanced MEL+PNCC+MRCG+Fset	16.98	23.20	16.72	12.89	10.37	8.24	7.57	13.17
	+sMBR	15.80	22.96	16.16	12.55	9.55	7.86	7.34	12.74
	+adaptation	15.28	22.04	15.49	12.16	9.21	7.66	7.12	12.28
Re-training	Enhanced Mel+MEL+PNCC+MRCG+Fset	17.08	22.60	16.53	12.74	10.14	8.24	7.38	12.94
	+sMBR	15.52	22.87	15.58	12.61	9.40	7.70	6.76	12.49
	+adaptation	14.97	20.85	14.68	12.07	9.06	7.42	6.61	11.78
Jointly training frontend, AM and filterbank	Jointly enhanced MEL+PNCC+MRCG+Fset	15.58	20.23	14.40	11.73	9.73	7.38	7.45	11.82
	+sMBR	14.33	19.20	13.30	10.74	8.76	6.89	6.84	10.96
	+adaptation	**13.81**	**18.23**	**13.02**	**10.39**	**8.67**	**6.86**	**6.61**	**10.63**

Comparing the results from plug-and-play and re-training, we find that the former strategy typically scores higher. One possible reason is that, when re-training is used, the separation frontend significantly reduces the variations seen by the acoustic model at the training stage [137]. In addition, the distortion it introduces for the training utterances may be different from that for the test utterances. Another possible explanation is related to overfitting. Since the separation frontend is also trained on the multi-conditional training data, we can reasonably assume that the separation frontend performs better on the training set than on the development and test set. Therefore, if the enhanced training data is subsequently used to re-train the acoustic models, overfitting would likely happen. This is exactly what we encountered in our experiments. For the re-training approach, the loss of the acoustic model on the development set is much better than that of the plug-and-play or the direct multi-condition training approach; however it gives us worse performance after decoding.

2.3.3. Joint Training

Considering that more variations are seen by the acoustic models trained on noisy-reverberant utterances and the plug-and-play approach normally gets better performance on the development set as shown in Table 2-2 and Table 2-3, we use the parameters in the acoustic models from this approach, together with the separation frontend, to initialize the corresponding parameters in the joint-training DNN, and then perform joint training. When joint training is done, sMBR training and run-time adaptation are conducted. Note that for the run-time adaptation, we learn a linear transformation of the input of the separation frontend. The number of parameters to be learn is 322 (161+161) for each utterance.

As reported in Table 2-2, after joint training, the performance can be improved from 14.77% to 13.99% average WER. After sMBR training, the performance is improved to 12.07%. The performance is further pushed up to 11.23% after run-time unsupervised adaption, which is helpful especially in low SNR conditions. For example, when the input SNR is -6 dB, the WER is reduced from 20.44% to 18.72%.

If we do not use extra robust features for acoustic modeling, compared with plug-and-play, we reduce the average WER by absolute 0.78% or relative 5.3% (14.77% to 13.99%) if only the cross-entropy criterion is used for joint training. The performance gap is enlarged to absolute 1.35% or relative 10.06% (13.42% to 12.07%) after sequence training is applied. If we further perform run-time unsupervised adaptation, the performance difference is further increased to absolute 1.39% or relative 11.01% (12.62% to 11.23%). Interestingly, the relative improvement becomes larger after sequence training and unsupervised adaptation are applied to the joint-training DNN. This trend can also be observed by comparing the first entry with the fourth one in Table 2-3, where more features

are used for acoustic modeling. This is desirable since, in joint modeling, the noise compensation module can be seamlessly combined with other ASR techniques, such as sequence training and adaptation, to obtain further improvement.

As presented in the fourth and fifth entries of Table 2-2, co-adapting the filterbank with the separation frontend and acoustic model leads to slightly better results. If the parameters in the filterbank are co-adapted, the performance is 0.24% (14.23% to 13.99%) average WER better after joint training, 0.12% (12.19% to 12.07%) better after sMBR training, and 0.1% (11.33% to 11.23%) better after run-time adaptation.

These results clearly demonstrate the effectiveness of joint training. We think that it is due to the reduction of the distortion problem and the linguistic information back-propagated from the acoustic model to the separation frontend. In addition, the separation frontend used in this study treats all the frames and T-F units equally important, without considering the underlying linguistic information that is critical for senone states discrimination. In contrast, with joint modeling, the separation frontend can be informed by the acoustic model to produce more discriminative enhancement results.

The best performance we obtained on the test set is 11.23% average WER if no extra robust features are used. With extra robust features, the performance is further improved to 10.63%. With more sophisticated training and adaptation techniques, the effectiveness of extra features is reduced. This would be welcome as using a small number of features, such as log mel-spectrogram, is favored in industry. On the other hand, incorporating more robust features for acoustic modeling is a simple and effective technique towards improved robustness of ASR systems.

Table 2-4. Performance (%WER) comparison of proposed approach with other studies.

Study	dev. set	test set						
	Average	-6dB	-3dB	0dB	3dB	6dB	9dB	Average
Weng *et al*. [190]	-	38.11	29.07	22.98	17.92	14.96	13.60	22.77
Chen *et al.*[19]	20.11	-	-	-	-	-	-	16.04
Narayanan-Wang [116]	-	25.1	19.2	15.1	12.8	10.5	9.5	15.4
Weninger *et al*. [191]	17.87	23.48	17.02	13.71	10.72	8.95	8.67	13.76
sMBR+joint training+multi-stream +run-time adaptation (proposed)	13.81	18.23	13.02	10.39	8.67	6.86	6.61	10.63

It might be argued that the joint training approach just performs acoustic modeling multi-conditionally by training a very deep and large DNN on a combination of features. To address this possibility, we train a DNN with 12 (4+1+7) hidden layers, each with 1,600 ReLUs, on the combination of the log power spectrogram and MEL features (without robust features) using multi-condition training directly. Note that the number of parameters in this new DNN is almost the same as that in the joint-training DNN. The performance, shown in the last entry of Table 2-2, is much worse than that of joint training. This is likely because the joint training approach has better network architecture and better parameter initialization.

2.3.4. Comparison with Other Studies

In Table 2-4, we list the results of several other studies that report competitive results on the same dataset. All of them use the DNN-HMM hybrid approach and clean alignments from clean utterances as the labels to train their acoustic models. The system described in [190] employs an RNN to perform acoustic modeling on the noisy-reverberant training data and does not use any speech enhancement or separation. Chen *et al.* [19] utilize LSTM for both speech separation and acoustic modeling. Their ASR systems follow the re-

training approach, and an iterative strategy using alignment information from their ASR system is proposed to improve speech separation and recognition simultaneously. Weninger *et al*. [191] build their frontend by training an RNN with the LSTM activation function to predict a phase-sensitive spectrum approximation objective function. They also use re-training and additional alignment information from ASR systems to boost the performance of speech separation. Their DNN based acoustic models are built in a way similar to the standard recipes in the Kaldi toolkit. Both enhanced and unenhanced log mel-filterbank features without delta components are utilized for acoustic modeling, and no extra robust features are used in their study. Han *et al.* [53] use a spectral mapping based separation frontend to enhance both the training and test set first, and perform acoustic modeling on the enhanced training set using the standard DNN training recipes in the Kaldi toolkit. Their overall WER is 15.6%, which is slightly worse than obtained by Narayanan and Wang [116]. To our knowledge, the results by Weninger *et al*. [191] are the best on the CHiME-2 dataset reported in the literature. As shown in Table 2-4, we have now pushed the performance to 10.63% average WER. This represents a 22.75% relative error reduction over [191], and the best result at the time of publication.

2.4. Conclusion

Speech separation and recognition are two closely related problems. In this study, a joint training strategy is presented to integrate speech separation and acoustic modeling at the training stage. By further applying sequence training and run-time adaptation, the performance advantage of the joint modeling approach becomes even larger. Still, speech separation is done in a bottom-up fashion at the test stage. How to leverage top-down

information, such as the knowledge from language models, to help speech separation at the test stage is an interesting direction for future research. We think that the joint modeling approach presented in this paper could be an important step towards this goal, because language models are about the relations among words, or in a wider sense, among phonemes or states, while speech separation is commonly done in the T-F domain or at the signal level [173]. There is clearly a gap between them. The joint modeling approach utilizes acoustic models to bridge these two modules so that the information can be potentially flowed back and forth.

Chapter 3. Robust Speaker Localization

This chapter studies robust speaker localization in noisy and reverberant conditions, which serves as a key step for multi-channel speech enhancement and source separation. The main idea is to identify T-F units dominated by direct sound and only use these T-F units for speaker localization. This work has been published in Interspeech 2018 [174] and IEEE/ACM T-ASLP in 2019 [175].

3.1. Introduction

Robust speaker localization has many applications in real-world tasks. The ability to localize a speaker in daily environments is important for a voice-based interface such as Amazon Echo. Localization is also widely used in beamforming for speech separation or enhancement [40].

Conventionally, GCC-PHAT [80] (or SRP-PHAT [28]) and MUSIC [134] are the two most popular algorithms for sound source localization. However, their speaker localization performance is unsatisfactory in noisy and reverberant environments; in such environments, the summation of GCC coefficients exhibits spurious peaks and the noise subspace constructed in the MUSIC algorithm does not correspond to the true noise subspace.

To improve the robustness, frequency-dependent SNR weighting is designed to emphasize frequencies with higher SNR for the GCC-PHAT algorithm. SNR can be computed in various ways, such as rule-based methods [155] and VAD based algorithms [127]. T-F unit level SNR based on minima controlled recursive averaging or inter-channel coherence has also been applied to emphasize T-F units with higher SNR or coherence [9], [124], [156]. However, these algorithms typically assume stationary noise, which is an unrealistic assumption in real-world acoustic environments.

While it is difficult to perform localization in noisy and reverberant environments, with two ears the human auditory system shows a remarkable capacity at localizing sound sources. Psychoacoustic evidence suggests that sound localization largely depends on sound separation [12], [54], [163], which operates according to auditory scene analysis principles [12]. Motivated by perceptual organization, we approach robust speaker localization from the angle of monaural speech separation.

It is well-known that, even for a severely corrupted utterance, there are still many T-F units dominated by target speech [163]. As analyzed [45], [106], [156], [196], [211], [216], these T-F units carry relatively clean phase and may be sufficient for speaker localization. Motivated by this observation, our approach aims at identifying speech dominant T-F units at each microphone channel and only using such T-F units for multi-channel localization. A profound consequence of this new approach is that deep learning can be brought to bear on T-F unit level classification or regression for robust localization.

In this context, we perform robust DOA estimation by utilizing deep learning based T-F masking. We make three contributions. First, DNN estimated masks are utilized to improve the robustness of conventional cross-correlation, beamforming and subspace

based algorithms [28] for DOA estimation in environments with strong noise and reverberation, following previous research along similar directions [119], [201]. A key ingredient, we believe, is balancing the contributions of individual frequency bands for the DOA estimation of broadband speech signals. Second, we find that using the IRM and its variants, which consider direct sound as the target signal, leads to high localization accuracy, suggesting that such training targets are very effective for robust speaker localization (see also [119]). Third, we show that the trained model is versatile in application to sensor arrays with diverse geometries and with various numbers of microphones.

The rest of this chapter is organized as follows. The proposed algorithms are presented in Chapter 3.2. Experimental setup and evaluation results are reported in Chapter 3.3, and 3.4. Chapter 3.5 concludes this paper.

3.2. System Description

We start with a review of the classic GCC-PHAT algorithm, which motivates our algorithm design. The following three sections propose three localization algorithms based on mask-weighted GCC-PHAT, mask-weighted steered-response SNR, and steering vectors. They respectively represent cross-correlation , beamforming and subspace based approaches for localization. Deep learning based T-F masking for the purpose of speaker localization is described in the last subsection.

3.2.1. GCC-PHAT

Suppose that there is only one target speaker, the physical model for a pair of signals in noisy and reverberant environments under the narrowband approximation assumption can be formulated as

$$\boldsymbol{Y}(t,f) = \boldsymbol{c}(f;q)S_q(t,f) + \boldsymbol{H}(t,f) + \boldsymbol{N}(t,f), \tag{3.1}$$

where $S(t,f)$ is the STFT value of the direct-path signal of the target speaker captured by a reference microphone q at time t and frequency f, and $\boldsymbol{c}(f;q)$ is the relative transfer function. $\boldsymbol{c}(f;q)S_q(t,f)$, $\boldsymbol{H}(t,f)$, $\boldsymbol{N}(t,f)$, and $\boldsymbol{Y}(t,f)$ respectively represent the STFT vectors of the direct signal, its reverberation, reverberated noise, and received mixture. By designating the first microphone as the reference, the relative transfer function $\boldsymbol{c}(f;q)$ in the two-microphone case can be described as

$$\boldsymbol{c}(f;q) = \left[1, A(f)e^{-j2\pi\frac{f}{D}f_s\tau^*}\right]^T, \tag{3.2}$$

where τ^* denotes the time difference of arrival (TDOA) between the two signals in seconds, $A(f)$ is a real-valued relative gain, j is the imaginary unit, f_s is the sampling rate in Hz, D is the number of discrete Fourier transform (DFT) frequencies, and $[\cdot]^T$ stands for transpose. The range of f is from 0 to $D/2$.

The classical GCC-PHAT algorithm [80], [28] estimates the time delay of a pair of microphones p and q by computing their generalized cross-correlation coefficients with a weighting mechanism based on phase transform

$$GCC_{p,q}(t,f,k) = \text{Real}\left\{\frac{Y_p(t,f)Y_q(t,f)^H}{\left|Y_p(t,f)\right|\left|Y_q(t,f)^H\right|}e^{-j2\pi\frac{f}{D}f_s\tau_{p,q}(k)}\right\} \tag{3.3}$$

$$= \cos\Big(\angle Y_p(t,f) - \angle Y_q(t,f) - 2\pi \frac{f}{D} f_s \tau_{p,q}(k) \Big),$$

where Real$\{\cdot\}$ extracts real component and $\angle(\cdot)$ extracts phase. $\tau_{p,q}(k) = (d_{kq} - d_{kp})/c_s$ denotes the time delay of a candidate direction or location k, where c_s is the speed of sound in the air, and d_{kq} and d_{kp} respectively represent the distance between the hypothesized sound source to microphone p and q. Assuming that the target speaker is still within a single utterance, the GCC coefficients are then summated and the time delay producing the largest summation represents the delay estimate.

Intuitively, this algorithm first aligns two microphone signals using a candidate time delay τ and then computes their cosine distance at each T-F unit pair. If the cosine distance is close to one, it means that the candidate time delay is close to the true time delay at that T-F unit. The summation functions as a voting mechanism to combine the observations at all the unit pairs. Since each GCC coefficient is naturally bounded between -1 and 1, each T-F unit pair has an equal contribution to the summation. We emphasize that PHAT weighting [14], [210], i.e. the magnitude normalization term in Eq. (3.3), is essential, as the energy of human speech is mostly concentrated in lower frequency bands. If the magnitude normalization is not performed, lower frequency components would have much larger GCC coefficients and dominate the summation, making it less sharp. In addition, the scales of the two signals are usually different in near-field or binaural cases. It is hence beneficial to remove the influence of different energy levels.

We emphasize that summation over frequencies is very important for broadband speech signals. Because of spatial aliasing [40], the cross-correlation function at high frequencies

is typically periodic, containing multiple peaks. It is important to summate over all the frequencies to sharpen the peak corresponding to the true timed delay [163].

Although GCC-PHAT performs well in environments with low to moderate reverberation, it is susceptible to strong reverberation and noise. To see this, suppose that there is a strong directional noise source. There would be many T-F units dominated by the noise source. In this case, the noise source would exhibit the highest peak in the summated GCC coefficients. Similarly, diffuse noise and reverberation would broaden GCC peaks and corrupt TDOA estimation.

3.2.2. Mask-Weighted GCC-PAHAT

The time delay information is contained in the direct-path signal $\boldsymbol{c}(f;q)S_q(t,f)$. Including the GCC coefficients of any T-F unit pairs dominated by noise or reverberation in the summation would weaken localization performance. To improve robustness, we multiply the GCC coefficients for a pair of microphones and a masking-based weighting term following [156], [45]

$$MGCC_{p,q}(t,f,k) = \widehat{M}_{p,q}^{(s)}(t,f)GCC_{p,q}\left(t,f,\tau_{p,q}(k)\right), \tag{3.4}$$

where $\widehat{M}_{p,q}^{(s)}(t,f)$ represents the importance of the T-F unit pair for TDOA estimation (superscript (s) indicates target signal – see Eq. (3.1)). It is computed using

$$\widehat{M}_{p,q}^{(s)}(t,f) = \widehat{M}_p(t,f)\widehat{M}_q(t,f), \tag{3.5}$$

where $\widehat{M}_p$ and $\widehat{M}_q$ are the T-F masks representing the estimated speech portion at each T-F unit of microphone p and q, respectively. The estimated masks should be close to one for T-F units dominated by direct sound signals and zero for T-F units dominated by noise

or reverberation. Mask estimation based on deep learning will be discussed later in Chapter 3.2.5. The time delay or direction is then computed as

$$\hat{k} = \underset{k}{\operatorname{argmax}} \sum_{(p,q)\epsilon\Omega} \sum_{t} \sum_{f=1}^{D/2} MGCC_{p,q}(t,f,k), \tag{3.6}$$

where Ω represents the set of microphones pairs in an array used for the summation. Note that the above delay estimation is formulated for a general array with at least two sensors.

Through the product of the masks of individual microphone channels, the weighting mechanism in Eq. (3.5) places more weights on the T-F units dominated by target speech across all the microphone channels. This makes sense as target-dominant T-F units carry cleaner phase information for localization than other ones. Therefore, adding this weighting term should sharpen the peak corresponding to the target source in the summation and suppress the peaks corresponding to noise sources and reverberation.

At a conceptual level, T-F masking guides localization in the following sense. First, T-F masking serves to specify what the target source is through supervised training. Although we are interested in speaker localization in this study, the framework does not change if one is interested in localizing, for example, musical instruments instead. Second, masking suppresses the impact of interfering sounds and reverberation in localization. Without the guidance of masking, traditional DOA estimation could be considered *blind* as it is indiscriminately based on sound energy in one form or another.

One property of the proposed algorithm is that, for relatively clean utterances, estimated mask values would all be close to one. In such a case, the proposed algorithm simply reduces to the classic GCC-PHAT algorithm, which is known to perform very well in clean environments [28].

We point out that our approach is different from applying the GCC-PHAT algorithm to enhanced speech signals obtained via T-F masking. To explain this, let us substitute $\widehat{M}_p(t,f)Y_p(t,f)$ and $\widehat{M}_q(t,f)Y_q(t,f)$ for $Y_p(t,f)$ and $Y_q(t,f)$ in Eq. (3.3). Doing it this way produces the same GCC coefficients as using the unprocessed $Y_p(t,f)$ and $Y_q(t,f)$, because the real-valued masks are cancelled out due to the PHAT weighting (unless time-domain re-synthesis is performed). The proposed algorithm utilizes estimated masks as a weighting mechanism to identify for localization speech dominant T-F units where the phase information is less contaminated, as localization cues are mostly contained in inter-channel phase differences.

Our study first estimates a T-F mask for each single-channel signal and then combines the estimated masks using their product. In this way, the resulting DNN for mask estimation can be readily applied to microphone arrays with various numbers of microphones arranged in arbitrary geometry, although geometrical information is still necessary for DOA estimation. This flexibility distinguishes our algorithms from classification based approaches [15], [38], [98], [99], [199] for DOA estimation, which typically require fixed microphone geometry, fixed number of microphones and fixed spatial resolution for DNN training and testing. In addition, the trained neural network for mask estimation can be directly employed for related tasks such as VAD, spatial covariance matrix estimation, beamforming, and single-channel post-filtering [58], [213].

Following [156], [45], a recent study [119] proposed to use DNN based T-F masking to improve the SRP-PHAT algorithm. This method first averages the log-magnitudes from all the channels and then uses a convolutional neural network to estimate an average mask from the averaged magnitudes. The estimated average mask is then used as weights for

SRP-PHAT. Averaging log-magnitudes would not be a good idea when the signals at different channels vary significantly, for example in the binaural case where interaural level differences can be large. In addition, averaging would incorporate contaminated T-F units for DOA estimation. In contrast, our approach estimates a mask from each microphone signal separately, using features extracted from that microphone. We then combine estimated masks using the product rule in Eq. (3.5). As a result, our approach places more weights on the T-F units dominated by target speech in all the microphone channels. It should, however, be noted that performing channel-wise mask estimation comes at the cost of increased computation compared to estimating an average mask. Furthermore, as described in Chapter 3.2.5, our study uses powerful recurrent neural networks (RNNs) to estimate the IRM [166] and phase-sensitive mask [35], [170], yielding better mask estimation for localization.

3.2.3. Mask-Weighted Steered Response SNR

The GCC-PHAT, SRP-PHAT or BeamScan [84], [217] algorithms steer a beam towards a hypothesized direction and compute the steered-response power of noisy speech to determine whether the hypothesized direction is the target direction, i.e. with the strongest response. The proposed mask-weighted GCC-PHAT algorithm utilizes a T-F mask to emphasize speech dominant T-F units so that the steered-response power of estimated target speech, rather than noisy speech, is used as the location indicator. This section uses steered-response SNR as the indicator, as SNR considers both speech power and noise power, and more importantly, the SNR at each frequency can be bounded between zero and one so that DOA estimation would not be biased towards high-energy lower-frequency components. Specifically, for each direction of interest, we design a

beamformer to point towards that direction, and the direction producing the highest SNR is considered as the predicted target direction [9]. Speech and noise covariance matrices for beamforming and SNR computation can be robustly estimated with the guidance of T-F masking.

Let $\boldsymbol{Y}_{p,q}(t,f) = \left[Y_p(t,f), Y_q(t,f)\right]^T$. The speech and noise covariance matrices between microphone p and q at each frequency are computed in the following way,

$$\widehat{\Phi}_{p,q}^{(s)}(f) = \sum\nolimits_t \widehat{M}_{p,q}^{(s)}(t,f)\boldsymbol{Y}_{p,q}(t,f)\,\boldsymbol{Y}_{p,q}(t,f)^H \Big/ \sum\nolimits_t \widehat{M}_{p,q}^{(s)}(t,f) \tag{3.7}$$

$$\widehat{\Phi}_{p,q}^{(n)}(f) = \sum\nolimits_t \widehat{M}_{p,q}^{(n)}(t,f)\boldsymbol{Y}_{p,q}(t,f)\boldsymbol{Y}_{p,q}(t,f)^H \Big/ \sum\nolimits_t \widehat{M}_{p,q}^{(n)}(t,f) \tag{3.8}$$

where $\widehat{M}_{p,q}^{(s)}(t,f)$ is given in Eq. (3.5) and $\widehat{M}_{p,q}^{(n)}(t,f)$ is computed as (superscript (n) indicates noise or interference)

$$\widehat{M}_{p,q}^{(n)}(t,f) = \left(1 - \widehat{M}_p(t,f)\right)\left(1 - \widehat{M}_q(t,f)\right) \tag{3.9}$$

Motivated by the work in masking-based beamforming for ASR [203], [58] (see also [213]), the weights in Eq. (4.11) are empirically designed so that only the T-F units dominated by speech in both microphone channels are utilized to compute the speech covariance matrix, and the more speech-dominant a T-F unit is, the more weight is placed on it. The noise covariance matrix is computed in a similar fashion, where the noise mask is simply obtained in Eq. (3.9) as the complement of the speech mask.

Next, under the plane-wave and far-field assumption [40], the steering vector for a candidate direction k is hypothesized as

$$\boldsymbol{c}_{p,q}(f,k) = \left[e^{-j2\pi \frac{f}{D} f_s \frac{d_{kp}}{c_s}}, e^{-j2\pi \frac{f}{D} f_s \frac{d_{kq}}{c_s}} \right]^T \tag{3.10}$$

Then, $\boldsymbol{c}_{p,q}(f,k)$ is normalized to unit length

$$\bar{\boldsymbol{c}}_{p,q}(f,k) = \frac{\boldsymbol{c}_{p,q}(f,k)}{\left\| \boldsymbol{c}_{p,q}(f,k) \right\|} \tag{3.11}$$

and an MVDR beamformer is constructed

$$\widehat{\boldsymbol{w}}_{p,q}(f,k) = \frac{\widehat{\Phi}_{p,q}^{(n)}(f)^{-1} \bar{\boldsymbol{c}}_{p,q}}{\bar{\boldsymbol{c}}_{p,q}{}^{H} \widehat{\Phi}_{p,q}^{(n)}(f)^{-1} \bar{\boldsymbol{c}}_{p,q}} \tag{3.12}$$

Afterwards, the SNR of the beamformed signal is estimated as the ratio between the beamformed speech energy and beamformed noise energy

$$\mathrm{SNR}_{p,q}(f,k) = \frac{\widehat{\boldsymbol{w}}_{p,q}(f,k)^H \widehat{\Phi}_{p,q}^{(s)}(f) \widehat{\boldsymbol{w}}_{p,q}(f,k)}{\widehat{\boldsymbol{w}}_{p,q}(f,k)^H \widehat{\Phi}_{p,q}^{(n)}(f) \widehat{\boldsymbol{w}}_{p,q}(f,k)} \tag{3.13}$$

Finally, the speaker location is estimated as

$$\hat{k} = \underset{k}{\mathrm{argmax}} \sum_{(p,q)\epsilon\Omega} \sum_{f=1}^{D/2} \mathrm{SNR}_{p,q}(f,k) \tag{3.14}$$

One issue with Eq. (3.13) is that the computed energy and SNR are unbounded at each frequency band. In such cases, several frequency bands may dominate the SNR calculation. To avoid this problem, we restrict it to between zero and one in the following way

$$\mathrm{SNR}_{p,q}(f,k) = \frac{\widehat{\boldsymbol{w}}_{p,q}(f,k)^H \widehat{\Phi}_{p,q}^{(s)}(f) \widehat{\boldsymbol{w}}_{p,q}(f,k)}{\widehat{\boldsymbol{w}}_{p,q}(f,k)^H \widehat{\Phi}_{p,q}^{(s)}(f) \widehat{\boldsymbol{w}}_{p,q}(f,k) + \widehat{\boldsymbol{w}}_{p,q}(f,k)^H \widehat{\Phi}_{p,q}^{(n)}(f) \widehat{\boldsymbol{w}}_{p,q}(f,k)} \tag{3.15}$$

Eq. (3.15) shares the same spirit as PHAT weighting, where the GCC coefficient at each unit pair is bounded between -1 and 1, making each frequency contribute equally to the summation.

One can also explore alternative ways of weighting different frequency bands. One of them is to place more weights on higher-SNR frequency bands, i.e.

$$\mathrm{SNR}_{p,q}(f,k) = \frac{\bar{M}_{p,q}(f)\, \widehat{\boldsymbol{w}}_{p,q}(f,k)^H \widehat{\Phi}_{p,q}^{(s)}(f) \widehat{\boldsymbol{w}}_{p,q}(f,k)}{\widehat{\boldsymbol{w}}_{p,q}(f,k)^H \widehat{\Phi}_{p,q}^{(s)}(f) \widehat{\boldsymbol{w}}_{p,q}(f,k) + \widehat{\boldsymbol{w}}_{p,q}(f,k)^H \widehat{\Phi}_{p,q}^{(n)}(f) \widehat{\boldsymbol{w}}_{p,q}(f,k)} \tag{3.16}$$

$$\bar{M}_{p,q}(f) = \sum\nolimits_t \widehat{M}_{p,q}^{(s)}(t,f) \Big/ \sum\nolimits_{t,f} \widehat{M}_{p,q}^{(s)}(t,f) \tag{3.17}$$

where the sum of the speech mask $\widehat{M}_{p,q}^{(s)}(t,f)$ within each frequency band is used to indicate the importance of that band for localization. This frequency weighting, which counters the energy normalization, is motivated by the mask-weighted GCC-PHAT algorithm, which implicitly places more weights on frequencies with larger $\bar{M}_{p,q}(f)$. In our experiments, consistently better performance is observed using Eq. (3.16) than using Eq. (3.13) and (3.15) (see Chapter 3.4).

3.2.4. DOA Estimation Based on Steering Vectors

In the recent CHiME-3 and 4 challenges [8], [160], deep learning based T-F masking has been prominently employed for acoustic beamforming and robust ASR [58], [203], [213]. The main idea is to utilize estimated masks to compute the spatial covariance matrices and steering vectors that are critical for accurate beamforming. Remarkable improvements in terms of ASR performance have been reported over conventional beamforming techniques that employ traditional DOA estimation algorithms such as GCC-PHAT [2] and SRP-PHAT [8] for steering vector computation. This success is largely attributed to the power of deep learning based mask estimation [161]. In this context, we

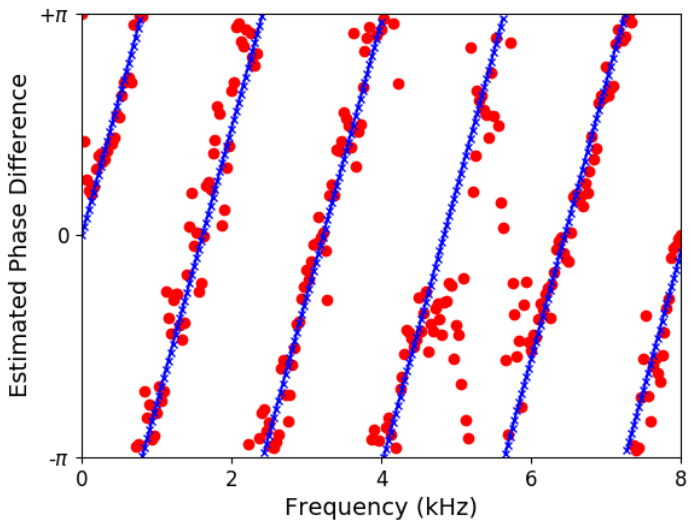


Figure 3-1. Illustration of DOA estimation based on estimated steering vectors for a 2.4 s two-microphone (spacing: 24 cm) signal with babble noise. The SNR level is -6 dB and reverberation time is 0.16 s. Dots indicate the estimated phase differences $\angle(\hat{c}_{p,q}(f))_1 - \angle(\hat{c}_{p,q}(f))_2$ obtained using the IRM, and crosses the fitted phase differences $2\pi\frac{f}{D}f_s\tau_{p,q}(k)$ for a candidate direction k at each frequency.

propose to perform DOA estimation from estimated steering vectors, as they contain sufficient information about the underlying target direction.

Following [203], [213], the steering vector for microphone p and q, $\hat{c}_{p,q}(f)$, is estimated as the principal eigenvector of the estimated speech covariance matrix computed using Eq. (3.7). If $\widehat{\Phi}_{p,q}^{(s)}(f)$ is accurately estimated, it would be close to a rank-one matrix, as the target speaker is a directional source and its principal eigenvector is a reasonable estimate of the steering vector [40].

To derive the underlying time delay or direction, we enumerate all the candidate directions and find the direction that maximizes the following similarity:

$$Sim_{p,q}(f,k) = cos\left(\angle\left(\hat{c}_{p,q}(f)\right)_1 - \angle\left(\hat{c}_{p,q}(f)\right)_2 - 2\pi\frac{f}{D}f_s\tau_{p,q}(k)\right) \tag{3.18}$$

$$\hat{k} = \underset{k}{\text{argmax}} \sum_{(p,q)\epsilon\Omega}\sum_{f=1}^{D/2} Sim_{p,q}(f,k) \tag{3.19}$$

The rationale is that $\hat{\boldsymbol{c}}_{p,q}(f)$ is independently estimated at each frequency, and therefore the estimated phase difference, $\angle\left(\hat{c}_{p,q}(f)\right)_1 - \angle\left(\hat{c}_{p,q}(f)\right)_2$, between the two complex values in $\hat{\boldsymbol{c}}_{p,q}(f)$ does not strictly follow the linear phase assumption. We enumerate all the candidate directions and find as the final estimate a direction k with its hypothesized phase delay $2\pi\frac{f}{D}f_s\tau_{p,q}(k)$ that best matches the estimated phase difference at every frequency band. As illustrated in Figure 3-1, this approach can be understood as performing circular linear regression between the estimated phase difference and frequency index f, where the slope is determined by $\tau_{p,q}(k)$ and the periodic cosine operation is employed to deal with phase wrapping. The cosine operation is naturally bounded between -1 and 1, thus explicit energy normalization as in Eq. (3.3) and (3.15) is not necessary. When there are more than two microphones, we simply combine all the microphone pairs by the summation. We optimize the similarity function through explicit enumeration. Eq. (3.18) in form is similar to Eq. (3.3). The key difference is that the phase difference per frequency is obtained from robustly estimated steering vectors rather than from the observed phase difference at each unit pair.

Similar to Eq. (3.16), we emphasize the frequency bands with higher SNR using $\bar{M}_{p,q}(f)$ given in Eq. (3.17).

$$Sim_{p,q}(f,k) = \bar{M}_{p,q}(f) cos\left(\angle\left(\hat{c}_{p,q}(f)\right)_1 - \angle\left(\hat{c}_{p,q}(f)\right)_2 - 2\pi\frac{f}{D}f_s\tau_{p,q}(k)\right) \quad (3.20)$$

Previous studies [124], [3], [151] have computed time delays from estimated steering vectors at each frequency band or each T-F unit pair. They divide the estimated phase difference by the angular frequency to get the time delay, assuming that the microphones are placed sufficiently close and no phase wrapping occurs. However, using closely spaced microphones would make the time delay too small to be accurately estimated and also make location triangulation harder. When phase wrapping is present, multiple time delays could give exactly the same phase difference at a specific frequency band. Our method addresses this ambiguity via enumerating all the time delays and checking the similarity measure in Eq. (3.18) of each time delay. This method is sensible because a time delay deterministically corresponds to a phase difference. Another difference is that we use DNN based T-F masking for steering vector computation. In contrast, previous studies use spatial clustering [3] or empirical rules [151].

Our proposed algorithm differs from the classic MUSIC algorithm [134] and its recent extension in [201] where a recurrent neural network with uni-directional long short-term memory (LSTM) is used to estimate the IBM and the estimated mask is then utilized to weight spatial covariance matrix estimation for MUSIC. Whereas these studies find the target direction with its hypothesized steering vector orthogonal to the noise subspace, the proposed algorithm directly searches for a direction that is closely matched to target steering vectors between each pair of microphones at all frequencies. The steering vector in our study is robustly estimated using supervised T-F masking. Similar to GCC-PHAT, our algorithm implicitly equalizes the contribution of each frequency as all frequencies

contain information for the DOA estimation of broadband speech signals. In contrast, the pseudospectrum at each frequency in the broadband MUSIC algorithm used in [201] is unbounded, and some frequencies could dominate the summation of the pseudospectrams.

3.2.5. Deep Learning Based T-F Masking

Clearly, the estimated mask of each microphone signal $\widehat{M}_p$ plays an essential role in the proposed algorithms. Deep learning based T-F masking has advanced monaural speech separation and enhancement performance by large margins [161]. Many DNNs have been applied to T-F masking. Among them, RNNs with bi-directional LSTM (BLSTM) have shown consistently better performance over feed-forward neural networks, convolutional neural networks, simple RNNs [176], and RNNs with uni-directional LSTM [191], [66], due to their better modeling of contextual information. In this study, we train an RNN with BLSTM to estimate the IRM (see Chapter 3.3 for more details of BLSTM training). When computing the IRM of a noisy and reverberant utterance, we consider the direct sound as the target signal and the remaining components as interference, as the direct sound contains phase information for DOA estimation.

$$\mathrm{IRM}_p(t,f) = \sqrt{\frac{\left|c_p(f;q)S_p(t,f)\right|^2}{\left|c_p(f;q)S_p(t,f)\right|^2 + \left|H_p(t,f) + N_p(t,f)\right|^2}} \tag{3.21}$$

See Eq. (3.1) for relevant notations in the above equation.

In single-channel speech enhancement, the estimated real-valued mask is element-wise multiplied with the STFT coefficients of unprocessed noisy speech to obtain enhanced speech [166]. In this study, we use an estimated IRM to weight T-F units for DOA estimation. Our study uses log power spectrogram features for mask estimation.

The IRM is *ideal* for speech enhancement only when the mixture phase is the same as the clean phase at each T-F unit. The phase-sensitive mask (PSM) [35], [170] takes the phase difference into consideration by scaling down the ideal mask when the mixture phase is different from the clean phase using a cosine operation. In a way, it represents the best mask if a real-valued mask is multiplied with the STFT coefficients of unprocessed noisy speech for enhancement [35], [192]. We define a form of the phase-sensitive mask in the following way:

$$\mathrm{PSM}_p(t,f) = \max\left\{0, \mathrm{IRM}_p(t,f)\cos\left(\angle Y_p(t,f) - \angle\left(c_p(f;q)S_p(t,f)\right)\right)\right\} \quad (3.22)$$

The inclusion of phase in an ideal mask seems particularly suited for our task as phase is key for localization and we need to identify T-F units with cleaner phase for this task. The cosine term serves to reduce the contributions of contaminated T-F units for localization. Note the difference between the PSM defined in Eq. (3.22) and the definition in [35].

3.3. Experimental Setup

The proposed localization algorithms are evaluated in reverberant environments with strong diffuse babble noise. Our neural network is trained only on simulated RIRs using just single-channel information for mask estimation, and directly tested on three unseen sets of RIRs for DOA estimation using microphone arrays with various numbers of microphones arranged in diverse ways. An illustration of the test setup is shown in Figure 3-2. The first test set includes a relatively matched set of simulated two-microphone RIRs,

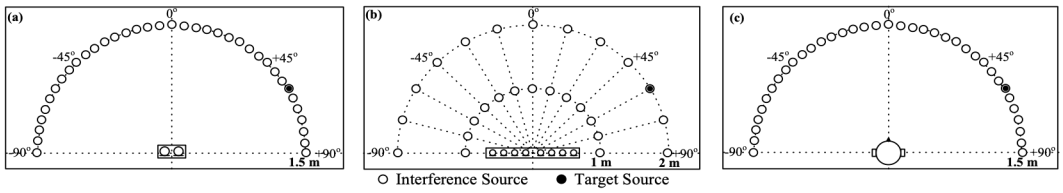


Figure 3-2. Illustration of (a) two-microphone setup, (b) eight-microphone setup, and (c) binaural setup.

the second set consists of real RIRs measured on an eight-microphone array, and the third set contains real binaural RIRs (BRIR) measured on a dummy head.

The RIRs used in the training and validation data are simulated using an RIR generator [47], which is based on the classic image method. An illustration of this setup is shown in Figure 3-2(a). For the training and validation set, we place 36 different interfering speakers at the 36 directions uniformly spaced between -87.5° and 87.5° in steps of 5°, i.e. one different competing speaker in each direction, resulting in a 36-talker diffuse babble noise. The target speaker is randomly placed at one of the 36 directions. For the testing data, we put 37 different interference speakers at the 37 directions spanning from $-90°$ to 90° in steps of 5° (one different competing speaker in each direction), and the target speaker randomly at one of the 37 directions. This way, the test RIRs are different from the RIRs used for training and validation. The distance between each speaker and the array center is 1.5 m (see Figure 3-2(a)). The room size is fixed at $8 \times 8 \times 3$ m, and the two microphones are placed around the center of the room. The spacing between the two microphones is 0.2 m and the microphone heights are both set to 1.5 m. The reverberation time (T60) of each mixture is randomly selected from 0.0 s to 1.0 s in steps of 0.1 s. Target speech comes from the IEEE corpus [68] with 720 sentences uttered by a female speaker. We split the

utterances into sets of 500, 100 and 120 (in the same order as listed in the IEEE corpus) to generate training, validation and test data. To create the diffuse babble noise for each mixture, we randomly pick 37 (or 36) speakers from the 462 speakers in the TIMIT training set and concatenate all the utterances of each speaker, and then place them at all 37 (or 36) directions, with a randomly chosen speech segment of each speaker per direction. Note that we use the first half of the concatenated utterance of each speaker to generate the training and validation diffuse babble noise, and the second half to generate the test diffuse noise. There are in total 50,000, 1,000, and 3,000 two-channel mixtures in the training, validation and test set, respectively. The average duration of the mixtures is 2.4 s. The input SNR computed from reverberant speech and reverberant noise is fixed at -6 dB. Note that if the direct sound is considered as target speech and the remaining signal as noise, as is done in Eq. (3.21) and (3.22), the SNR will vary a lot and be much lower than -6 dB, depending on the direct-to-reverberant ratio (DRR) of the RIRs. We therefore fix the SNR between the reverberant speech and reverberant noise at -6 dB and systematically vary the RIRs to change the SNR between the direct sound signal and the remaining components.

We train our BLSTM using all the single-channel signals (50,000×2 in total) in the training data. The log power spectrogram is used as the input features for mask estimation. Global mean-variance normalization is performed on the input features. The BLSTM consists of two hidden layers each with 600 units in each direction. Sigmoidal units are utilized in the output layer, as the IRM and PSM are bounded between zero and one. During training, the Adam algorithm is utilized to minimize the mean squared error. The frame length is 32 ms, the frame shift is 8 ms, and the sampling rate is 16 kHz. A 512-point FFT (fast Fourier transform) is performed to extract 257-dimensional log spectrogram feature

at each frame. The input and output dimension are thus both 257. The sequence length for BLSTM training and testing is just the utteance length.

The proposed algorithms are also evaluated on the Multi-Channel Impulse Responses Database [50] measured at Bar-Ilan University using a set of eight-microphone linear arrays. We use the microphone array with 8 cm spacing between the two center microphones, and 4 cm spacing between the other adjacent microphones in our experiments, i.e. 4-4-4-8-4-4-4. The setup is depicted in Figure 3-2(b). The RIRs are measured in a room with the size $6\times6\times2.4$ m in steps of $15°$ from $-90°$ to $90°$, at a distance of 1.0 and 2.0 m to the array center, and at three reverberation time (0.16, 0.36 and 0.61 s). Similar to the two-microphone setup, the IEEE and TIMIT utterances are utilized to generate 3,000 eight-channel test utterances for each of the two distances. We put one different interference speaker at each of the 26 locations, resulting in a 26-talker diffuse babble noise. For each of the two distances, the target speaker is placed at one of the 11 interior locations on the hemi-circle (to avoid endfire directions). Note that the RIRs, number of microphones, source-to-array distance, and microphone geometry in this dataset are all unseen during training. In addition, the diffuse babble noise is generated using different locations and different number of interfering speakers. The trained BLSTM is directly tested on the generated test utterances using randomly selected sets of microphones to demonstrate the versatility of our approach to arrays with varying numbers of microphones arranged in diverse geometries.

We also evaluate our algorithm on a binaural setup illustrated in Figure 3-2(c). The real BRIRs[1] captured using a Cortex head and torso simulator (HATS dummy head) in four real

[1]Available at https://github.com/IoSR-Surrey/RealRoomBRIRs.

rooms with different sizes and T60s at the University of Surrey are utilized to generate the test utterances. The dummy head is placed at various heights between 1.7 m and 2.0 m in each room, and the source to array distance is 1.5 m. The real BRIRs are measured using 37 directions ranging from $-90°$ to $90°$ in steps of 5°. The IEEE and TIMIT utterances are utilized to generate 3,000 binaural test utterances in the same way as in the two-microphone setup. The only difference from the two-microphone setup illustrated in Figure 3-2(a) is that now real BRIRs rather than simulated two-channel RIRs are used to generate test utterances. Note that we directly apply the trained BLSTM on this new binaural test set for DOA estimation, although the BLSTM is not trained specifically on any binaural data and the binaural setup is completely unseen during training.

For setup (a) and (b), the location or direction of interest k is enumerated from $-90°$ to $90°$ in steps of 1° on the hemi-circle. The hypothesized time delay between microphone p and q for location or direction k, $\tau_{p,q}(k)$, is computed as $(d_{kq} - d_{kp})/c_s$, where c_s is 343 m/s in the air. Note that setup (b) uses real RIRs measured by a given microphone array, so the distance between each candidate location and each microphone, and microphone configurations are all subject to inaccuracies. In addition, the assumed sound speed may not equal the actual sound speed. These factors complicate accurate localization. For setup (c), the hypothesized time delay cannot be obtained from the distance difference due to the shadowing of head and torso. $\tau_{1,2}(k)$ is instead enumerated from -15 to 15 samples in steps of 0.1 sample. The estimated time delay is then mapped to the azimuth giving the closest time delay. This mapping is obtained from the group delay of the measured BRIRs of the HATS dummy head in the anechoic condition, as is done in [196].

Note that we assume that the target speaker is fixed within each utterance (average length is 2.4 s), and compute a single DOA estimate per utterance. For setup (a) and (c), which use 5º step size for the candidate directions, we measure localization performance using gross accuracy, which considers a prediction correct if it is within 5º (inclusive) of the true target direction. For the Multi-Channel Impulse Response Database with a coarser spatial resolution, we consider a prediction correct if it is within 7.5° of the true direction. Gross accuracy is given as percent correct over all test utterances.

In Eq. (3.6), (3.14) and (3.19), Ω contains all the microphone pairs of an array for the summation.

3.4. Evaluation Results

Table 3-1 presents localization gross accuracy results for two-microphone setup (a), together with the DRR at each T60 and the oracle performance marked in grey. We report DRR together with T60 as it is an important factor for the performance of sound localization in reverberant environments. The rows of eIRM and ePSM in the table mean that estimated IRM and estimated PSM are used for DOA estimation, respectively. All the three proposed algorithms lead to large improvements over GCC-PHAT and MUSIC (on average 72.0%, 86.7% and 75.1% using ePSM vs. 21.6% and 25.2%). PSM estimation yields consistently better performance than IRM estimation for all the algorithms; similar trends are observed from later results in Table 3-2, Table 3-3, and Table 3-4. As is reported in Table 3-1, frequency weighting based on estimated masks, i.e. using Eq. (3.16) and (3.20), leads to consistent improvements (more than 5% on average). Among the three proposed algorithms, mask-weighted steered-response SNR performs the best, especially

Table 3-1. DOA estimation performance (%gross accuracy) of different methods in two-microphone setup.

Method	Frequency Weighting	Mask	T60(s)/DRR(dB)										AVG
			0.0/Inf	0.2/3.8	0.3/-0.4	0.4/-2.5	0.5/-4.0	0.6/-5.1	0.7/-6.0	0.8/-6.8	0.9/-7.4	1.0/-8.0	
GCC-PHAT	-	-	33.7	35.6	30.1	26.1	16.7	15.6	19.5	14.3	15.2	8.9	21.6
MUSIC	-	-	35.1	41.6	33.9	26.7	20.6	20.5	23.6	16.7	19.3	13.9	25.2
Mask-weighted GCC-PHAT	-	eIRM	94.3	95.7	87.0	80.1	74.6	64.0	53.4	49.0	47.2	38.6	68.3
	-	IRM	99.3	99.7	98.7	96.1	96.9	97.1	96.8	94.9	96.2	95.7	97.1
	-	ePSM	96.4	95.4	88.3	82.7	80.1	69.2	59.1	53.7	51.0	44.6	72.0
	-	PSM	100.0	100.0	100.0	100.0	100.0	99.7	99.7	99.3	100.0	99.3	99.8
Mask-weighted Steered-response SNR	Eq. (3.15)	eIRM	94.6	93.7	84.8	78.5	80.1	80.2	68.1	59.5	59.7	57.8	75.7
	Eq. (3.16)	eIRM	95.0	95.0	87.7	84.0	85.7	87.7	75.7	69.7	66.6	64.7	81.2
	Eq. (3.16)	IRM	100.0	99.7	99.1	99.3	99.3	99.4	99.4	99.3	99.3	99.3	99.4
	Eq. (3.15)	ePSM	94.6	95.4	87.0	82.7	87.1	84.7	75.1	65.6	66.6	62.0	80.1
	Eq. (3.16)	ePSM	96.1	96.4	91.1	89.6	91.3	89.0	84.0	76.9	76.9	75.9	86.7
	Eq. (3.16)	PSM	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	99.7	100.0
DOA Estimation from Steering Vectors	Eq. (3.19)	eIRM	89.6	92.4	84.2	73.3	70.4	64.6	55.6	51.4	50.0	40.6	67.2
	Eq. (3.20)	eIRM	93.5	95.7	86.4	80.8	76.7	69.2	61.0	58.8	55.2	47.2	72.4
	Eq. (3.20)	IRM	98.9	99.7	99.1	97.1	97.2	96.8	96.2	94.2	95.9	96.4	97.1
	Eq. (3.19)	ePSM	90.7	92.4	84.5	76.5	72.5	67.9	60.1	51.4	50.7	43.9	69.0
	Eq. (3.20)	ePSM	96.1	97.0	88.3	82.7	80.8	70.5	66.1	58.8	57.2	54.1	75.1
	Eq. (3.20)	PSM	99.6	100.0	100.0	100.0	100.0	99.7	99.7	99.3	99.7	99.3	99.7

when reverberation time is high and DRR is low. For all the three proposed algorithms, using the PSM or IRM results in close to 100% gross accuracy, even when reverberation time is as high as 1.0 s, the DRR is as low as -8.0 dB, and the SNR between reverberant speech and reverberant noise is as low as -6 dB. These oracle results demonstrate the effectiveness of T-F masking: the PSM and IRM can be considered as strong training targets for robust speaker localization, just like for speech separation and enhancement [162], [166]. Better estimated masks in the future will likely produce better localization results.

For the mask-weighted GCC-PHAT algorithm, we have also evaluated the average of estimated mask instead of the product in Eq. (3.5), motivated by [119]. We find that the product rule produces significantly better localization than the average, 68.3% vs. 55.3% using eIRM and 72.0% vs. 61.6% using ePSM. We should note that the average mask is not exactly what is used in [119] and there are many differences between our system and

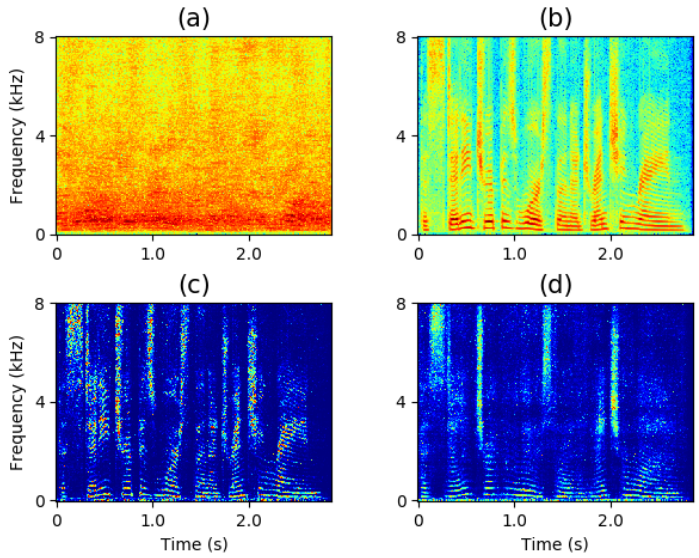


Figure 3-3. Illustration of an estimated IRM for a mixture with babble noise in the two-microphone setup (SNR = -6 dB and T60 = 0.9 s). (a) Mixture log power spectrogram; (b) clean log power spectrogram; (c) IRM; (d) estimated IRM.

[119], as discussed in Chapter 3.2.2. These differences complicate a direct comparison. Another way is to compare the relative improvement over a baseline where no masking is performed. It appears that our overall system obtains larger improvements.

Figure 3-3 illustrates IRM estimation for a very noisy and reverberant mixture. As can be observed by comparing the IRM in Figure 3-3(c) and the estimated IRM in Figure 3-3(d), the estimated mask well resembles the ideal mask in this case, indicating the effectiveness of BLSTM based mask estimation. Upon a closer examination, we observe that the IRM is more accurately estimated at speech onsets and lower frequencies, likely because the direct speech energy is relatively stronger in these T-F regions.

Table 3-2 presents the accuracy of DOA estimation in setup (b), which uses measured real RIRs. For each utterance, we randomly choose two microphones from the eight microphones for testing. Note that the microphone distances can vary from 4 cm at

Table 3-2. DOA estimation performance (%gross accuracy) of different methods in multi-microphone setup by randomly selecting two microphones for each test utterance.

Method	Mask	Distance	T60(s)/DRR(dB)			AVG	Distance	T60(s)/DRR(dB)			AVG
			0.16/10.5	0.36/7.4	0.61/4.7			0.16/6.3	0.36/1.6	0.61/-1.3	
GCC-PHAT	-	1 m	37.9	38.5	31.7	36.1	2m	31.7	29.8	22.8	28.1
MUSIC	-		34.6	35.8	31.7	34.1		30.0	23.9	21.1	25.0
Mask-weighted GCC-PHAT	eIRM		84.0	83.0	84.3	83.7		82.1	74.4	67.5	74.6
	IRM		92.7	91.6	93.0	92.4		92.8	93.0	91.7	92.5
	ePSM		85.6	85.9	83.0	84.9		85.2	78.3	70.6	78.1
	PSM		94.0	94.0	92.5	93.5		93.3	93.2	92.4	93.0
Mask-weighted Steered-response SNR	eIRM		84.0	84.2	82.5	83.6		81.5	69.3	66.6	72.4
	IRM		93.2	92.9	92.7	92.9		93.1	92.2	92.7	92.6
	ePSM		86.7	86.5	86.4	86.5		85.0	77.1	72.4	78.2
	PSM		92.8	93.8	92.5	93.1		95.2	92.6	91.8	93.2
DOA Estimation from Steering Vectors	eIRM		80.4	80.6	81.6	80.8		79.5	67.9	65.3	70.9
	IRM		92.3	90.2	92.8	91.7		92.8	92.6	91.1	92.2
	ePSM		83.6	83.4	81.3	82.8		81.9	73.8	68.1	74.6
	PSM		93.8	93.9	92.2	93.3		92.9	92.9	92.4	92.8

Table 3-3. DOA estimation performance (%gross accuracy, averaged over all reverberation times) of different methods at 2 m distance in multi-microphone setup by randomly selecting different numbers of microphones for each test utterance.

Method	Mask	# microphones						
		2	3	4	5	6	7	8
GCC-PHAT	-	28.1	36.1	38.9	41.8	41.5	41.4	42.8
MUSIC	-	25.0	30.4	31.3	32.2	32.8	32.7	32.8
Mask-weighted GCC-PHAT	eIRM	74.6	89.3	93.8	94.6	95.1	96.0	96.1
	IRM	92.5	98.2	99.6	100.0	100.0	100.0	100.0
	ePSM	78.1	90.3	93.7	95.5	95.9	96.2	96.2
	PSM	93.0	98.7	99.7	100.0	100.0	100.0	100.0
Mask-weighted Steered-response SNR	eIRM	72.4	85.8	90.1	92.1	92.9	93.4	93.5
	IRM	92.6	98.7	99.6	100.0	100.0	100.0	100.0
	ePSM	78.2	90.0	93.5	94.7	95.6	95.8	95.8
	PSM	93.2	98.9	99.8	100.0	100.0	100.0	100.0
DOA Estimation from Steering Vectors	eIRM	70.9	85.6	89.8	91.3	92.2	92.4	92.6
	IRM	92.2	98.3	99.6	100.0	100.0	100.0	100.0
	ePSM	74.6	88.9	92.6	94.4	94.8	95.1	95.1
	PSM	92.8	98.7	99.7	100.0	100.0	100.0	100.0

minimum to 28 cm at maximum for the test utterances. As the DNN in our algorithms only utilizes single-channel information, our approach can still apply even as geometry varies substantially. As can be seen, the proposed algorithms using PSM lead to large improvements over GCC-PHAT and MUSIC, 84.9%, 86.5% and 82.8% vs. 36.1% and 34.1% for 1 m distance, and 78.1%, 78.2% and 74.6% vs. 28.1% and 25.0% for 2 m

Table 3-4. DOA estimation performance (%gross accuracy) of different methods in binaural setup.

Method	Mask	Room - T60(s)/DRR(dB)					AVG
		Anechoic 0.0/Inf	A 0.32/7.2	B 0.47/7.0	C 0.68/10.9	D 0.89/7.3	
GCC-PHAT	-	56.7	28.7	36.6	33.4	25.3	36.0
MUSIC	-	56.4	26.0	36.1	28.0	26.1	34.3
Mask-weighted GCC-PHAT	eIRM	96.6	94.7	94.8	95.1	91.2	94.5
	IRM	100.0	99.4	99.8	99.3	100.0	99.7
	ePSM	97.4	95.3	96.6	95.6	94.3	95.8
	PSM	100.0	99.5	100.0	99.3	100.0	99.8
Mask-weighted Steered-response SNR	eIRM	96.6	88.6	89.1	87.6	85.8	89.5
	IRM	99.7	99.5	99.5	99.2	99.8	99.5
	ePSM	97.4	93.6	93.8	89.3	90.4	92.9
	PSM	100.0	100.0	99.7	99.8	100.0	99.9
DOA Estimation from Steering Vectors	eIRM	97.6	91.3	91.3	86.0	85.2	90.2
	IRM	100.0	99.4	99.8	99.2	99.8	99.6
	ePSM	97.6	95.3	93.9	89.6	89.9	93.3
	PSM	100.0	99.5	99.8	99.0	100.0	99.7

distance. In this setup, the three proposed algorithms perform similarly, with the mask-weighted steered-response SNR performing slightly better. Clearly, the performance is better when the source to array distance is 1 m than 2 m. Using the IRM or the PSM does not reach 100% accuracy in this setup, likely because the aperture size can be as small as 4 cm, posing a fundamental challenge for accurate localization of a distant speaker.

In Table 3-3, we show that our algorithms can be directly extended to multi-channel cases. This is done by combining different microphone pairs as in the classic SRP-PHAT algorithm. For each utterance, we randomly select a number of microphones for testing. As can be seen from the results, using more microphones leads to better performance for all the algorithms. A significant improvement occurs going from two to three microphones, likely because three microphone pairs become available for localization in a three-sensor array versus one pair in a two-sensor array. The performance starts to plateau after five

microphones. Among the proposed algorithms, the mask-weighted GCC-PHAT algorithm performs slightly better than the other two when more microphones become available.

Table 3-4 reports the results on binaural setup (c). Although the neural network trained for mask estimation has not seen binaural signals and binaural geometry, directly applying it to binaural speaker localization results in substantial gains over GCC-PHAT and MUSIC. Notably, the mask-weighted steered-response SNR algorithm is slightly worse than the other two (92.9% vs. 95.8% and 93.3% using ePSM). The reason, we think, is that the energy levels at the two channels cannot be treated as equal as is done in Eq. (3.10), as head shadow effects occur in the binaural setup. For the microphone array setup (a) and (b), assuming equal energy levels is reasonable as there is no blockage from sound sources to an array. Also the localization performance in this binaural setup appears much higher than the two-microphone setup, likely because the DRR is much higher.

3.5. Conclusion

We have investigated a new approach to robust speaker localization that is guided by T-F masking. Benefiting from deep learning based monaural masking, our approach dramatically improves the robustness of conventional cross-correlation, beamforming and subspace based approaches for speaker localization in noisy-reverberant environments. We have found that balancing the contribution of each frequency is important for the DOA estimation of broadband speech signals. Although the neural network is trained using single-channel information, our study shows that it is versatile in its application to arrays with various numbers of microphones and diverse geometries.

Before closing, we emphasize that the proposed approach achieves robust speaker localization as guided by T-F masking. Our experiments find that even for severely corrupted utterances, ratio masking in the proposed algorithms leads to accurate localization. Our study suggests that ideal ratio masks can serve as strong training targets for robust speaker localization. Clearly, the major factor limiting the localization performance is the quality of estimated masks. Nonetheless, the proposed T-F masking guided approach promises further localization improvements as robust speaker localization can directly benefit from the rapid development of deep learning based T-F masking. Through training, masking guidance plays the dual role of specifying the target source and attenuating sounds interfering with localization. T-F masking affords a view of the signal to be localized, as opposed to traditional localization that blindly relies on signal energy.

Chapter 4. Multi-Channel Blind Speaker Separation

This chapter investigates multi-channel talker-independent speaker separation, by tightly integrating complementary spectral and spatial features for deep learning based multi-channel speaker separation in reverberant environments. The primary idea is to localize individual speakers so that an enhancement network can be trained on spatial as well as spectral features to extract the speaker from an estimated direction and with specific spectral structure. This work has been published in ICASSP 2018 [177], [178], Interspeech 2018 [179], and IEEE/ACM T-ASLP in 2019 [180].

4.1. Introduction

Recent years have witnessed major advances of monaural talker-independent speaker separation since the introduction of deep clustering [57], [69], [177], [181], deep attractor networks [20], [94], and permutation invariant training (PIT) [206], [81]. These algorithms address the label permutation problem in the challenging monaural speaker-independent setup [161], [122] and demonstrate substantial improvements over conventional algorithms, such as spectral clustering [5], CASA based approaches [163] and target- or speaker-dependent systems [212], [161].

When multiple microphones are available, spatial information can be leveraged to alleviate the label permutation problem, as speaker sources are directional and typically

spatially separated in real-world scenarios. One conventional stream of research is focused on spatial clustering [40], [103], [70], where individual T-F units are clustered into sources using complex GMMs or their variants based on spatial cues such as inter-channel time, phase or level differences (ITDs, IPDs or ILDs) and spatial spread, under the speech sparsity assumption. However, such spatial cues degrade significantly in reverberant environments and lead to inadequate separation when the sources are co-located, close to one another or when spatial aliasing occurs. In addition, conventional spatial clustering does not exploit spectral information. In contrast, recent developments in deep learning based monaural speaker separation suggest that, even with spectral information alone, remarkable separation can be obtained [122], although most of such studies are only evaluated in anechoic conditions.

One promising research direction is hence to harness the merits of these two streams of research so that spectral and spatial processing can be tightly combined to improve separation and at the same time, make the trained models as blind as possible to microphone array configuration. In [29], [62], monaural deep clustering is employed for T-F masking based beamforming. Their methods follow the success of T-F masking based beamforming in the CHiME challenges [160]. Although beamforming is very helpful in tasks such as robust ASR, for tasks such as speaker separation and speech enhancement, it typically cannot achieve sufficient separation in reverberant environments, when sources are close to each other, or when the number of microphones is limited. For such tasks, further spectral masking would be very helpful. The studies in [21], [22] apply single-channel separation on the outputs of a set of fixed beamformers. A major motivation is that fixed beamformers together with a separate beam prediction network can be efficient to compute

in an online low-latency system. However, their approach requires the information of microphone geometry to carefully design the fixed beamformers, which are manually designed for a single fixed device and typically not as powerful as data-dependent beamformers that can exploit signal statistics for significant noise reduction. In addition, the fixed beamformers point towards a set of discretized directions. This leads to resolution problems and would become cumbersome to apply when elevation is a consideration. Different from the approaches that apply deep clustering and its variants on monaural spectral information, a recent study [178] includes inter-channel phase patterns for the training of deep clustering networks to better resolve the permutation problem. However, this approach only produces a magnitude-domain binary mask and does not exploit beamforming, which is capable of phase enhancement and is known to perform very well especially in modestly reverberant conditions or when many microphones are available.

In this context, our study tightly integrates spectral and spatial processing for blind source separation (BSS), where spatial information is encoded as additional input features to leverage the representational power of deep learning for better separation. The overall proposed approach is a *Separate-Localize-Enhance* strategy. More specifically, a two-channel chimera++ network that takes inter-channel phase patterns into account is first trained to resolve the label permutation problem and perform initial separation. Next, the resulting estimated masks are used in a localization-like procedure to estimate speaker directions and signal statistics. After that, directional (or spatial) features, computed by compensating IPDs or by using data-dependent beamforming, are designed to combine all the microphones for the training of an enhancement network to further separate each source. Here, beamforming is incorporated in two ways: one uses the magnitude produced

by beamforming as additional input features of the enhancement networks to improve the magnitude estimation of each source and the other further considers the phase provided by beamforming as the enhanced phase. The proposed approach aligns with human ability to focus auditory attention on one particular source with its associated spectral structures and arriving from a particular direction, and suppress the other sources [24].

Our study makes five major contributions. First, inter-channel phase and level patterns are incorporated for the training of two-channel chimera++ networks. Second, two effective spatial features are designed for the training of an enhancement network to utilize the spatial information contained in all the microphones. Third, data-dependent beamforming based on T-F masking is effectively integrated in our system by means of its magnitudes and phases. Fourth, a run-time iterative approach is proposed to refine the estimated masks for T-F masking based beamforming. Fifth, the trained models are blind to the number of microphones and microphone geometry. On reverberant versions of the speaker-independent wsj0-2mix and wsj0-3mix corpus [57], spatialized by measured and simulated RIRs, the proposed approach exhibits large improvements over various algorithms including MESSL [102], oracle and estimated time-invariant multi-channel Wiener filter, GCC-NMF [195], ILRMA [78] and multi-channel deep clustering [178].

In the rest of this chapter, we first introduce the physical model in Chapter 4.2, followed by a review of the monaural chimera++ networks [177] in Chapter 4.3. Next, we extend them to two-microphone cases in Chapter 4.4.1. Based on the estimated masks obtained from pairwise microphone processing, Chapter 4.4.2 encodes the spatial information contained in all the microphones as directional features to train an enhancement network for further separation, with or without utilizing the estimated phase produced by

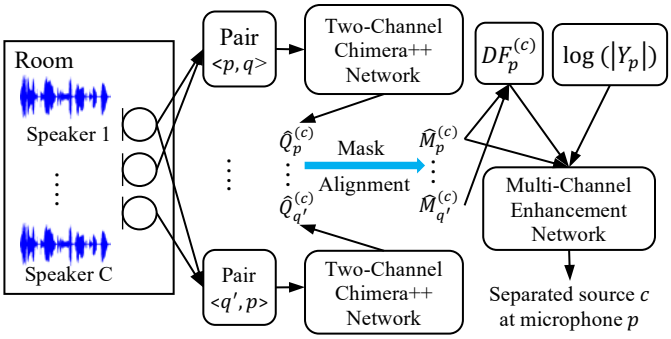


Figure 4-1. Illustration of proposed system for BSS. A two-channel chimera++ network is applied to each microphone pair of interest for initial mask estimation. A multi-channel enhancement network is then applied for each source at a reference microphone for further separation.

beamforming. An optional run-time iterative mask refining algorithm is presented in Chapter 4.4.3. Figure 4-1 illustrates the proposed system. We present our experimental setup and evaluation results in Chapter 4.5 and 4.6, and conclude this paper in Chapter 4.7.

4.2. Physical Models and Objectives

Given a reverberant P -microphone C -speaker time-domain mixture $\boldsymbol{y}[n] = \sum_{c=1}^{C} \boldsymbol{s}^{(c)}[n]$, the physical model in the STFT domain is formulated as:

$$\boldsymbol{Y}(t,f) = \sum_{c=1}^{C} \boldsymbol{S}^{(c)}(t,f), \tag{4.1}$$

where $\boldsymbol{S}^{(c)}(t,f)$ and $\boldsymbol{Y}(t,f)$ respectively represent the P-dimensional STFT vectors of the reverberant image of source c and the reverberant mixture captured by the array at time t and frequency f. Our study proposes multiple algorithms to separate the mixture Y_p captured at a reference microphone p to individual reverberant sources $\hat{S}_p^{(c)}$, by integrating single- and multi-channel processing under a deep learning framework. Note that the

proposed algorithms focus on separation and do not address de-reverberation, although they can be straightforwardly modified for that purpose.

4.3. Monaural Chimera++ Networks

A recent study [177] proposed for monaural speaker separation a novel multi-task learning approach, which combines the permutation resolving capability of deep clustering [57], [69] and the mask inference ability of PIT [206], [81], yielding significant improvements over the individual models. The objective function of deep clustering pulls in the T-F units dominated by the same speaker and pushes away those dominated by different speaker, creating hidden representations that can be utilized by PIT to predict continuous mask values more easily and more accurately. The objective function is also considered as a regularization term to improve the permutation resolving ability of utterance-level PIT. This subsection first introduces deep clustering and PIT, and then reviews the chimera++ networks.

The key idea of deep clustering [57] is to learn a unit-length embedding vector for each T-F unit using a DNN such that for the T-F units dominated by the same speaker, their embeddings are close to one another, while farther otherwise. This way, simple clustering algorithms such as k-means can be applied to the embeddings at run time to determine the speaker assignment at each T-F unit. More specifically, let $\boldsymbol{v}_i$ denote the D-dimensional embedding vector of the i^{th} T-F unit and $\boldsymbol{u}_i$ represent a C-dimensional one-hot vector denoting which of the C sources dominates the i^{th} T-F unit. Vertically stacking them yields the embedding matrix $V \in \mathbb{R}^{TF \times D}$ and the label matrix $U \in \mathbb{R}^{TF \times C}$. The embeddings are learned to approximate the affinity matrix UU^T

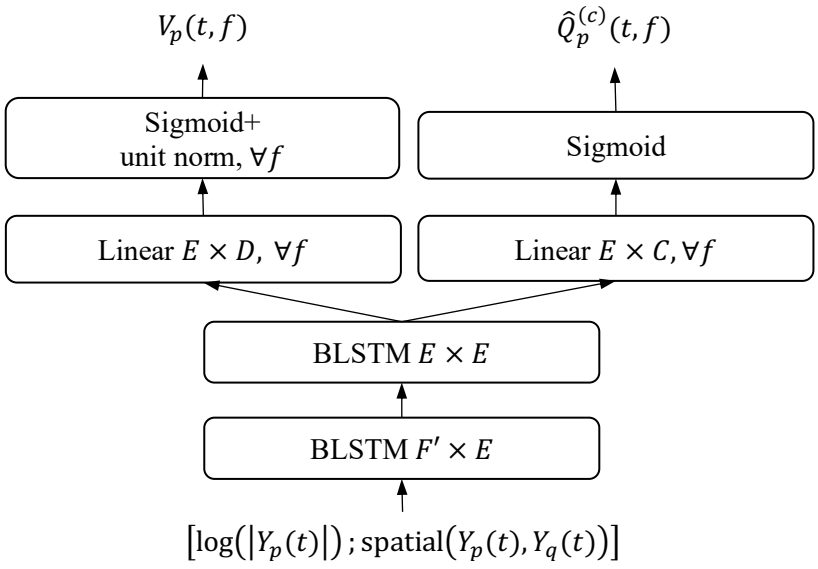


Figure 4-2. Illustration of two-channel chimera++ networks on microphone pair $< p, q >$. $\text{spatial}(Y_p(t), Y_q(t))$ can be a combination of $\cos(\angle Y_p - \angle Y_q)$, $\sin(\angle Y_p - \angle Y_q)$ and $\log(|Y_p|/|Y_q|)$ for microphones p and q. F' represents input feature dimension and E is number of units in each BLSTM layer.

$$\mathcal{L}_{\text{DC}} = \|VV^T - UU^T\|_F^2 \tag{4.2}$$

Recent studies [177] suggested that a variant deep clustering loss function that whitens the embeddings based on a k-means objective leads to better separation performance.

$$\mathcal{L}_{\text{DC,W}} = \left\| V(V^TV)^{-\frac{1}{2}} - U(U^TU)^{-1}U^TV(V^TV)^{-\frac{1}{2}} \right\|_F^2 \tag{4.3}$$

$$= D - trace((V^TV)^{-1}V^TU(U^TU)^{-1}U^TV) \tag{4.4}$$

It is important in deep clustering to discount the importance of silence T-F units, as their labels are ambiguous and they do not carry directional phase information for multi-channel separation [178]. Following [177], the weight of each T-F is computed as the magnitude of each T-F unit over the sum of the magnitudes of all the T-F units. This

weighting mechanism can be simply implemented by broadcasting the weight vector to V and U before computing the loss.

A recurrent neural network with BLSTM units is usually utilized to model the contextual information from past and future frames. The network architecture of deep clustering is shown in the left branch of Figure 4-2.

A permutation-free objective function was proposed in [57], and later reported to work well when combined with deep clustering in [69]. In [206], [81], a permutation invariant training technique was proposed, first showing that such objective function can produce comparable results by itself. The key idea is to train a neural network to minimize the minimum utterance-level loss of all the permutations. The PSM [35] is typically used as the training target. Following [81], the loss function for phase-sensitive spectrum approximation (PSA) is defined as

$$\mathcal{L}_{\text{PIT}} = \min_{\varphi_p \in \Psi} \sum_c \left\| \hat{Q}_p^{\varphi_p(c)} |Y_p| - T_0^{|Y_p|}\left(|S_p^{(c)}| \cos\left(\angle S_p^{(c)} - \angle Y_p \right) \right) \right\|_1, \tag{4.5}$$

where p indexes a microphone channel, Ψ is a set of permutations over C sources, $S_p^{(c)}$ and Y_p are the STFT representations of source c and the mixture captured at microphone p, $T_0^{|Y_p|}(\cdot) = \max\left(0, \min(|Y_p|, \cdot)\right)$ truncates the PSM to the range [0,1] and $\hat{Q}$ denotes the estimated masks. We denote the best permutation as $\hat{\varphi}_p(\cdot)$. Following our recent studies [176], [177], the L_1 loss is used as the loss function, as it leads to consistently better separation than the L_2 loss. Following [177], sigmoidal units are utilized in the output layer to obtain $\hat{Q}_p^{(c)}$ for separation. See the right branch of Figure 4-2 for the network structure.

In [177], a multi-task learning approach is proposed to combine the merits of both algorithms. The objective function is a combination of the two loss functions

$$\mathcal{L}_{\text{chi++}} = \alpha\mathcal{L}_{\text{DC,W}} + (1-\alpha)\mathcal{L}_{\text{PIT}} \tag{4.6}$$

At run time, only the PIT output is needed to make predictions: $\hat{S}_p^{(c)} = \hat{Q}_p^{(c)} Y_p$.

4.4. Proposed Algorithms

4.4.1. Two-Channel Extension of Chimera++ Networks

Following previous studies on multi-channel speech enhancement [73], [214] and speaker separation [178], the key idea of the proposed approach for two-channel separation is to utilize not only spectral but also spatial features for model training. This way, complementary spectral and spatial information can be simultaneously utilized to benefit from the representational power of deep learning to better resolve the permutation problem and achieve better mask estimation. See Figure 4-2 for an illustration of the network architecture.

Given a pair of microphones p and q, it is well-known that, because of speech sparsity, the STFT ratio $Y_p/Y_q = |Y_p|/|Y_q| e^{j(\angle Y_p - \angle Y_q)}$, indicative of the relative transfer function [182], naturally forms clusters within each frequency for spatially separated speaker sources with different time delays to the array [103], [40]. This property establishes the foundations of conventional narrowband spatial clustering [33], [63], [132], [131], which typically first employs spatial information such as directional statistics and mixture STFT vectors for within-frequency bin-wise clustering based on complex GMM and its variants, and then aligns the clusters across frequencies. However, such approaches perform clustering largely based on spatial information, and typically do not leverage spectral cues, although there are recent attempts at using spectral embeddings produced by deep

clustering for spatial clustering [29]. In addition, the clustering is usually only conducted independently within each frequency because of the IPD ambiguity, and thus does not exploit inter-frequency structures. By IPD ambiguity we mean that IPD varies with frequency and the underlying time delay cannot be uniquely determined only from the IPD at a frequency when spatial aliasing and phase wrapping occur.

Our study investigates the incorporation of the spatial information contained in Y_p/Y_q for the training of a two-channel chimera++ network. We consider the following inter-channel phase and level patterns

$$\text{IPD} = \angle e^{j(\angle Y_p - \angle Y_q)} = \text{mod}\big(\angle Y_p - \angle Y_q + \pi, 2\pi\big) - \pi \tag{4.7}$$

$$\text{cosIPD} = \cos(\angle Y_p - \angle Y_q) \tag{4.8}$$

$$\text{sinIPD} = \sin(\angle Y_p - \angle Y_q) \tag{4.9}$$

$$\text{ILD} = \log\left(|Y_p|/|Y_q|\right) \tag{4.10}$$

In our experiments, the combination of cosIPD and sinIPD leads to consistently better performance than the individual ones and the IPD. Our insight is that according to the Euler's formula, the distribution of cosIPD and sinIPD for directional sources naturally follows a helix-like structure with respect to frequency. See Figure 4-3(c) for an illustration of the cosIPD and sinIPD distribution of an anechoic three-speaker mixture. Such helix structure could be exploited by a strong learning machine like deep neural networks to better model inter-frequency structures and achieve better separation. Indeed, in conventional spectral clustering, which significantly motivated the design of deep clustering [5], [57], it is suggested that spectral clustering has the capability of modeling

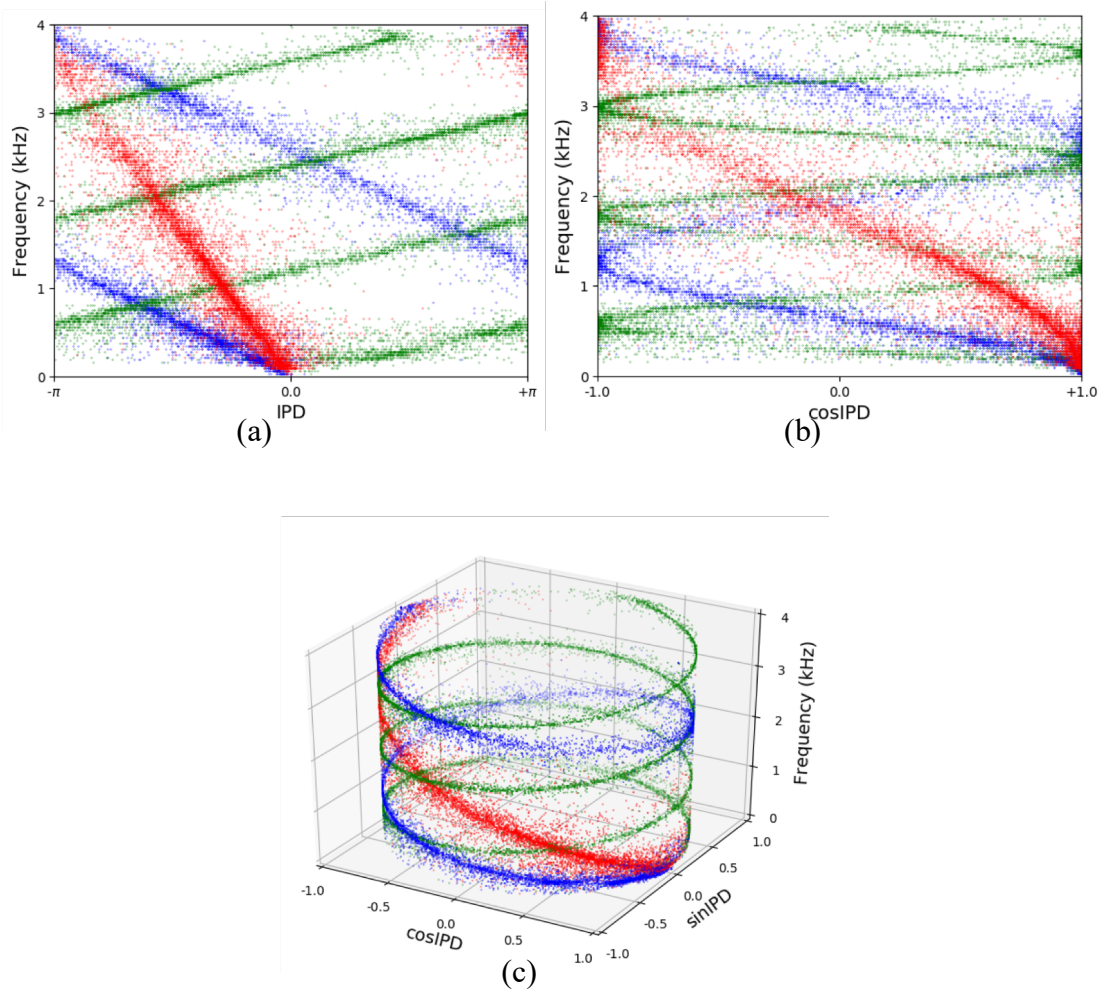


Figure 4-3. Distribution of inter-channel phase patterns of an example anechoic three-speaker mixture with $T_{60} = 0.54\ s$ and microphone spacing $21.6\ cm$. Each T-F unit is colored according to its dominant source. (a) IPD vs. Frequency; (b) cosIPD vs. Frequency; (c) cosIPD and sinIPD vs. Frequency.

such a distribution for clustering [138]. The distribution of an alternative representation, IPD, is depicted in Figure 4-3(a). Clearly, the wrapped lines are not continuous across frequencies because of phase wrapping. Such abrupt discontinuity could make it harder for the neural network to exploit the inter-frequency structures. As a workaround, the distribution of cosIPD is depicted in Figure 4-3(b). Although the continuity improves,

without sinIPD, the number of crossings among the wrapped lines significantly increases. Such crossings, also observed in Figure 4-3(a) and Figure 4-3(c), are mostly resulted from spatial aliasing and phase wrapping, indicating that the inter-channel phase patterns are indistinguishable even though the sources are spatially separated with different time delays and therefore posing fundamental difficulties for conventional BSS techniques that only utilize spatial information. In such cases, spectral information would be the only cue to rely on for separation. Our study hence also incorporates spectral features $\log\left(\left|Y_p\right|\right)$ for model training, and leverages the recently proposed chimera++ networks [177], which have been shown to produce state-of-the-art monaural separation, although only tested in anechoic conditions. Another advantage of including spectral features is that IPD itself is ambiguous across frequencies when the microphone spacing is large, meaning that there does not exist a one-to-one mapping between IPDs and ideal mask values. The incorporation of spectral features could help at resolving this ambiguity, as is suggested in our recent study [178]. Note that the chimera++ network naturally models all the frequencies simultaneously to exploit inter-frequency structures, hence avoiding an error-prone second-stage frequency alignment step that is necessary in conventional narrowband spatial clustering. In addition, the BLSTM better models temporal structures than complex GMMs and their variants, which typically make strong independence assumptions along the temporal axis.

We also incorporate ILDs, computed as in Eq. (4.10), to train chimera++ networks, as they become indicative about target directions especially when the microphone spacing is large and in setups like the binaural setup [163], [152].

4.4.2. Multi-Channel Speech Enhancement

To extend the proposed two-channel approach to multi-channel cases, one straightforward way is to concatenate the inter-channel phase patterns and spectral features of all the microphone pairs as the input features for model training, as is done in [204]. However, this makes the input dimension dependent on the number of microphones and could make the trained model accustomed to one particular microphone geometry. Our recent study [178] proposes an ad-hoc approach to extend two-channel deep clustering to multi-channel cases by performing run-time K-means clustering on a super-vector obtained by concatenating the embeddings computed from each microphone pair. However, it only performs model training using pairwise microphone information, hence incapable of exploiting the geometrical constraints and the spatial information contained in all the microphones.

To build a model that is directly applicable to arrays with any number of microphones arranged in diverse layouts, we think that it is necessary to constructively combine all the microphones into a fixed-dimensional representation. Under this guideline, we propose two fixed-dimensional directional features, one based on compensating ambiguous IPDs using estimated phase differences and the other based on T-F masking based beamforming, as additional inputs to train an enhancement network to improve the mask estimation of each source at the reference microphone. See Figure 4-1 for an illustration of the overall pipeline of our proposed approach. Note that at run time, we need to run the enhancement network once for each source for separation.

4.4.2.1. Compensated IPD

Specifically, for the $P(\geq 2)$ microphones, we first apply the trained two-channel chimera++ network to each of the P pairs consisting of one pair $< p, q >$ between the reference microphone p and a randomly-chosen non-reference microphone q, and $P-1$ pairs $< q', p >$ for any non-reference microphone $q'(\neq p)$. The motivation of using this set of pairs is that we try to obtain an estimated mask for each source at each microphone. Note that for any non-reference microphone q', we can indeed randomly select another microphone to make a pair, but here we simply pair it and the reference microphone p. After obtaining the estimated masks $\hat{Q}_1^{(c)}$, …, $\hat{Q}_P^{(c)}$ of all the P pairs from the two-channel chimera++ network, we permute the C masks at each microphone to create for each source c a new set of masks $\widehat{M}_1^{(c)}$, …, $\widehat{M}_P^{(c)}$ such that they are all aligned to source c. At training time, such an alignment is readily available from Eq. (4.5), i.e. $\widehat{M}_1^{(c)} = \hat{Q}_1^{\hat{\varphi}_1(c)}$,…, $\widehat{M}_P^{(c)} = \hat{Q}_P^{\hat{\varphi}_P(c)}$. At run time, we align the masks using Algorithm 4-1, where an average mask is maintained for each source in the alignment procedure to determine the best permutation for each non-reference microphone. We then compute the speech covariance matrix of each source using the aligned estimated masks, following recent developments of T-F masking based beamforming [203], [58], [213].

$$\widehat{\Phi}^{(c)}(f) = \frac{1}{T}\sum_t \eta^{(c)}(t,f)\boldsymbol{Y}(t,f)\,\boldsymbol{Y}(t,f)^H, \tag{4.11}$$

where T is the number of frames within the utterance and $\eta^{(c)}(t,f)$ is the median [58] of the aligned estimated masks

$$\eta^{(c)} = \text{median}\left(\widehat{M}_1^{(c)}, \ldots, \widehat{M}_P^{(c)}\right) \tag{4.12}$$

Input: $\hat{Q}_1^{(c)}, \ldots, \hat{Q}_P^{(c)}$, for $c = 1, \ldots, C$, and reference microphone p.
Output: Aligned masks $\widehat{M}_1^{(c)}, \ldots, \widehat{M}_P^{(c)}$, for $c = 1, \ldots, C$;
(1) $\widehat{M}_p^{(c)} = \hat{Q}_p^{(c)}$, for $c = 1, \ldots, C$;
(2) $\widehat{M}_{avg}^{(c)} = \widehat{M}_p^{(c)}$, for $c = 1, \ldots, C$;
(3) $counter = 1$;
For non-reference microphone q' in $\{1, \ldots, p-1, p+1, \ldots, P\}$ **do**
(4) $\varphi^* = \text{argmin}_{\varphi \epsilon \Psi} \sum_{c=1}^{C} \left\| W(\widehat{M}_{avg}^{(c)} - \hat{Q}_{q'}^{\varphi(c)}) \right\|_1$;
(5) $\widehat{M}_{q'}^{(c)} = \hat{Q}_{q'}^{\varphi^*(c)}$, for $c = 1, \ldots, C$;
(6) $\widehat{M}_{avg}^{(c)} = (\widehat{M}_{avg}^{(c)} * counter + \widehat{M}_{q'}^{(c)})/(counter + 1)$, for $c = 1, \ldots, C$;
(7) $counter \mathrel{+}= 1$;
End

Algorithm 4-1. Mask alignment procedure at run time. Binary weight matrix W used in step (4) indicates T-F units with energy larger than -40 dB of the mixture's maximum energy.

The key idea here is to only use the T-F units dominated by source c for the estimation of its covariance matrix. The steering vector for each source $\hat{\boldsymbol{r}}^{(c)}(f)$ is then computed as

$$\hat{\boldsymbol{r}}^{(c)}(f) = \mathcal{P}\{\widehat{\Phi}^{(c)}(f)\}, \tag{4.13}$$

where $\mathcal{P}\{\cdot\}$ compute the principal eigenvector. The motivation is that if $\widehat{\Phi}^{(c)}(f)$ is well-estimated, it would be close to a rank-one matrix for a directional speaker source [203], [213], [40]. Its principal eigenvector is hence a reasonable estimate of the steering vector. Note that this steering vector estimation step is essentially similar to DOA estimation.

Following our recent study [183], the directional features are then compensated in the following way:

$$DF_p^{(c)}(t,f) = \frac{1}{P-1} \sum_{<q',p> \epsilon \Omega} \cos\left\{ \angle Y_{q'}(t,f) - \angle Y_p(t,f) - \left(\angle \hat{r}_{q'}^{(c)}(f) - \angle \hat{r}_p^{(c)}(f) \right) \right\}, \tag{4.14}$$

where Ω contains all the $P-1$ pairs between each non-reference microphone q' and the reference microphone p. $\angle Y_{q'}(t,f)-\angle Y_p(t,f)$ represents the observed phase difference and $\angle \hat{r}_{q'}^{(c)}(f)-\angle \hat{r}_p^{(c)}(f)$ the estimated phase difference (or the phase compensation term for source c). The motivation is that if a T-F unit is dominated by source c, the observed phase difference is expected to be aligned with its estimated phase difference. The phase compensation term is used to establish the consistency of the directional features along frequency such that at any frequency and no matter which direction source c arrives from, a value close to one in $DF_p^{(c)}(t,f)$ would indicate that the T-F unit is likely dominated by the source c, while dominated by other sources if much smaller than one, only if the steering vector can be estimated accurately. This property makes the directional features highly discriminative for DNN based T-F masking to enhance the signal from a specific direction. In addition, by establishing the consistency along frequency, the phase compensation term alleviates the ambiguity of IPDs, which could be problematic when directly used for the training of the two-channel chimera++ networks in Chapter 4.4.1. When there are more than two microphones, we simply average the compensated IPDs together. This makes the trained models directly applicable to arrays with various numbers of microphones arranged in diverse geometry. The phase compensation term is designed to combine all the microphone pairs constructively.

There were previous studies [73], [4], [118], [214] utilizing spatial features for deep learning based speech enhancement (i.e. speech vs. noise). The spatial features in those studies are only designed for binaural speech enhancement, where only two sensors are considered and the target is right in the front direction. However, in more general cases, the target speaker may originate in any directions and the spatial features used in those

studies would no longer work well. There was one speech enhancement study [118] considering compensating cosIPDs. However, it needs a separate DOA module that requires microphone geometry, and does not address DOA estimation in a robust way. Diffuseness features have also been applied in deep learning and T-F masking based beamforming for speech enhancement [183], [91]. However, such features are incapable of suppressing directional interferences, which we aim to suppress in this study. On the other hand, directional features are capable of suppressing diffuse noises.

4.4.2.2. T-F Masking Based Beamforming

Another alternative directional feature is derived using beamforming, as beamforming constructively combines target signals captured by different microphones and destructively for non-target signals, only if the signal statistics or target directions critical for beamforming can be accurately determined. Recent development in the CHiME challenges has suggested that deep learning based T-F masking can be utilized to compute such signal statistics accurately [160], demonstrating state-of-the-art robust ASR performance. Here, we leverage this recent development to construct a multi-channel Wiener filter [40]

$$\widehat{\boldsymbol{w}}_p^{(c)}(f) = \left(\widehat{\Phi}^{(y)}(f)\right)^{-1} \widehat{\Phi}^{(c)}(f)\boldsymbol{u}_p, \tag{4.15}$$

where $\widehat{\Phi}^{(y)}(f) = \frac{1}{T}\sum_t \boldsymbol{Y}(t,f)\boldsymbol{Y}(t,f)^H$ is the mixture covariance matrix and $\boldsymbol{u}_p$ a one-hot vector with the p^{th} element being one. Clearly, this way of constructing beamformers is blind to microphone geometry and the number of microphones. The directional feature is then computed as

$$DF_p^{(c)}(t,f) = \log\left(\left|\widehat{\boldsymbol{w}}_p^{(c)}(f)^H \boldsymbol{Y}(t,f)\right|\right) \tag{4.16}$$

4.4.2.3. Enhancement Network I

Using the spatial features alone for enhancement network training is not sufficient enough for accurate separation, as the sources could be spatially close and the reverberation components of other sources could also arrive from the estimated direction. We hence combine $DF_p^{(c)}$ with spectral features $\log(|Y_p|)$, and the initial mask estimates $\widehat{M}_p^{(c)}$ obtained from the two-channel chimera++ network to train an enhancement network to estimate the phase-sensitve spectrum of source c at microphone p. This way, the neural network can take in both spectral and spatial information, and learn to enhance the signals with particular spectral characteristics and arriving from a particular direction. The objective function for training the enhancement network (denoted as **Enh$_1$**) is

$$\mathcal{L}_{\mathrm{Enh}_1} = \left\| \hat{R}_p^{(c)}|Y_p| - T_0^{|Y_p|}\left(|S_p^{(c)}|\cos\left(\angle S_p^{(c)} - \angle Y_p\right)\right) \right\|_1, \tag{4.17}$$

where $\hat{R}_p^{(c)}$ denotes the estimated mask from the Enh$_1$ network. At run time, we execute the enhancement network once for each source, and the separated source c is obtained as $\hat{S}_p^{(c)} = \hat{R}_p^{(c)} Y_p$. Here the mixture phase is used for signal re-resynthesis.

4.4.2.4. Enhancement Network II

The above approach however cannot utilize the enhanced phase provided by beamforming. When the number of microphones is large, the enhanced phase $\hat{\theta}_p^{(c)}(t,f) = \angle\left(\widehat{\boldsymbol{w}}_p^{(c)}(f)^H \boldsymbol{Y}(t,f)\right)$ is expected to be better than $\angle Y_p$, if the speech distortion introduced by beamforming is minimal. We hence use the former as the phase estimate of source c. To obtain a good magnitude estimate, we train an enhancement network (denoted as **Enh$_2$**) to predict the phase-sensitive spectrum of source c with respect to $|Y_p|e^{j\hat{\theta}_p^{(c)}}$, based on the

same features used in Enh$_1$, i.e. $DF_p^{(c)}$, $\log(|Y_p|)$ and $\widehat{M}_p^{(c)}$. The loss function used for training is

$$\mathcal{L}_{\text{Enh}_2} = \left\| \hat{Z}_p^{(c)} |Y_p| - T_0^{|Y_p|} \left(|S_p^{(c)}| \cos\left(\angle S_p^{(c)} - \hat{\theta}_p^{(c)} \right) \right) \right\|_1, \tag{4.18}$$

where $\hat{Z}_p^{(c)}$ denotes the estimated mask of the Enh$_2$ network. At run time, the separated source c is obtained as $\hat{S}_p^{(c)} = \hat{Z}_p^{(c)} |Y_p| e^{j\hat{\theta}_p^{(c)}}$.

Different from the above two ways of integrating beamforming, another alternative is to extract spectral features from the beamformed mixture, train an enhancement network to predict the ideal masks computed from the beamformed sources, and at run time apply the estimated masks to the beamformed mixture [214]. In contrast, our approach uses beamforming results as directional features to improve the mask estimation at the reference microphone p, with or without using the phase of the beamformed mixture, since $S_p^{(c)}$, rather than beamformed sources $\boldsymbol{w}^{(c)}(f)^H \boldsymbol{S}^{(c)}(t,f)$, is considered as the reference for metric computation. This way, we can systematically compare the performance of single- and multi-channel processing, as well as the effects of various algorithms for reverberant source separation. Note that we do not use beamformed sources as the reference signals for metric computation, as they usually contain speech distortions in reverberant environments, and are sensitive to the number of microphones, microphone geometry, and the type of beamformer used to obtain $\boldsymbol{w}^{(c)}(f)$. In addition, for BSS algorithms that do not involve any beamforming, such as spatial clustering or independent component analysis, it is not reasonable to use beamformed sources as the reference signals for evaluation.

We emphasize again that our models, once trained, can be directly applied to arrays with any numbers of microphones arranged in various layouts. At run time, we can first apply the trained two-channel chimera++ network on each microphone pair of interest, then use Eq. (4.14) or (4.16) to constructively combine the spatial information contained in all the microphones, and finally apply the well-trained Enh_1 or Enh_2 networks for further separation. Note that the two-channel chimera++ network essentially functions as a DOA module to estimate target directions and signal statistics for spatial feature computation and beamforming. Indeed, it can be replaced by a monaural chimera++ network, while the two-channel one produces much better initial mask estimation because of the effective exploitation of spatial information, although in a very straightforward way.

4.4.3. Iterative Mask Refinement

In Eq. (4.12), $\eta^{(c)}$ is computed from the estimated masks $\widehat{M}_p^{(c)}$ produced by the chimera++ network that only exploits two-channel information. Such masks are expected to be not as accurate as $\hat{R}_p^{(c)}$ produced by Enh_1, which can utilize the spatial information from all the microphones and suffers less from IPD ambiguity. Using $\hat{R}_p^{(c)}$ for T-F masking based beamforming would hence likely leads to better beamforming results, which can in turn benefit the enhancement networks.

More specifically, at run time, after obtaining $\hat{R}_p^{(c)}$ using Enh_1, we use it in Eq. (4.12) to recompute a multi-channel Wiener filter $\hat{\widehat{\boldsymbol{w}}}_p^{(c)}$ and feed the combination of $\log\left(|\hat{\widehat{\boldsymbol{w}}}_p^{(c)}(f)^H \boldsymbol{Y}(t,f)|\right)$, $\log\left(\left|Y_p\right|\right)$ and $\hat{R}_p^{(c)}$ directly to Enh_2 to get $\hat{\hat{Z}}_p^{(c)}$. The separated source is then obtained as $\hat{S}_p^{(c)} = \hat{\hat{Z}}_p^{(c)}\left|Y_p\right|e^{j\hat{\hat{\theta}}_p^{(c)}}$, where $\hat{\hat{\theta}}_p^{(c)}(t,f) = \angle\left(\hat{\widehat{\boldsymbol{w}}}_p^{(c)}(f)^H \boldsymbol{Y}(t,f)\right)$.

Input: wsj0-3mix;
Output: spatialized reverberant wsj0-3mix;
For each source s_1, source s_2, source s_3 in wsj0-3mix **do**
Sample room length r_x and width r_y from [5,10] m;
Sample room height r_z from [3,4] m;
Sample mic array height a_z from [1,2] m;
Sample displacement n_x and n_y of mic array from $[-0.2,0.2]$ m;
Place array center at $\left[\frac{r_x}{2}+n_x, \frac{r_y}{2}+n_y, a_z\right]$ m;
Sample microphone spacing a_r from [0.02,0.09] m;
For $p = 1{:}\,P(=8)$ **do**
Place mic p at $\left[\frac{r_x}{2}+n_x-\frac{P-1}{2}a_r+(p-1)a_r, \frac{r_y}{2}+n_y, a_z\right]$ m;
End
Sample speaker locations in the frontal plane:

$$s_x^{(1)}, s_y^{(1)}, s_z^{(1)} = a_z;$$
$$s_x^{(2)}, s_y^{(2)}, s_z^{(2)} = a_z;$$
$$s_x^{(3)}, s_y^{(3)}, s_z^{(3)} = a_z;$$

such that any two speakers are at least 15° apart from each other with respect to the array center, and the distance from each speaker to the array center is in between [0.75,2] m;
Sample T60 from [0.2,0.7] s;
Generate impulse responses using RIR generator and convolve them with s_1, s_2 and s_3;
Concatenate channels of reverberated s_1, s_2 and s_3, scale them to match SIR among original s_1, s_2 and s_3, and add them to obtain reverberated mixture;
End

Algorithm 4-2. Data spatialization process (simulated RIRs).

We denote this iterative mask estimation approach as **Enh$_1$+Enh$_2$**. We emphasize this approach is performed at run time and does not require any model training. $\hat{R}_p^{(c)}$ can be improved with more iterations, but we only do one iteration due to computation considerations.

4.5. Experimental Setup

We train our models using only simulated RIRs, while test on simulated as well as real-recorded RIRs. The RIRs are convolved with the anechoic two-speaker and three-speaker

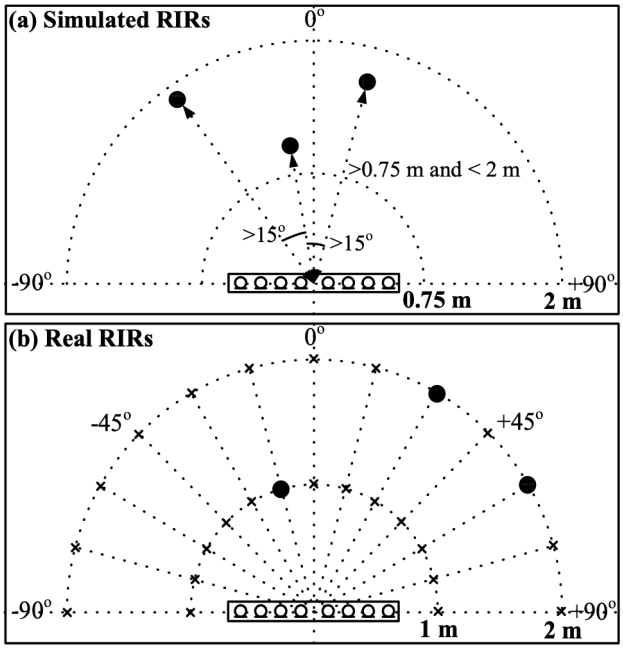


Figure 4-4. Illustration of experimental setup.

mixtures in the recently proposed wsj0-2mix and wsj0-3mix corpus [57], each of which contains 20,000, 5,000 and 3,000 anechoic monaural speaker mixtures in its 30-hour training, 10-hour validation and 5-hour test data. Note that the speakers in the training set and test set are not overlapped. The task is hence speaker-independent. The signal to interference ratio (SIR) for wsj0-2mix mixtures are randomly drawn from -5 to 5 dB. For wsj0-3mix, the third speaker is added such that its energy is the same as that of the first two speakers combined. The sampling rate is 8 kHz.

The data spatialization process using simulated RIRs for wsj0-3mix is detailed in Algorithm 4-2. The RIR generator [47] is employed to generate the simulated RIRs. The general guideline is to make the setup as random as possible while still subject to realistic constraints. For each wsj0-3mix mixture, we randomly generate a room with random room

characteristic, speaker locations, and microphone spacing. Our study considers a linear array setup, where the target speakers are placed in the frontal plane and are at least 15° apart from each other. We generate 20,000, 5,000, and 3,000 eight-channel mixtures for training, validation and testing, respectively. A T60 value for each mixture is randomly drawn in the range [0.2, 0.7] s. See Figure 4-4(a) for an illustration of this setup. The spatialization of wsj0-2mix is performed similarly. The average speaker-to-microphone distance is 1.38 m with 0.37 m standard deviation and the average DRR is 0.49 dB with 3.92 dB standard deviation.

We also generate another 3,000 eight-channel mixtures using the Multi-Channel Impulse Responses Database [50], which is recorded using eight-microphone linear arrays with three different inter-microphone spacing, including 3-3-3-8-3-3-3, 4-4-4-8-4-4-4, 8-8-8-8-8-8-8 cm, under three reverberant time (0.16, 0.36, 0.61 s) created by using a number of covering panels on the walls. The RIRs are measured in steps of $15°$ from $-90°$ to $90°$ and at a distance of 1 m and 2 m to the array center, in a room with size approximately at $6\times6\times2.4$ m. See Figure 4-4(b) for an illustration of this setup. For each mixture, we place each speaker in a random direction and at a random distance, using a randomly-chosen linear array and a randomly-chosen reverberation time among 0.16, 0.36 and 0.61 s. Note that for any two speakers, they are at least 15° apart with respect to the array center. The average DRR is 2.8 dB with 3.8 dB standard derivation in this case. We emphasize that this is a very realistic setup, as it is speaker-independent and more importantly, we use simulated RIRs for training and real RIRs for testing.

At run time, we randomly pick a subset of microphones for each utterance for testing. The aperture size can be 2 cm at minimum and 63 cm at maximum for the simulated RIRs, and 3 cm and 56 cm for the real RIRs.

The chimera++ and enhancement network respectively contains four and three BLSTM layers, each with 600 units in each direction. The window size is 32 ms and the hop size is 8 ms. A 256-point DFT is applied to extract 129-dimensional log magnitude features after square-root Hann window is applied to the signal. The α in Eq. (4.6) is empirically set to 0.975 and the embedding dimension set to 20, following [177]. We emphasize that the enhancement network is trained using the directional features computed from various numbers of microphones, as the quality of the directional features varies with the number of microphones. For all the input features, we apply global mean-variance normalization before feed-forwarding.

Following the SiSEC challenges [142], average signal-to-distortion ratio (SDR) computed using the *bss_eval_images* software is used as the major evaluation metric. We also report average perceptual estimation of speech quality (PESQ) and extended short-time objective intelligibility (eSTOI) [72] scores to measure speech quality and intelligibility.

We consider the reverberant image of each source at the reference microphone, i.e. $s_p^{(c)}$, as the reference signal for metric computation.

Table 4-1. SDR (dB) results on spatialized reverberant wsj0-2mix using up to two microphones.

Approaches	Input Features	Simu RIRs	Real RIRs
Unprocessed	-	0.0	0.0
1ch PIT	$\log(\lvert Y_p \rvert)$	7.5	7.3
1ch deep clustering	$\log(\lvert Y_p \rvert)$	7.3	7.4
1ch chimera++	$\log(\lvert Y_p \rvert)$	8.4	8.4
2ch chimera++	$\log(\lvert Y_p \rvert)$,IPD	10.2	9.8
2ch chimera++	$\log(\lvert Y_p \rvert)$,cosIPD	9.7	10.0
2ch chimera++	$\log(\lvert Y_p \rvert)$,cosIPD,sinIPD	10.4	10.1
+ Enh$_1$	$\log(\lvert Y_p \rvert)$,$\widehat{M}_p^{(c)}$	10.7	10.5
+ Enh$_1$	$\log(\lvert Y_p \rvert)$,$DF_p^{(c)}$(Eq. (4.14)),$\widehat{M}_p^{(c)}$	10.8	10.7
+ Enh$_1$	$\log(\lvert Y_p \rvert)$,$DF_p^{(c)}$(Eq. (4.16)),$\widehat{M}_p^{(c)}$	11.1	11.1
2ch chimera++	$\log(\lvert Y_p \rvert)$,cosIPD,sinIPD,ILD	10.4	10.1

Table 4-2. SDR (dB) results on spatialized reverberant wsj0-3mix using up to two microphones.

Approaches	Input Features	Simu RIRs	Real RIRs
Unprocessed	-	-3.3	-3.2
1ch chimera++	$\log(\lvert Y_p \rvert)$	4.0	4.0
2ch chimera++	$\log(\lvert Y_p \rvert)$,IPD	7.1	6.1
2ch chimera++	$\log(\lvert Y_p \rvert)$,cosIPD	5.8	5.9
2ch chimera++	$\log(\lvert Y_p \rvert)$,cosIPD,sinIPD	7.3	6.3
+ Enh$_1$	$\log(\lvert Y_p \rvert)$,$\widehat{M}_p^{(c)}$	7.6	6.7
+ Enh$_1$	$\log(\lvert Y_p \rvert)$,$DF_p^{(c)}$(Eq. (4.14)),$\widehat{M}_p^{(c)}$	7.8	6.9
+ Enh$_1$	$\log(\lvert Y_p \rvert)$,$DF_p^{(c)}$(Eq. (4.16)),$\widehat{M}_p^{(c)}$	7.9	7.1

4.6. Evaluation Results

We first report the results on the reverberant wsj0-2mix spatialized using the simulated RIRs in the second last column of Table 4-1. Clearly, the chimera++ network shows clear improvements over the individual models (8.4 vs. 7.5 and 7.3 dB), which align with the findings in [177]. Even with random microphone spacing, incorporating inter-channel phase patterns for model training produces large improvement compared with only using

monaural spectral information. This is likely because inter-channel phase patterns naturally form clusters within each frequency regardless of microphone spacing, and we use a clustering-based DNN model to exploit such information for separation. Among various forms of IPD features, the combination of cosIPD and sinIPD leads to consistently better performance over using IPD or cosIPD (10.4 vs. 10.2 and 9.7 dB), likely because this combination naturally maintains the helix structures that can be exploited by the network. Further including the ILD features for training does not lead to clear improvement (10.4 vs. 10.4 dB), likely because level differences are very small in far-field conditions. Using the Enh_1 network brings further improvement as it provides better magnitude estimates. Compensating IPDs (i.e. Eq. (4.14)) using estimated phase differences to reduce the ambiguity and using beamforming results (i.e. Eq. (4.16)) as directional features push the performance from 10.4 to 10.8 and 11.1 dB, respectively. The former feature is worse than the latter one, likely because the former is mathematically similar to the delay-and-sum beamformer, which is known to be less powerful than the multi-channel Wiener filter. In the following experiments, we use Eq. (4.16) to compute the directional feature if not specified. The last column of Table 4-1 presents the results on the real RIRs. The performance is as comparably good as on the simulated RIRs, although the model is trained only on the simulated RIRs.

Table 4-2 presents the results obtained on the spatialized wsj0-3mix using the simulated RIRs and real RIRs, with up to two microphones. Similar trends as in Table 4-1 are observed.

Table 4-3 and Table 4-4 compare the proposed algorithms with other systems along with the oracle performance of various ideal masks, using up to eight microphones, and in

Table 4-4. Performance comparison with other approaches on real RIRs using various numbers of microphones on spatialized reverberant wsj0-3mix.

Metrics	#mics	Mixture	MESSL [102]	GCC-NMF [195]	ILRMA [78]	MCDC [178]	MC-Chimera++	Using $\eta^{(c)}$ in Eq. (4.12)	eMCWF	Enh_1	Enh_2	Enh_1+ Enh_2	tPSM-MCWF	Oracle Masks			
														IRM	IBM	tPSM	MC-tPSM
SDR (dB)	2	-3.2	2.0	2.6	-	5.6	5.5	6.6	3.9	7.1	7.3	7.4	4.5	9.2	10.1	11.3	11.6
	3		-	-	4.6	6.1	5.9	6.7	4.9	7.5	7.9	8.2	5.7				12.1
	4		-	-	5.0	6.3	6.2	7.0	5.7	7.8	8.4	8.8	6.5				12.5
	5		-	-	5.1	6.4	6.3	7.2	6.3	8.0	8.9	9.4	7.2				12.9
	6		-	-	5.2	6.5	6.4	7.3	6.7	8.2	9.3	9.8	7.7				13.2
	7		-	-	5.2	6.5	6.4	7.3	7.0	8.3	9.6	10.1	8.2				13.5
	8		-	-	5.3	6.5	6.4	7.3	7.3	8.4	9.8	10.4	8.5				13.7
PESQ	2	1.67	1.87	1.68	-	1.49	1.48	2.45	2.10	2.48	2.55	2.59	2.14	3.60	2.87	3.64	3.73
	3		-	-	2.22	1.55	1.54	2.46	2.26	2.64	2.74	2.81	2.30				3.79
	4		-	-	2.26	1.57	1.56	2.53	2.35	2.73	2.85	2.94	2.41				3.83
	5		-	-	2.28	1.58	1.57	2.54	2.43	2.81	2.95	3.05	2.48				3.85
	6		-	-	2.29	1.59	1.58	2.56	2.48	2.84	3.00	3.12	2.54				3.87
	7		-	-	2.30	1.59	1.59	2.56	2.52	2.88	3.05	3.17	2.59				3.89
	8		-	-	2.31	1.59	1.59	2.57	2.55	2.90	3.09	3.21	2.63				3.91
eSTOI (%)	2	37.5	43.3	37.9	-	53.0	52.4	62.5	47.5	65.4	66.9	68.2	49.4	87.6	80.4	88.5	90.2
	3		-	-	54.3	55.5	55.0	62.9	53.2	68.5	70.7	72.5	55.9				91.2
	4		-	-	56.3	56.7	56.4	64.9	57.2	70.7	73.4	75.5	60.0				91.8
	5		-	-	57.0	57.3	56.9	65.2	60.1	72.4	75.5	77.8	63.1				92.3
	6		-	-	57.5	57.6	57.3	65.9	62.2	73.4	76.8	79.2	65.4				92.7
	7		-	-	57.8	57.7	57.4	65.8	63.9	74.2	77.9	80.3	67.4				93.0
	8		-	-	58.0	57.6	57.6	66.2	65.2	74.7	78.6	81.1	69.0				93.3

terms of SDR, PESQ and eSTOI. Because of utilizing the phase provided by beamforming, Enh_2 shows consistent improvement over Enh_1, especially when more microphones are available. This justifies the proposed way of integrating beamforming for separation. Performing run-time iterative mask refinement using Enh_1+Enh_2 leads to slight improvement over Enh_2 in the two-speaker case, while clear improvement is observed in the three-speaker case, especially when more microphones are available. This indicates the effectiveness of using $\widehat{R}_p^{(c)}$ for T-F masking based beamforming, especially when $\widehat{M}_p^{(c)}$ is not good enough.

Recent studies [62] apply monaural deep clustering on each microphone signal to derive a T-F masking based beamformer for each frequency for separation. To compare with their algorithms, we use the truncated PSM (tPSM), computed as

$T_0^{1.0}\left(|S_p^{(c)}|\cos\left(\angle S_p^{(c)} - \angle Y_p\right)/|Y_p|\right)$, in Eq. (4.12) to compute oracle $\widehat{\Phi}^{(c)}$ and report oracle time-invariant MCWF results (denoted as tPSM-MCWF). We also report the estimated time-invariant MCWF (eMCWF) performance obtained using $\widehat{M}_p^{(c)}$ computed from the two-channel chimera++ network. Clearly, the beamforming approach requires relatively large number of microphones to produce reasonable separation. Although using estimated masks, the eMCWF is comparable to tPSM-MCWF. As can be observed, both of them are not as good as Enh_2, which combines beamforming with spectral masking. We also compare the proposed algorithms with MESSL[2] [102], a popular wideband GMM based spatial clustering algorithm proposed for two-microphone arrays, and GCC-NMF[3] [195], a location based stereo BSS algorithm, where dictionary atoms obtained from non-negative matrix factorization (NMF) are assigned to individual sources over time according to their time difference of arrival estimates obtained from GCC-PHAT. Note that oracle microphone spacing information is supplied to MESSL and GCC-NMF for the enumeration of time delays. Independent low-rank matrix analysis (ILRMA)[4] [78], originated from the ICA stream of research, is a strong and representative algorithm for determined and over-determined BSS. It unifies independent vector analysis (IVA) and multi-channel NMF by exploiting NMF decomposition to capture the spectral characteristics of each source as the generative source model in IVA. The recently proposed multi-channel deep clustering (MCDC) [178] integrates conventional spatial clustering with deep clustering by including inter-channel phase patterns to train deep

[2]Available at https://github.com/mim/messl.

[3]Available at https://github.com/seanwood/GCC-nmf.

[4]Available at http://d-kitamura.net/programs/ILRMA_release20180411.zip.

clustering networks. Its extension to multi-channel cases is achieved by first applying a well-trained two-channel deep clustering model on every microphone pair, then stacking the embeddings obtained from all the pairs, and finally performing K-means on the stacked embeddings to obtain an estimated binary mask for separation. Following the suggestions by an anonymous reviewer, we evaluate two extensions of MCDC as alternative ways of exploiting multi-channel spatial information. The first one, denoted as MC-Chimera++, concatenates the embeddings provided by our two-channel chimera++ network for K-means clustering, and the second one uses the median mask produced in Eq. (4.12) for separation, i.e. $\hat{S}_p^{(c)} = \eta^{(c)} Y_p$. Clearly, the proposed algorithms are consistently better than the MCDC approach and the two extensions, likely because the proposed algorithm is more end-to-end and better exploits spatial information contained in more than two microphones.

The performance of various oracle masks is presented in the last columns of Table 4-3 and Table 4-4. The IBM is computed based on which source is dominant at each T-F unit. The IRM is calculated as the magnitude of each source over the sum of all the magnitudes. Compared with such monaural ideal masks that use mixture phase for re-synthesis, the multi-channel tPSM (MC-tPSM), calculated as $T_0^{1.0}(|S_p^{(c)}|\cos(\angle S_p^{(c)} - \hat{\theta}_p^{(c)})/|Y_p|)$ where $\hat{\theta}_p^{(c)}$ here is computed from tPSM-MCWF and used as the phase for re-synthesis, is clearly better and becomes even better when more microphones are available. Note that MC-tPSM represents the upper bound performance of Enh$_2$. The results clearly show the effectiveness of using $\hat{\theta}_p^{(c)}$ as the phase estimate.

By exploiting spatial information, we improve the performance of monaural chimera++ network from 8.4 to 11.2 dB when using two microphones and to 14.2 dB when using eight

microphones on the spatialized wsj0-2mix corpus, and from 4.0 to 7.4 and 10.4 dB on the spatialized wsj0-3mix corpus. These results are comparable to the oracle performance of the monaural IBM, IRM and tPSM in terms of the SDR metric, confirming the effectiveness of multi-channel processing.

4.7. Conclusion

We have proposed a novel approach that combines complementary spectral and spatial features for deep learning based multi-channel speaker separation in reverberant environments. This spatial feature approach is found to be very effective for improving the magnitude estimate of the target speaker from an estimated direction and with particular spectral structures. In addition, leveraging the enhanced phase provided by masking based beamforming driven by a two-channel chimera++ network produces further improvements.

Chapter 5. Magnitude Based Phase Reconstruction

This chapter investigates phase reconstruction for deep learning based monaural talker-independent speaker separation in the STFT domain. The key observation is that, for a mixture of two sources, with their magnitudes accurately estimated and under a geometric constraint, the absolute phase difference between each source and the mixture can be uniquely determined; in addition, the source phases at each T-F unit can be narrowed down to only two candidates. To pick the right candidate, we propose three algorithms based on iterative phase reconstruction, group delay estimation, and phase-difference sign prediction. At the time of publication, state-of-the-art results are obtained on the publicly available wsj0-2mix and 3mix corpus. This work has been published in Interspeech 2018 [181] and ICASSP 2019 [184].

5.1. Introduction

Audio source separation concerns the separation of a C-source discrete time-domain mixture $y[n] = \sum_{c=1}^{C} s^{(c)}[n]$ to its individual time-domain sources $s^{(c)}$. As speech is short-time stationary, a common approach decomposes the time-domain mixture to frequency domain to reveal its frequency components using STFT, and performs separation therein. One major recent advance is the introduction of DNN for the estimation of the IBM, IRM, spectral magnitude mask (SMM) [161], or PSM, where source separation is

converted to a magnitude-domain T-F unit level classification or regression problem, typically retaining the mixture phase for re-synthesis. Notable works include masking based speech enhancement studies [161], [166], [165], and speaker separation studies such as deep clustering (DC) [57], [177], [181] and PIT [122], [206]. These studies suggest that magnitude estimation can be substantially improved using deep learning based T-F masking.

In this context, this study investigates magnitude-based methods for phase reconstruction for monaural speaker separation. The key insight is that the possible solutions of phase can be significantly narrowed down given sufficiently accurate magnitude estimates, under the following geometric constraint in the STFT domain

$$Y(t,f) = \sum_{c=1}^{C} S^{(c)}(t,f) = \sum_{c=1}^{C} A^{(c)}(t,f) e^{j\theta^{(c)}(t,f)}, \tag{5.1}$$

where $S^{(c)}(t,f)$ and $Y(t,f)$ respectively denote the STFT values of source signal c and the mixture signal y at time t and frequency f, and $A^{(c)}(t,f) = |S^{(c)}(t,f)|$ and $\theta^{(c)}(t,f) = \angle S^{(c)}(t,f)$ are the magnitude and phase of $S^{(c)}(t,f)$, respectively. In the simplest case, suppose that there are only two sources and the two magnitude spectrums can be perfectly estimated (i.e. $\hat{A}^{(c)}(t,f) = A^{(c)}(t,f)$), are there any closed-form solution for phase estimation? It would be reasonable to say yes as there are two equations with two unknowns

$$|Y(t,f)|\cos\left(\angle Y(t,f)\right) = \hat{A}^{(1)}(t,f)\cos\left(\hat{\theta}^{(1)}(t,f)\right) + \hat{A}^{(2)}(t,f)\cos\left(\hat{\theta}^{(2)}(t,f)\right) \tag{5.2}$$

$$|Y(t,f)|\sin\left(\angle Y(t,f)\right) = \hat{A}^{(1)}(t,f)\sin\left(\hat{\theta}^{(1)}(t,f)\right) + \hat{A}^{(2)}(t,f)\sin\left(\hat{\theta}^{(2)}(t,f)\right) \tag{5.3}$$

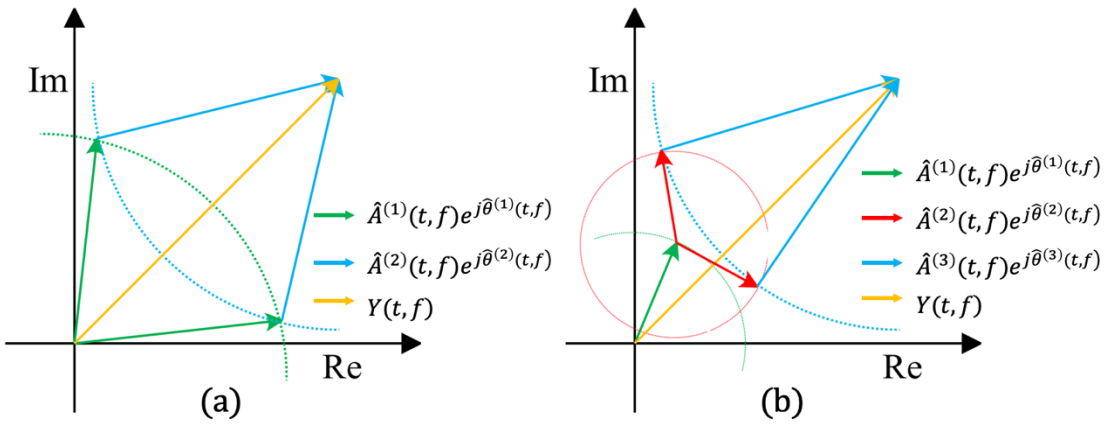


Figure 5-1. Illustration of sign ambiguity when magnitudes are known in the complex plane. (a) Two-source case; (b) three-source case: for each possible $\hat{\theta}^{(1)}(t,f)$, there could be two solutions for $\hat{\theta}^{(2)}(t,f)$ and $\hat{\theta}^{(3)}(t,f)$.

However, the underlying phase cannot be determined, because depending on the sign of the phase difference, there are two candidates satisfying the above two equations

$$\hat{\theta}^{(1)}(t,f) = \angle Y(t,f) \pm \arccos\left(\frac{|Y(t,f)|^2 + \hat{A}^{(1)}(t,f)^2 - \hat{A}^{(2)}(t,f)^2}{2|Y(t,f)|\hat{A}^{(1)}(t,f)}\right) \tag{5.4}$$

$$\hat{\theta}^{(2)}(t,f) = \angle Y(t,f) \mp \arccos\left(\frac{|Y(t,f)|^2 + \hat{A}^{(2)}(t,f)^2 - \hat{A}^{(1)}(t,f)^2}{2|Y(t,f)|\hat{A}^{(2)}(t,f)}\right) \tag{5.5}$$

as is also suggested in earlier studies [107], [108]. See Figure 5-1(a) for an illustration. Intuitively, this sign ambiguity occurs because the phase of each source could be either ahead of or behind the mixture phase within each T-F unit in an almost random way, posing fundamental difficulties for STFT- or time-domain phase estimation. One thing we can conclude, though, is that one of the two candidates is the true $\theta^{(1)}(t,f)$ and $\theta^{(2)}(t,f)$.

To resolve this sign ambiguity, we think that inter-T-F unit phase relations such as group delay (GD) or instantaneous frequency [109] and phase regularizations such as phase

consistency [86] could help. We propose three algorithms for phase reconstruction, leveraging good magnitude estimates produced by DNNs. The first one uses estimated magnitudes to drive an iterative phase reconstruction algorithm, which could implicitly resolve the sign ambiguity. The second one finds a sign assignment per T-F unit such that the resulting GD is closest to an estimated one. The third one implicitly predicts a sign at each T-F unit within a neural network that enforces the geometric constraint in Eq. (5.1).

For a mixture with $C \geq 3$, even if the magnitudes are known, there are still infinite numbers of phase candidates satisfying the geometric constraint, as is illustrated in Figure 5-1(b). This suggests that it could be helpful to approach multi-source separation from a *one-vs.-the-rest* angle, where a model is trained to estimate the magnitude of source c and the magnitude of the rest sources combined (denoted as $\neg c$), and at run time, the model is applied once for each source for separation. This way, there are only two possible phase candidates at each T-F unit to resolve for each source. For speaker separation, our study hence first uses a chimera++ network [177] to perform C-speaker separation to resolve the permutation problem and then uses an enhancement network taking into account the initial separation results of source c to further estimate the magnitudes of source c and $\neg c$ for phase reconstruction. Our best performing algorithm achieves state-of-the art performance on the public wsj0-2mix and 3mix dataset [57], at the time of publication.

Why do we rely so much on magnitude estimates for phase reconstruction? This is because magnitude is much more structured and predictable than phase, and also more stable. Even if the signal is shifted slightly, the magnitude remains almost unchanged, while the phase will exhibit a phase change at every frequency and become very random if

Input: Estimated magnitudes $\hat{A}^{(c')}$ and starting phases $\hat{\vartheta}^{(c')}(0)$ initialized as mixture phase $\angle Y$ or enhanced phase $\hat{\theta}^{(c')}$ for c' in $\{c, \neg c\}$, and iteration number K;
Output: Reconstructed phase $\hat{\vartheta}^{(c')}(K)$ of source c', for c' in $\{c, \neg c\}$;
For $k = 1{:}\,K$ **do**
(1) $\hat{s}^{(c')}(k) = \text{iSTFT}(\hat{A}^{(c')}, \hat{\vartheta}^{(c')}(k-1))$, for c' in $\{c, \neg c\}$;
(2) $\varepsilon(k) = y - \sum_{c' \in \{c, \neg c\}} \hat{s}^{(c')}(k)$;
(3) $\hat{\vartheta}^{(c')}(k) = \angle\text{STFT}(\hat{s}^{(c')}(k) + \varepsilon(k)/2)$, for c' in $\{c, \neg c\}$;
End

Algorithm 5-1. Two-source MISI. $\text{iSTFT}(\cdot,\cdot)$ reconstructs a time-domain signal from a magnitude and a phase. $\text{STFT}(\cdot)$ computes the magnitude and phase of a signal.

phase wrapping is incurred [109]. In addition, good magnitude estimation is achievable as is indicated in recent advance on deep learning based speech separation [161].

5.2. Chimera++ Networks Revisit

For speaker separation, we need to first resolve the label-permutation problem. This section uses the chimera++ networks introduced in Chapter 4.3, which combine DC and PIT in a multi-task learning way, to resolve the permutation problem. The resulting masks obtained from the PIT branch are denoted as $\widehat{M}^{(c)}$ for each source.

In Chapter 4.3, a vanilla BLSTM is used in the chimera++ network. To improve mask estimation, we employ a BLSTM with convolutional encoder-decoder structures and skip connections [147] (see Figure 5-4).

5.3. Proposed Algorithms

With the label-permutation problem resolved, an enhancement network, which includes the estimated mask $\widehat{M}^{(c)}$ produced by the chimera++ network as inputs, is trained for each of the following three proposed algorithms to further estimate the magnitude of

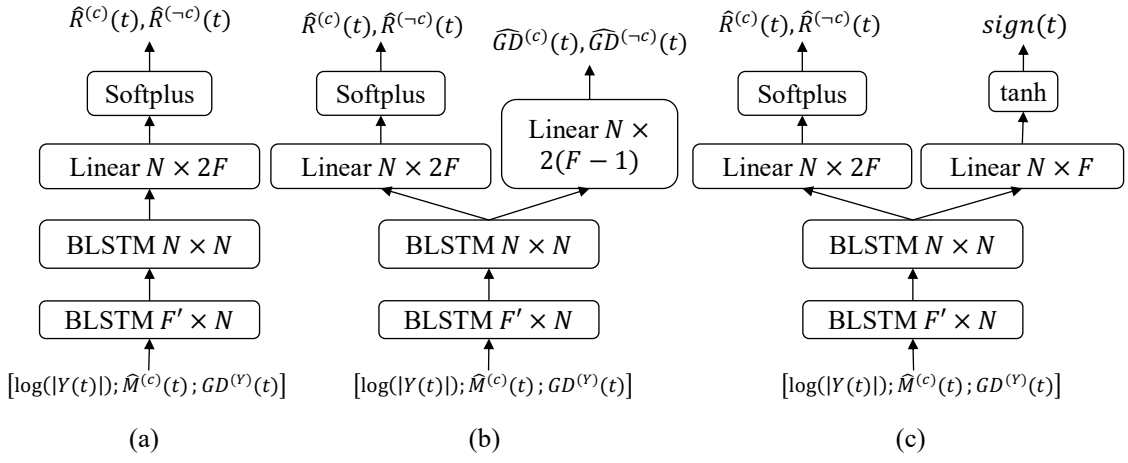


Figure 5-2. Enhancement network architectures. $GD^{(Y)}(t,f) = \angle e^{j(\angle Y(t,f+1) - \angle Y(t,f))}$.

source c and $\neg c$ for phase reconstruction. See Figure 5-2 for the network architectures. A side product of this research is a new way of computing the PSM using magnitude estimates (see Chapter 5.3.4).

5.3.1. Deep Learning Based Iterative Phase Reconstruction

One straightforward approach for phase reconstruction is to use estimated magnitudes to drive an iterative phase reconstruction algorithm [52], [215], [177], [181]. Here, we employ the multiple input spectrogram inverse (MISI) algorithm [46] (see Algorithm 5-1). Our insight is that the error distribution step (see step (2) and (3) in Algorithm 5-1) can ensure that the estimated phases are taken from reconstructed signals that add up to the mixture signal. The geometric constraint is hence roughly satisfied. If the magnitudes of the reconstructed signals are sufficiently accurate, the signs of many T-F units could be automatically determined, because the reconstructed signals are real signals that guarantee to have consistent phase structures and only particular ways of sign assignments exhibit consistent phase.

One issue with previous studies [177], [181] employing MISI for phase reconstruction is that the PSM is used as the training target in PIT and the resulting magnitude estimates are used for MISI. However, the sum of such magnitude estimates almost equals the mixture magnitude, as the sum of the PSMs of all the sources is one. Under the geometric constraint, the most reasonable phase estimate for each source is therefore simply the mixture phase. For example, in Figure 5-1(a), if $\hat{A}^{(1)}(t,f) + \hat{A}^{(2)}(t,f) = |Y(t,f)|$, the three sides cannot make a triangle and the absolute phase difference estimates $|\hat{\theta}^{(1)}(t,f) - \angle Y(t,f)|$ and $|\hat{\theta}^{(2)}(t,f) - \angle Y(t,f)|$ are both zero. Similar issues will be incurred if the sum of estimated magnitudes is implicitly or explicitly constrained to equal the mixture magnitude, such as using the IBM or IRM as the training target, using softmax as the output non-linearity, and estimating noise magnitude by subtracting estimated speech magnitude from the mixture magnitude.

This study hence estimates the SMM by using the magnitude spectrum approximation (MSA) loss function in Eq. (5.6), rather than the PSM using Eq. (5.7). See Figure 5-2(a) for the network structure. This minor change leads to large improvements in our experiments after MISI is applied for phase reconstruction.

$$\mathcal{L}_{MSA(\alpha)}^{Enh1} = \mathcal{L}_{MSA(\alpha)} = \sum_{c' \in \{c, \neg c\}} \left\| |Y| \otimes T_0^{\alpha}(\hat{R}^{(c')}) - T_0^{\alpha|Y|}(|S^{(c')}|) \right\|_1, \tag{5.6}$$

where $\hat{R}^{(c')}$ is the estimated SMM obtained by using softplus non-linearity. Based on the trigonometric perspective, α should be much larger than one so that the estimated magnitudes can be large enough compared with the mixture magnitude when necessary to elicit a large enough phase difference for phase reconstruction, such as when the sources cancel with each other at a T-F unit.

To facilitate comparison, we also train the same network with minimal changes to estimate the PSM using the following loss

$$\mathcal{L}_{PSA(\gamma,\beta)}^{Enh1} = \sum_{c'\in\{c,\neg c\}} \left\| |Y| \otimes T_{\gamma}^{\beta}(\hat{Q}^{(c')}) - T_{\gamma|Y|}^{\beta|Y|}\left(|S^{(c')}| \otimes \cos\left(\angle S^{(c')} - \angle Y\right)\right) \right\|_1, \quad (5.7)$$

where the estimated PSM $\hat{Q}^{(c')}$ is obtained by using sigmoid activation when $\beta = 1$ and $\gamma = 0$, linear activation when $\beta > 1$ and $\gamma < -1$, Softplus when $\beta > 1$ and $\gamma = 0$, and tanh when $\beta = 1$ and $\gamma = -1$.

Following [181], we unfold the MISI iterations as multiple layers in the network and compute the loss function in the time domain

$$\mathcal{L}_{MISI-K}^{Enh1} = \sum_{c'\in\{c,\neg c\}} \left\| \text{iSTFT}(\hat{A}^{(c')}, \hat{\vartheta}^{(c')}(K)) - s^{(c')} \right\|_1, \quad (5.8)$$

where $\hat{\vartheta}^{(c')}(K)$ denotes the reconstructed phase after K iterations of MISI (see Algorithm 5-1 for detailed definitions), which starts from estimated magnitude $\hat{A}^{(c')} = |Y| \otimes T_0^{\alpha}(\hat{R}^{(c')})$ and the mixture phase $\angle Y$.

5.3.2. Group Delay Based Phase Reconstruction

For a pair of T-F units at two consecutive frequencies, there are four (2^2) combinations of possible phase solutions, while only one combination exhibits a particular group delay. Our study first estimates the group delay of each source and then finds a sign assignment at each T-F unit in a way such that the resulting phase spectrum has a group delay closest to the estimated one. Note that group delay (GD) [110], computed as $GD^{(c)}(t,f) = \angle e^{j(\angle S^{(c)}(t,f+1) - \angle S^{(c)}(t,f))}$, exhibits patterns clearly predictable from (see Figure. 5-3), and is mathematically related to, log magnitude [109], [42], [111], [140].

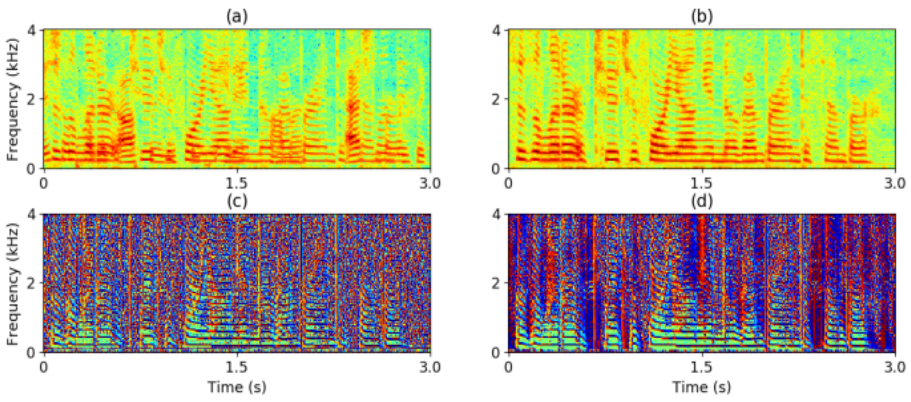


Figure. 5-3. Illustration of GD using a two-speaker mixture. (a) Log magnitude of mixture; (b) log magnitude of source 1; (c) clean GD of source 1; (d) estimated GD of source 1.

Figure 5-2(b) depicts the network structure. Magnitude weighted cosine distance is used as the loss function in the GD branch

$$\mathcal{L}_{GD1} = \sum_{c' \in \{c, \neg c\}} \sum_{t} \sum_{f=1}^{F-1} |S^{(c')}(t, f+1)| (1 - \cos(\widehat{GD}^{(c')}(t, f) - GD^{(c')}(t, f)))/2, \quad (5.9)$$

and the overall loss function is $\mathcal{L}_{MSA(\alpha)+GD1}^{Enh2} = \mathcal{L}_{MSA(\alpha)} + \mathcal{L}_{GD1}$.

At run time, assuming that $\hat{A}^{(c)}$, $\hat{A}^{(\neg c)}$ and $|Y|$ form a triangle at each T-F unit, we first estimate the absolute phase difference $\hat{\delta}^{(c')}$ between source c' and the mixture based on the law of cosines

$$\hat{\delta}^{(c')} = \left| \angle e^{j\left(\hat{\theta}^{(c')} - \angle Y\right)} \right| = \arccos\left(\mathcal{T}\left(\frac{|Y|^2 + \hat{A}^{(c')^2} - \hat{A}^{(\neg c')^2}}{2|Y| \otimes |\hat{A}^{(c')}|} \right) \right), \text{ for } c' \text{ in } \{c, \neg c\} \quad (5.10)$$

where $\mathcal{T}(\cdot)$ truncates the values outside of the range $[-1,1]$ to 1. Note that when $\hat{A}^{(c)}(t,f) + \hat{A}^{(\neg c)}(t,f) \leq |Y(t,f)|$, the three sides cannot make a triangle. This can happen as we are using estimated magnitudes. In addition, $\hat{A}^{(c)}$ and $\hat{A}^{(\neg c)}$ could have zero

values in some T-F units, if obtained via ReLU. We hence clip the values outside the range $[-1,1]$ to 1, meaning that the mixture phase is considered as the phase estimate for such T-F units since $\arccos(1) = 0$.

We then determine the sign assignment at each T-F unit, $\hat{g}(t,f) \in \{-1,1\}$, by maximizing the following similarity at each frame

$$\hat{g}(t,1), \dots, \hat{g}(t,F) = \underset{g(t,1),\dots,g(t,F)}{\operatorname{argmax}} \sum_{f=1}^{F-1} \sum_{c' \in \{c,\neg c\}} \cos\left(\hat{\theta}^{(c')}(t,f+1)(g(t,f+1)) - \hat{\theta}^{(c')}(t,f)(g(t,f)) - \widehat{GD}^{(c')}(t,f)\right), \tag{5.11}$$

where $\hat{\theta}^{(c)}(t,f)(g(t,f))$ and $\hat{\theta}^{(\neg c)}(t,f)(g(t,f))$are phases hypothesized as

$$\hat{\theta}^{(c)}(t,f)(g(t,f)) = \angle Y(t,f) + g(t,f)\hat{\delta}^{(c)}(t,f) \tag{5.12}$$

$$\hat{\theta}^{(\neg c)}(t,f)(g(t,f)) = \angle Y(t,f) - g(t,f)\hat{\delta}^{(\neg c)}(t,f) \tag{5.13}$$

Although Eq. (5.11) has 2^F possible solutions, our insight is that it can be efficiently solved with time complexity $O(2^2 F)$ by applying dynamic programming (or Viterbi decoding) within each frame, as the estimated GD only characterizes the phase relations between each consecutive T-F unit pair along frequency. The final phase estimates are obtained as $\angle Y + \hat{g} \otimes \hat{\delta}^{(c)}$ and $\angle Y - \hat{g} \otimes \hat{\delta}^{(\neg c)}$.

There are previous studies [107], [108] employing GD for sign determination. However, they resolve the ambiguity using an empirically hypothesized minimum GD deviation constraint and only consider a few frequencies with detected harmonic peaks.

5.3.3. Sign Prediction Networks

The GD based method is designed to be applied at run time as post processing. It is hard to perform end-to-end training. A possibly better approach is to let the network predict the sign explicitly (see Figure 5-2(c)), and compute the estimated phases as follows

$$\hat{\theta}^{(c)} = \angle Y + sign \otimes \hat{\delta}^{(c)} \tag{5.14}$$

$$\hat{\theta}^{(\neg c)} = \angle Y - sign \otimes \hat{\delta}^{(\neg c)} \tag{5.15}$$

where $sign$ is obtained via tanh non-linearity. Note that $\hat{\delta}^{(c')}$ is naturally bounded in the range $[0, \pi]$ and $sign \otimes \hat{\delta}^{(c')}$ in the range $[-\pi, \pi]$. The loss function on estimated phases is

$$\mathcal{L}_{GD2} = \sum_{c' \in \{c, \neg c\}} \sum_{t} \sum_{f=1}^{F-1} \left|S^{(c')}(t, f+1)\right| \left(1 - \cos\left(\hat{\theta}^{(c')}(t, f+1) - \hat{\theta}^{(c')}(t, f) - GD^{(c')}(t, f)\right)\right)/2, \tag{5.16}$$

and the overall loss function is: $\mathcal{L}^{Enh3}_{MSA(\alpha)+GD2} = \mathcal{L}_{MSA(\alpha)} + \mathcal{L}_{GD2}$. This way, the network could learn to produce a sign that can lead to GD spectrums close to the clean ones. An alternative is to compute the loss from the estimated phases and clean phases

$$\mathcal{L}_{phase} = \sum_{c' \in \{c, \neg c\}} \left\| |S^{(c')}| \otimes (1 - \cos(\hat{\theta}^{(c')} - \theta^{(c')}))/2 \right\|_1, \tag{5.17}$$

and the overall loss function is: $\mathcal{L}^{Enh3}_{MSA(\alpha)+phase} = \mathcal{L}_{MSA(\alpha)} + \mathcal{L}_{phase}$.

We emphasize that Eq. (5.14) and (5.15) (as well as (5.12) and (5.13)) implicitly constrain that, at each T-F unit, the two reconstructed STFT vectors ($\hat{A}^{(c)}(t, f) e^{j\hat{\theta}^{(c)}(t,f)}$ and $\hat{A}^{(\neg c)}(t, f) e^{j\hat{\theta}^{(\neg c)}(t,f)}$) have to be on the different sides of the mixture STFT vector $Y(t, f)$ in the complex plane, and $\hat{\theta}^{(c)}(t, f)$ and $\hat{\theta}^{(\neg c)}(t, f)$ cannot be, at the same time, more than $\pi/2$ away from $\angle Y(t, f)$, because only in this way could the two reconstructed

STFT vectors add up to the mixture STFT vector. This distinguishes our approach from studies that directly predict unbounded or unconstrained phase differences [1], [87], clean phases [145] and real and imaginary components of target sources [192], [193], or fully complex neural network approaches [130].

Following our recent study [181], we train through iSTFT for time-domain waveform approximation (WA), using $\hat{A}^{(c')}$ and $\hat{\theta}^{(c')}$

$$\mathcal{L}_{WA}^{Enh3} = \sum_{c' \in \{c, \neg c\}} \left\| \text{iSTFT}(\hat{A}^{(c')}, \hat{\theta}^{(c')}) - s^{(c')} \right\|_1 \tag{5.18}$$

Following [87], which uses estimated phases as the starting phases to train through MISI, we further train our model using

$$\mathcal{L}_{MISI-K}^{Enh3} = \sum_{c' \in \{c, \neg c\}} \left\| \text{iSTFT}(\hat{A}^{(c')}, \hat{\vartheta}^{(c')}(K)) - s^{(c')} \right\|_1, \tag{5.19}$$

where $\hat{\vartheta}^{(c')}(K)$ is obtained after K iterations of MISI starting from $\hat{A}^{(c')}$ and $\hat{\theta}^{(c')}$ produced by the sign prediction network. We will denote $\mathcal{L}_{WA}^{Enh_3}$ as $\mathcal{L}_{MISI-0}^{Enh_3}$, since $\hat{\vartheta}^{(c')}(0) = \hat{\theta}^{(c')}$ (see Algorithm 5-1).

Following [181], [117], [120], which computes loss using the magnitudes of reconstructed signals, we further train the network using

$$\mathcal{L}_{MISI-K-MSA}^{Enh3} = \sum_{c' \in \{c, \neg c\}} \left\| \left| \text{STFT}\left(\text{iSTFT}\left(\hat{A}^{(c')}, \hat{\vartheta}^{(c')}(K) \right) \right) \right| - \left| S^{(c')} \right| \right\|_1 \tag{5.20}$$

Our insight is that due to phase inconsistency, the reconstructed signal, $\text{iSTFT}(\hat{A}^{(c')}, \hat{\vartheta}^{(c')}(K))$, may not exhibit a magnitude as good as $\hat{A}^{(c')}$, although the iterative process in MISI can reduce their difference [44]. The network trained this way outputs two signals that almost add up to the mixture signal and each signal is expected to

exhibit a good magnitude. From the trigonometric perspective, the signs could be automatically determined because the two signals are real signals having consistent phase structures, as is explained in the first paragraph of Chapter 5.3.1.

5.3.4. Computing PSM from Estimated Magnitudes

A side product of this research is a new way of computing the PSM (defined as $|S^{(c)}| \otimes \cos(\angle S^{(c)} - \angle Y) / |Y|$) [35] in two-source cases, where the cosine term can be estimated as $\cos(\hat{\delta}^{(c)})$

$$\hat{Z}^{(c)} = \hat{A}^{(c)} \otimes \cos(\hat{\delta}^{(c)}) / |Y| \tag{5.21}$$

In the literature, the PSM is typically clipped to the range $[0,1]$ and directly predicted by a DNN in a way similar to Eq. (4.5) or using $\mathcal{L}_{PSA(0,1)}^{Enh1}$ (i.e. β=1 and γ=0) in Eq. (5.7) [35]. In contrast, the estimated PSM obtained here is assembled based on estimated magnitudes. It is not limited to the range $[0,1]$ and can even go negative.

5.4. Experimental Setup

We validate our algorithms on the wsj0-2mix and 3mix dataset [57], designed for a talker-independent speaker separation task. Each of them contains 20,000, 5,000, and 3,000 2(or 3)-speaker mixtures in its 30, 10 and 5 h training, validation, and test (open speaker condition, OSC) set, respectively. The sampling rate is 8 kHz. The SNR in each mixture is randomly sampled from -5 to 5 dB. We use 32 ms window size and 8 ms hop size. Square-root Hann window is applied before 256-point DFT is applied to extract 129-dimensional log magnitude features. λ in Eq. (4.6) is set to 0.975 and embedding dimension D set to 20. K in MISI is set to 5.

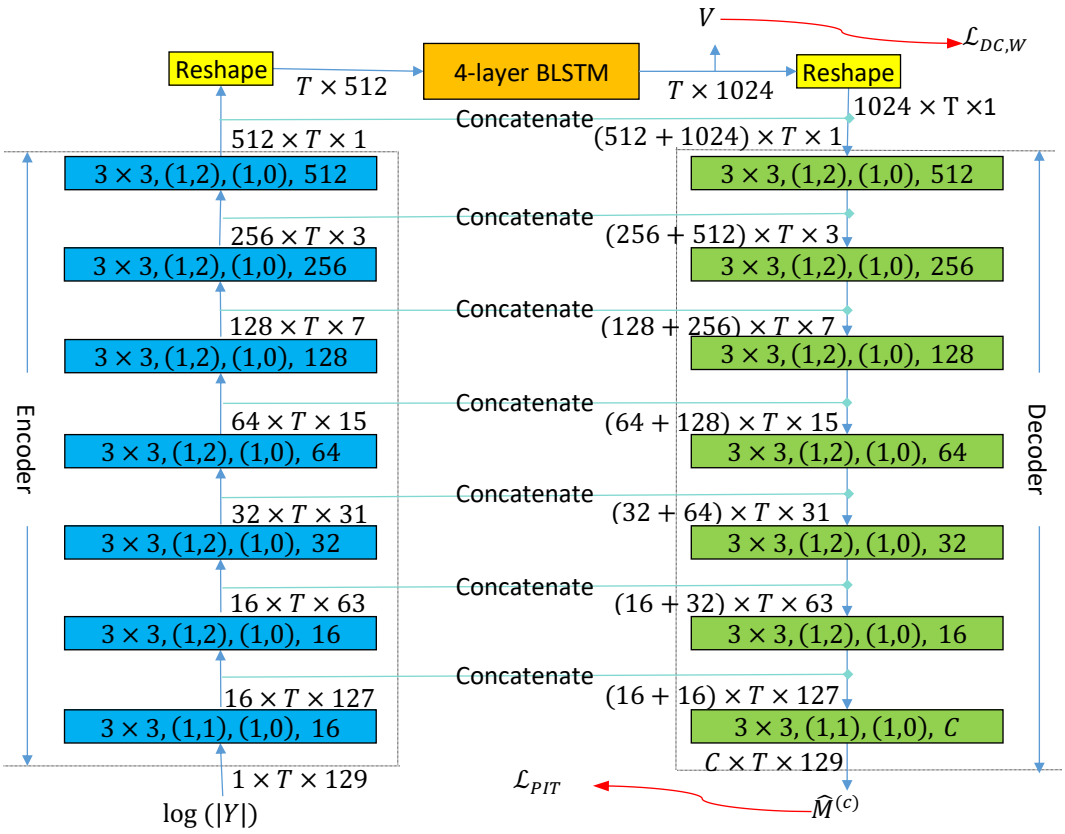


Figure 5-4. Chimera++ network architecture. The tensor shape after each block is in format: *featureMaps* × *timeSteps* × *frequencyChannels*. Each block is specified in the format: *kernelSizeTime* × *kernelSizeFreq*, *(stridesTime, stridesFreq), (paddingsTime, paddingsFreq), featureMaps*.

We use a 4-layer BLSTM with convolutional encoder-decoder structures and skip connections [126], [71] in the chimera++ network (see Figure 5-4). Similar network was found useful in a speech enhancement study [147]. The encoder contains seven convolutional blocks, each including 2D convolution, batch normalization and exponential linear units (ELU). The decoder contains six deconvolutional blocks, each consisting of 2D deconvolution, BN and ELU layers, and one 2D deconvolution layer and a sigmoidal layer to obtain estimated masks. The embedding layer grows out from the last BLSTM layer. Each BLSTM has 512 units in each direction.

Table 5-1. Average SI-SDRi (dB) and PESQ results on OSC of wsj0-2mix.

Approaches	Models	Enhanced Phase?	SI-SDRi	PESQ
Unprocessed	-	No	0.0	2.01
Chimera++(Encoder-BLSTM-Decoder)	$\mathcal{L}_{chi++}$	No	11.9	3.12
Deep learning based iterative phase reconstruction	$\mathcal{L}^{Enh1}_{PSA(0,1)}$	No	12.1	3.15
	+MISI-5	Yes	12.5	3.17
	$\mathcal{L}^{Enh1}_{PSA(0,5)}$	No	12.4	3.17
	+MISI-5	Yes	12.9	3.19
	$\mathcal{L}^{Enh1}_{PSA(-1,1)}$	No	12.4	3.21
	+MISI-5	Yes	12.9	3.24
	$\mathcal{L}^{Enh1}_{PSA(-5,5)}$	No	12.7	3.21
	+MISI-5	Yes	13.3	3.24
	$\mathcal{L}^{Enh1}_{MSA(5)}$	No	11.1	3.27
	+MISI-5	Yes	14.4	3.43
	$+\mathcal{L}^{Enh1}_{MISI-5}$	Yes	15.0	3.38
	+Eq. (5.21)	No	12.6	3.24
Group delay based phase reconstruction	$\mathcal{L}^{Enh2}_{MSA(5)+GD1}$	Yes	13.6	3.39
Sign prediction network	$\mathcal{L}^{Enh3}_{MSA(5)+GD2}$	Yes	14.2	3.39
	$\mathcal{L}^{Enh3}_{MSA(5)+phase}$	Yes	14.4	3.38
	+MISI-5	Yes	15.0	3.44
	$+\mathcal{L}^{Enh3}_{WA}$	Yes	14.6	3.36
	$+\mathcal{L}^{Enh3}_{MISI-5}$	Yes	15.3	3.36
	$+\mathcal{L}^{Enh3}_{MISI-5-MSA}$	Yes	15.2	3.45

Each enhancement network (see Figure 5-2) contains three BLSTM layers, each with 600 units in each direction.

We use scale-invariant SDR improvement (SI-SDRi) [94] as the major evaluation metric. We also report SDR improvement (SDRi) and PESQ.

5.5. Evaluation Results

Table 5-1 reports the performance on wsj0-2mix. Including the encoder-decoder structure into the chimera++ network improves SI-SDRi by 0.7 dB (from 11.2 to 11.9 dB), compared with [177] that uses a vanilla BLSTM. The enhancement network, which can also be thought of as stacking [176], [81], improves estimated PSM results from 11.9 to

12.1 dB, by using $\mathcal{L}_{PSA(0,1)}^{Enh1}$. Further applying 5 iterations of MISI (MISI-5) at run time only leads to slight improvement (from 12.1 to 12.5 dB). Similar trend is observed for models trained using $\mathcal{L}_{PSA(0,5)}^{Enh1}$, $\mathcal{L}_{PSA(-1,1)}^{Enh1}$, and $\mathcal{L}_{PSA(-5,5)}^{Enh1}$. In contrast, the model trained to estimate the SMM using $\mathcal{L}_{MSA(5)}^{Enh1}$ (i.e. α=5) exhibits substantial improvements when combined with MISI-5 (from 11.1 to 14.4 dB), indicating that the SMM is the preferred training target if MISI needs to be performed. Further training the model with $\mathcal{L}_{MISI-5}^{Enh1}$ pushes the performance to 15.0 dB. Compared with $\mathcal{L}_{MSA(5)}^{Enh1}$, using estimated group delay from $\mathcal{L}_{MSA(5)+GD1}^{Enh2}$ for phase reconstruction improves the performance from 11.1 to 13.6 dB, while this approach is not as good as the sign prediction networks that can be trained end-to-end. Compared with $\mathcal{L}_{MSA(5)}^{Enh1}$, $\mathcal{L}_{MSA(5)+phase}^{Enh3}$ and $\mathcal{L}_{MSA(5)+GD2}^{Enh3}$ both lead to substantial improvement (14.4 and 14.2 vs. 11.1 dB). The former is slightly better, likely because it directly compares estimated phases with clean ones for loss computation. Further applying MISI-5 on the estimated magnitudes and enhanced phase improves the results to 15.0 dB, which is 0.6 dB (15.0 vs. 14.4 dB) better than applying MISI-5 on the model trained with $\mathcal{L}_{MSA(5)}^{Enh1}$, indicating the benefits of using an enhanced phase as the starting phase for MISI over using the mixture phase. Further training through MISI using $\mathcal{L}_{MISI-5}^{Enh3}$ produces slight improvement (from 15.0 to 15.3 dB). Compared with $\mathcal{L}_{MISI-5}^{Enh3}$, $\mathcal{L}_{MISI-5-MSA}^{Enh3}$ leads to worse SI-SDRi (15.2 vs. 15.3 dB), which aligns with the findings in [181]. Different from [181], the PESQ score is improved significantly from 3.36 to 3.45. This could be that PESQ is computed by reducing the phase mismatch between enhanced signals and reference signals via a time alignment procedure, and considerably taking into account the magnitudes of resynthesized signals [123], while SI-SDR solely considers

Table 5-2. Average SI-SDRi (dB), SDRi (dB) and PESQ comparison between proposed algorithms and other methods on OSC of wsj0-2mix and wsj0-3mix.

Approaches	wsj0-2mix			wsj0-3mix		
	SI-SDRi	SDRi	PESQ	SI-SDRi	SDRi	PESQ
Unprocessed	0.0	0.0	2.01	0.0	0.0	1.66
DC++ [57], [69]	10.8	-	-	7.1	-	-
ADANet [94]	10.4	10.8	2.82	9.1	9.4	2.16
uPIT-ST [206], [81]	-	10.0	-	-	7.7	-
Chimera++ (BLSTM) [177]	11.2	11.5	-	-	-	-
+MISI-5 [177]	11.5	11.8	-	-	-	-
+WA-MISI-5 [181]	12.6	12.9	-	-	-	-
+ PhaseBook [87]	12.8	-	-	-	-	-
conv-TasNet-gLN [97], [95]	14.6	15.0	3.25	11.6	12.0	2.50
Proposed (Sign prediction net, $\mathcal{L}_{MISI-5}^{Enh3}$)	**15.3**	**15.6**	3.36	**12.1**	**12.5**	2.64
Proposed (Sign prediction net, $\mathcal{L}_{MISI-5-MSA}^{Enh3}$)	15.2	15.4	**3.45**	12.0	12.3	**2.77**

time-domain signals and is hence more sensitive to phase mismatches. For the side product in Eq. (5.21), which assembles an estimated PSM from the estimated magnitudes produced via $\mathcal{L}_{MSA(5)}^{Enh1}$, it obtains results comparable to $\mathcal{L}_{PSA(-5,5)}^{Enh1}$, and better than the other three models trained to directly estimate the PSM.

Table 5-2 compares the performance of our algorithm with other competitive systems on the wsj0-2mix and 3mix corpus. Our algorithm obtains dramatically better performance than the other STFT based approaches. Its performance is also better than a recent time-domain approach [95], particularly in terms of PESQ.

5.6. Conclusion

Thanks to a novel trigonometric perspective, we have proposed three phase reconstruction algorithms based on magnitude estimation. The obtained state-of-the-art speaker separation results at the time of publication suggest that deep learning based magnitude estimation can clearly benefit phase reconstruction. In closing, we emphasize

that a geometric constraint affords a mechanism to narrow down the possible solutions of phase, and it could play a fundamental role in future research on phase estimation.

Chapter 6. Multi-Channel Speech Dereverberation

This chapter investigates multi-channel speech dereverberation and its application to robust ASR in reverberant conditions using deep learning based complex spectral mapping. The work in this chapter has been published in IEEE/ACM T-ASLP in 2020 [185].

6.1. Introduction

Room reverberation is pervasive in modern hands-free speech communication. In a reverberant enclosure, speech signals propagate in the air and are inevitably reflected by the walls, ceiling, floor, and any objects in the room. As a result, the signal captured by a distant microphone is a summation of an infinite number of delayed and decayed copies of original source signals. Room reverberation is known to be detrimental to ASR systems, and severely degrades speech quality and intelligibility. Speech dereverberation is a challenging task, as reverberation is a convolutive interference, and it is difficult to distinguish the direct-path signal from its reverberated versions, especially when room reverberation is strong or environmental noise is also present [161].

For single-channel dereverberation, one conventional approach estimates the power spectral density (PSD) of late reverberation to compute a Wiener-like filter [48], [10]. The weighted prediction error (WPE) algorithm [112], [205] is probably the most widely used algorithm for speech dereverberation. It uses variance-normalized delayed linear prediction

to predict late reverberation from past observations, and subtracts the predicted reverberation to estimate target speech. It iteratively estimates the time-varying PSD of target speech and the linear filter, and is unsupervised in nature. Many ASR studies report that WPE suppresses reverberation with low speech distortions, and consistently improves ASR performance even for multi-conditionally trained ASR backends [26].

When multiple microphones are available, spatial information can be leveraged to filter out signals not arriving from the estimated target direction. Single-channel WPE can be extended to multi-channel WPE [112] by simply concatenating the observations across multiple microphones when performing linear prediction. Another popular approach for multi-channel speech dereverberation is the so-called suppression approach [49], [11], which decomposes a multi-channel Wiener filter into a product of a time-invariant or time-varying MVDR beamformer and a monaural Wiener post-filter. This approach can utilize the phase produced by linear beamforming, which is expected to be better than the mixture phase, since MVDR beamforming is distortionless. However, the phase improvement is dependent on linear beamforming, which is less effective when room reverberation is strong or when the number of microphones is small. In addition, the Wiener post-filter is a real-valued mask, and would inevitably introduce phase inconsistency problems [44], [184], when directly applied to the beamformed signal for enhancement.

Different from conventional algorithms, supervised learning based approaches train a DNN to predict the magnitudes or real-valued masks of the direct-path signal from reverberant observations [37], [52], [100], [104], [197]. However, the DNN operates in the magnitude domain, and mixture phase is typically utilized for signal re-synthesis. Phase estimation is hence a promising direction for further improvement. Another direction in

dereverberation uses DNN estimated speech magnitudes as the PSD estimate for WPE [59], [60], [76], [143]. This approach can leverage the spectral structure in speech for linear prediction, and most importantly eliminates the iterative process. In offline scenarios, although ASR improvement is observed on the eight-channel task of the REVERB challenge, it leads to slightly worse performance on the monaural task [76].

In this context, our study extends magnitude-domain masking and mapping based speech dereverberation to the complex domain, where a DNN is trained to predict the RI components of direct sound from reverberant ones. Although previous studies perform single-channel complex masking or mapping for speech denoising [39], [148], [192], their results in reverberant conditions are mixed [193] and how to extend to multi-channel processing is unclear.

Our study approaches multi-channel dereverberation from the angle of target cancellation, where a key assumption is that the target speaker is a directional source, and is typically non-moving within a short utterance. This suggests that we can point a null beam to cancel the target speaker, and the remaining signal would only contain a filtered version of reverberation. This filtered reverberation can be utilized as extra features for DNN to perform multi-channel complex spectral mapping based dereverberation. It should be noted that similar ideas of target cancellation were explored in binaural speech segregation [125] and multi-channel dereverberation [79], [11]. Their purposes are, however, different (e.g. on the PSD estimation of late reverberation), and they do not address phase estimation.

Our study makes four main contributions. First, we extend deep learning based magnitude-domain single-channel speech dereverberation to the complex domain for phase

estimation. The phase estimation method follows the complex spectral mapping idea in [39], [148], while our study addresses direct sound vs. reverberation and noise, rather than speech vs. noise in anechoic conditions. Second, we introduce for complex spectral mapping a magnitude-domain loss function, which dramatically improves speech quality measures in reverberant conditions. Third, we propose a novel target cancellation strategy to utilize spatial information to improve the estimation of direct sound. Fourth, we investigate the effectiveness of DNN based phase estimation for beamforming and post-filtering, while the DNN in previous deep learning based multi-channel enhancement operates in the magnitude domain.

We emphasize that the proposed algorithms are designed in a way such that the resulting models, once trained, can be directly applied to arrays with an arbitrary number of microphones arranged in an unknown geometry.

The rest of this paper is organized as follows. We introduce the physical model in Chapter 6.2. The proposed algorithms are detailed in Chapter 6.3, followed by experimental setup in Chapter 6.4. Evaluation and comparison results are presented in Chapter 6.5. Conclusions are made in Chapter 6.6.

6.2. Physical Models and Objectives

Given a P -microphone time-domain signal $\boldsymbol{y}[n] = \left[y_1[n], \dots, y_P[n]\right]^{\mathrm{T}} \in \mathbb{R}^{P\times 1}$ recorded in a reverberant and noisy enclosure, the physical model in the STFT domain is formulated as:

$$\boldsymbol{Y}(t,f) = \boldsymbol{c}(f;q)S_q(t,f) + \boldsymbol{H}(t,f) + \boldsymbol{N}(t,f) = \boldsymbol{S}(t,f) + \boldsymbol{V}(t,f), \tag{6.1}$$

where $S_q(t,f) \in \mathbb{C}$ is the complex STFT coefficient of the direct-path signal of the target speaker captured by a reference microphone q at time t and frequency f, $\boldsymbol{c}(f;q) \in \mathbb{C}^{P\times 1}$ is the relative transfer function with the q^{th} element being one, and $\boldsymbol{c}(f;q)S_q(t,f)$, $\boldsymbol{H}(t,f)$, $\boldsymbol{N}(t,f)$ and $\boldsymbol{Y}(t,f) \in \mathbb{C}^{P\times 1}$ respectively represent the STFT vectors of the direct-path signal, reverberation, reverberant noise and received mixture at a T-F unit.

We propose multiple deep learning algorithms to enhance the mixture Y_q captured at the reference microphone q to recover S_q, by exploiting single- and multi-channel information contained in $\boldsymbol{Y}$. In this study, $\boldsymbol{N}(t,f)$ is assumed to be a quasi-stationary air-conditioning noise, as our focus is on dereverberation; the proposed algorithms can be straightforwardly applied to deal with more noises. The target speaker is assumed to be still within an utterance. Our study also assumes offline scenarios: we normalize the time-domain sample variance of each input multi-channel signal $\boldsymbol{y}$ to one before any processing. This normalization would be important for mapping-based enhancement to deal with random gains in input signals.

In the remainder of this paper, we refer to $\boldsymbol{S}(t,f) = \boldsymbol{c}(f;q)S_q(t,f)$ as the target component we aim to extract, and $\boldsymbol{V}(t,f) = \boldsymbol{H}(t,f) + \boldsymbol{N}(t,f)$ as the non-target component to remove.

6.3. Proposed Algorithms

There are two DNNs in our system. The first DNN performs single-channel dereverberation by predicting the RI components of the direct-path signal from a mixture. Dereverberation results are utilized to compute an MVDR beamformer. The second DNN utilizes the RI components of beamformed speech as additional features to further improve

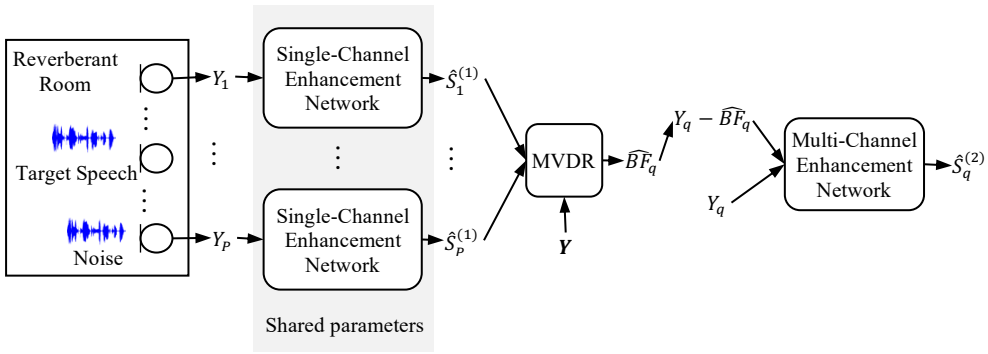


Figure 6-1. Illustration of overall system for single- and multi-channel speech dereverberation (or enhancement). There are two DNNs, one for single-channel and the other for multi-channel dereverberation and denoising. The superscript in $\hat{S}_1^{(1)}, \ldots, \hat{S}_P^{(1)}$ and $\hat{S}_q^{(2)}$ denotes the DNN used for processing.

the estimation of the RI components of the direct-path signal. Figure 6-1 illustrates the overall system.

6.3.1. Monaural Complex Spectral Mapping

Following recent studies [39], [148], we train a DNN to directly predict the RI components of the direct sound from reverberant and noisy ones. One key difference is that [39] and [148] deal with speech vs. noise, while our study addresses direct sound vs. reverberation and noise. We use the following loss function

$$\mathcal{L}_{\text{RI}} = \left\| \hat{R}_p - \text{Real}(S_p) \right\|_1 + \left\| \hat{I}_p - \text{Imag}(S_p) \right\|_1, \tag{6.2}$$

where $p \in \{1, \ldots, P\}$ indexes microphones, $\hat{R}_p$ and $\hat{I}_p$ are the estimated RI components obtained by using linear activation in the output layer, and Real(·) and Imag(·) respectively extract the RI components. The enhanced speech at microphone p is obtained

as $\hat{S}_p^{(k)} = \hat{R}_p^{(k)} + j\hat{I}_p^{(k)}$, where the superscript $k \in \{1,2\}$ denotes the output from the k^{th} DNN, as shown in Figure 6-1.

Following recent studies combining $\mathcal{L}_{\text{RI}}$ with a magnitude-domain loss [39], [194], we design the following loss function

$$\mathcal{L}_{\text{RI+Mag}} = \mathcal{L}_{\text{RI}} + \left\| \sqrt{\hat{R}_p^{\,2} + \hat{I}_p^{\,2}} - |S_p| \right\|_1 \tag{6.3}$$

Different from [39], [194], our study does not compress the estimated magnitudes or complex spectra using logarithmic or power compression. This way, the DNN is always trained to estimate a complex spectrum that has consistent magnitude and phase structures, and therefore would likely produce a consistent estimated STFT spectrum at run time [184].

Our experiments show that including a loss on magnitude leads to large improvements in objective measures of speech quality, along with a small degradation on time-domain SNR based measures, compared with only using $\mathcal{L}_{\text{RI}}$.

6.3.2. Multi-Channel Complex Spectral Mapping

We propose a target cancellation approach to exploit spatial information for dereverberation. The motivation is that given an oracle MVDR beamformer $\boldsymbol{w}(f;q)$, the beamformed signal is distortion-less, meaning that $S_q(t,f) = \boldsymbol{w}(f;q)^H \boldsymbol{S}(t,f)$. Therefore, the difference between the mixture and the beamformed signal at reference microphone q, computed as

$$Y_q(t,f) - BF_q(t,f)$$
$$= Y_q(t,f) - \boldsymbol{w}(f;q)^H \boldsymbol{Y}(t,f)$$

$$= S_q(t,f) + V_q(t,f) - \left(\boldsymbol{w}(f;q)^H \boldsymbol{S}(t,f) + \boldsymbol{w}(f;q)^H \boldsymbol{V}(t,f)\right)$$

$$= V_q(t,f) - \boldsymbol{w}(f;q)^H \boldsymbol{V}(t,f) \tag{6.4}$$

would only contain a filtered version of non-target signals, i.e. $V_q(t,f) - \boldsymbol{w}(f;q)^H \boldsymbol{V}(t,f)$. Intuitively, the more microphones there are and the more accurate the beamformer is, the weaker the beamformed non-target speech $\boldsymbol{w}(f;q)^H \boldsymbol{V}(t,f)$ would be, and the closer $V_q(t,f) - \boldsymbol{w}(f;q)^H \boldsymbol{V}(t,f)$ is to the actual non-target speech $V_q(t,f)$ we aim to remove at microphone q. This makes $Y_q - \widehat{BF_q}$ a highly discriminative feature for dereverberation, and hence motivates us to use it as an extra input for DNN to predict S_q via complex spectral mapping. Without this feature, the DNN may struggle at distinguishing direct-path signal from its reverberated versions, as the latter is a summation of the delayed and decayed copies of the former.

We apply the single-channel complex spectral mapping model to each microphone signal and directly use the estimated speech $\widehat{\boldsymbol{S}}^{(1)}$ to robustly compute an MVDR beamformer for cancelling target speech. Our study considers time-invariant MVDR (TI-MVDR) beamforming, as the target speaker is assumed still within each utterance, and reverberation and the considered noise are largely diffuse. The covariance matrices are computed as

$$\widehat{\Phi}^{(s)}(f) = \frac{1}{T}\sum_t \widehat{\boldsymbol{S}}(t,f)\, \widehat{\boldsymbol{S}}(t,f)^H \tag{6.5}$$

$$\widehat{\Phi}^{(v)}(f) = \frac{1}{T}\sum_t \widehat{\boldsymbol{V}}(t,f)\widehat{\boldsymbol{V}}(t,f)^H \tag{6.6}$$

where $\widehat{\boldsymbol{V}}(t,f) = \boldsymbol{Y}(t,f) - \widehat{\boldsymbol{S}}(t,f)$. The motivation is that the estimated complex spectra are expected to have cleaner phase than the mixture phase. In contrast, mask-weighted

ways of computing covariance matrices (see Eq. (6.11) for example) [36], [58], [200], [203], [213] are fundamentally limited when there are insufficient T-F units dominated by the direct-path signal, such as when room reverberation or environmental noise is very strong.

The relative transfer function is then computed in the following way

$$\hat{\boldsymbol{r}}(f) = \mathcal{P}\{\widehat{\Phi}^{(s)}(f)\} \tag{6.7}$$

$$\hat{\boldsymbol{c}}(f;q) = \hat{\boldsymbol{r}}(f)/\hat{r}_q(f) \tag{6.8}$$

where $\mathcal{P}\{\cdot\}$ extracts the principal eigenvector. The motivation is that $\widehat{\Phi}^{(s)}(f)$ would be close to a rank-one matrix if accurately estimated. Its principal eigenvector is therefore a reasonable estimate of the steering vector [40]. We then use Eq. (6.8) to obtain an estimated transfer function relative to a reference microphone q. We emphasize that, without using Eq. (6.8), a different complex gain would be introduced at each frequency, leading to distortions in the beamformed signal.

A TI-MVDR beamformer is then computed as

$$\widehat{\boldsymbol{w}}(f;q) = \frac{\widehat{\Phi}^{(v)}(f)^{-1}\hat{\boldsymbol{c}}(f;q)}{\hat{\boldsymbol{c}}(f;q)^{H}\widehat{\Phi}^{(v)}(f)^{-1}\hat{\boldsymbol{c}}(f;q)} \tag{6.9}$$

The beamformed signal is obtained using

$$\widehat{BF}_q(t,f) = \widehat{\boldsymbol{w}}(f;q)^{H}\boldsymbol{Y}(t,f) \tag{6.10}$$

For multi-channel dereverberation, we feed the RI components of $Y_q - \widehat{BF}_q$, in addition to the RI components of Y_q, to a DNN to estimate the RI components of the direct-path signal S_q (see Figure 6-1).

We point out that this strategy is in spirit similar to the classic generalized sidelobe canceller [40], which contains three components: a delay-and-sum (DAS) beamformer computed to enhance the target signal, a blocking matrix used to block the target signal, and an adaptive noise canceller designed to cancel the sidelobes produced by the DAS beamformer based on the blocked signal. The key difference here is that we compute an MVDR beamformer to block the target signal, and use deep learning to cancel the non-target signal in Y_q based on $Y_q - \widehat{BF}_q$.

From the spatial feature perspective, popular for deep learning based multi-channel speech enhancement [4], [73], [118], [214] and speaker separation [180], the RI components of $\widehat{BF}_q$ or $Y_q - \widehat{BF}_q$ can be considered as complex-domain spatial features, which can be utilized by the DNN to extract a target speech signal with specific spectral structure and arriving from a particular direction. Such features are more general than those previously proposed for improving magnitude estimation, such as plain IPD [204], cosine and sine IPD [178], and target direction compensated IPD and the magnitudes of beamformed mixtures [180].

6.4. Experimental Setup

Our models for dereverberation are trained on reverberant and noisy data created by using simulated RIRs and recorded noises. We first measure the performance on a relatively matched simulated test set, and then evaluate the trained models directly on the test set of the REVERB challenge [77] to show their generalization ability. This section describes the datasets and the setup for model training, and several baseline systems for comparison purposes.

Input: WSJCAM0;
Output: spatialized reverberant (and noisy) WSJCAM0;
REP[train]=5; *REP[validation]*=4; *REP[test]*=3;
For *dataset* in {*train*, *validation*, *test*} set of WSJCAM0 **do**
  For each anechoic speech signal s in *dataset* **do**
  Repeat *REP[dataset]* times **do**
    - Sample room length r_x and width r_y from [5,10] m;
    - Sample room height r_z from [3,4] m;
    - Sample mic array height a_z from [1,2] m;
    - Sample array displacement n_x and n_y from $[-0.5,0.5]$ m;
    - Place array center at $\langle \frac{r_x}{2} + n_x, \frac{r_y}{2} + n_y, a_z \rangle$ m;
    - Sample array radius a_r from [0.03,0.1] m;
    - Sample angle of first mic angle ϑ from $[0, \frac{\pi}{4}]$;
    For $p = 1 : P(= 8)$ **do**
    - Place mic p at $\langle \frac{r_x}{2} + n_x + a_r \cos\left(\vartheta + (p-1)\frac{\pi}{4}\right), \frac{r_y}{2} + n_y + a_r \sin\left(\vartheta + (p-1)\frac{\pi}{4}\right), a_z \rangle$ m;
    End
    - Sample target speaker locations in the $0 - 360^{\circ}$ plane:
    $$\langle s_x, s_y, s_z(= a_z) \rangle$$
    such that the distance from target speaker to array center is in between [0.75,2.5] m, and target speaker is at least 0.5 m from each wall;
    - Sample T60 from [0.2,1.3] s;
    - Generate multi-channel impulse responses using RIR generator and convolve them with *s*;
    If *dataset* in {*train*, *validation*} do
      - Sample a *P*-channel noise signal n from the training noise of REVERB corpus;
    Else
      - Sample a *P*-channel noise signal n from the testing noise of REVERB corpus;
    End
    - Concatenate channels of reverberated *s* and n respectively, scale them to an SNR randomly sampled from [5,25] dB, and add them to obtain reverberant and noisy mixture;
  End
  End
End

Algorithm 6-1. Data spatialization process (simulated RIRs).

6.4.1. Datasets and Evaluation Setup

Following REVERB [77], our training data for dereverberation is generated using the WSJCAM0 corpus. Different from REVERB, which only uses 24 measured eight-channel RIRs to generate its training data, we use a much larger set of RIRs (in total 39,305 eight-channel RIRs for training) generated by an RIR generator [47] to simulate room reverberation. See Algorithm 6-1 for the detailed simulation procedure. For each utterance,

we randomly generate a room with different room characteristics, speaker and microphone locations, microphone array characteristics, and noise levels. Our study considers eight-channel circular arrays with radius spanning from 3 to 10 cm. The target speaker is placed on the same plane as the array, at a distance randomly drawn from 0.75 to 2.5 m. The reverberation time (T60) is randomly sampled between 0.2 and 1.3 s. We use the training and test noise (mostly diffuse quasi-stationary fan noise) in REVERB to simulate noisy reverberant mixtures in our training and test sets, respectively. The SNR between the direct sound and reverberant noise of each mixture is randomly drawn between 5 and 25 dB. The average DRR is -3.7 dB with 4.4 dB standard deviation. There are 39,305 (7,861×5, ~80 h), 2,968 (742×4, ~6 h), and 3,264 (1,088×3, ~7 h) eight-channel utterances in the training, validation and test set, respectively. Note that the training and the test speakers are different. We denote this test set as **Test Set I**. At run time, we randomly choose a subset of microphones for each test mixture for evaluation. This setup therefore covers a wide range of microphone geometry. We use the direct-path signal at a reference microphone (i.e. the signal corresponding to S_q) as the reference for metric computation, and the first microphone is always considered as the reference. For P-channel processing, we randomly select $P-1$ microphones from the non-reference microphones and always report the performance on the reference microphone. This way, we can directly compare single- and multi-channel processing as they are both evaluated using the same reference signals.

We apply the trained models, without re-training, to the test set of REVERB, which contains simulated as well as recorded reverberant and noisy mixtures. We first evaluate the enhancement performance of the trained models on the simulated test set (denoted as **Test Set II**), where six measured eight-channel RIRs are used to simulate 2,176 reverberant

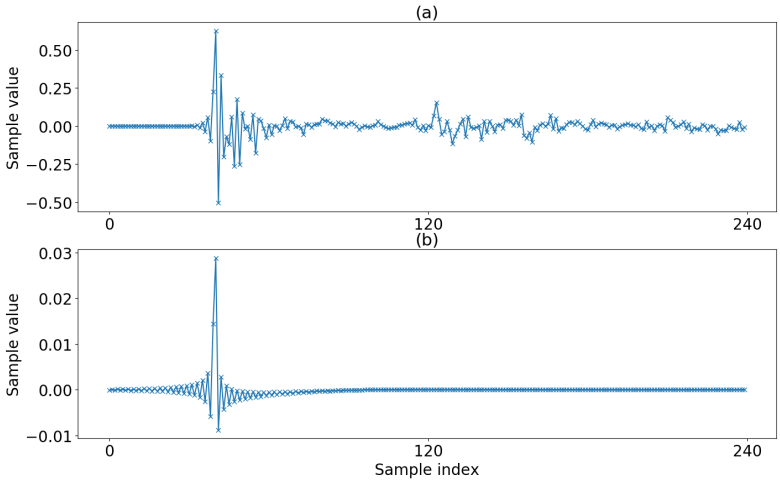


Figure 6-2. RIR illustration. (a) Example RIR segment from REVERB (*RIR_SimRoom3_far_AnglB.wav*); (b) Example direct-path RIR simulated using RIR generator.

and noisy mixtures. The six RIRs are measured in small-, medium- and large-size rooms, where the T60s are 0.25, 0.5 and 0.7 s respectively, and the speaker to microphone distance is around 0.5 m in the near-field case and 2.0 m in the far-field case. Recorded environmental noise is added at an SNR of 20 dB. In the REVERB challenge setup, only the sample at n_d, which is the index corresponding to the highest value in the measured RIR, is used to compute the direct-path signal (i.e. reference signal) for metric computation. However, due to measurement inaccuracy, this may not be realistic, since the samples in a small window around n_d are typically considered as in the direct-path RIR [34]. A short segment of an example RIR from REVERB is shown in Figure 6-2(a), where T60 is around 0.7 s. If we only use the sample at n_d to simulate the direct-path signal, the resulting DRR would be unrealistically low, as the samples around the peak exhibit non-negligible energy; as a result, the reverberation generated by the surrounding samples would be difficult to remove. These surrounding samples should be considered when computing the direct-path

signal, as they are in a measured RIR. Also, the sound source may not be a point source strictly and for a 16 kHz sampling rate, one discrete sample can have around $340/16{,}000$ m measurement error, where 340 (m/s) is the sound speed in the air. Furthermore, simulated direct-path RIRs are usually computed based on low-pass filtering, and they will be similar to a Sinc function even for a point source [47]. In Figure 6-2(b) we show an example direct-path RIR simulated using the RIR generator by setting the T60 parameter to zero. In our study, we hence use the samples in the range $[n_d - 0.0025 \times 16{,}000, n_d + 0.0025 \times 16{,}000]$ (i.e. a 5-ms window surrounding the peak) of the measured RIRs to compute the direct-path signal for metric computation. This strategy aligns with the setup in the ACE challenge [34]. We then evaluate the dereverberation models on the ASR task of REVERB (denoted as **REVERB ASR**). The test utterances are real recordings with T60 (reverberation time) around 0.7 s and the speaker to microphone distances approximately 1 m in the near-field case and 2.5 m in the far-field case. Both Test Set II and REVERB ASR use an eight-microphone circular array with a 20 cm diameter, and the target speaker is non-moving within each utterance. We follow a *plug-and-play* approach for ASR, where enhanced signals are directly fed into a multi-conditionally trained ASR backend for decoding. The backend is built based on the official REVERB corpus using the Kaldi script[5]. It is composed of a GMM-HMM system, a time-delay DNN (TDNN) trained with lattice-free maximum mutual information based on online-extracted i-vectors and MFCCs, and a tri-gram language model. Note that the window length and hop size for ASR are

[5] https://github.com/kaldi-asr/kaldi/tree/master/egs/reverb/s5 (commit 61637e6c8ab01d3b4c54a50d9b20781a0aa12a59). Different from the Kaldi script, our study (1) performs sentence-level cepstral mean normalization on the input features of TDNN; (2) reduces the initial batch size of TDNN training by changing the trainer.num-chunk-per-minibatch flag from 256,128,64 to 128,64; (3) increases the number of TDNN training epochs from 10 to 20; (4) uses wsj/s5/local/wer_output_filter and wsj/s5/local/wer_hyp_filter to filter out tokens such as NOISE and SPOKEN_NOISE when utilizing local/score.sh to compute WER; and (5) enforces the same word insertion penalty (WIP) for near- and far-field conditions, and uses the averaged WER on the near- and far-field conditions of the validation set to select the best WIP.

respectively 25 and 10 ms, following the default setup in Kaldi. During testing, we first obtain enhanced time-domain signals using our frontend and then feed them to the ASR backend for decoding, meaning that our frontend does not leverage any knowledge of the backend. We emphasize that the purpose of Test Set II and REVERB ASR is to show the generalization ability of our dereverberation models, which are trained based on simulated training data, as well as to compare the proposed algorithms with unsupervised methods such as WPE, not to obtain state-of-the-art performance using dereverberation frontends trained on the REVERB training data.

The two DNNs in Figure 6-1 are trained sequentially. We first train the single-channel model using the first channel of all the multi-channel signals (in total 7,861×5 utterances). Designating the first microphone as the reference, we use the trained model to obtain a beamformed signal based on a random subset of microphones. The beamforming result is then combined with the mixture signal to train the second network. This way, the second DNN can deal with beamforming results produced by using up to eight microphones. Figure 6-3 illustrates the DNN architecture. We use two-layer recurrent neural networks with BLSTM having an encoder-decoder structure similar to U-Net, skip connections, and dense blocks as the learning machines for masking and mapping. The motivation for this DNN design is that BLSTM can model long-term temporal information, U-Net can maintain fine-grained local information as is suggested in image semantic segmentation [126], and DenseNet encourages feature reuse and improves the discriminative capability of the network [67], [92], [144]. In our experiments, this network architecture shows consistent improvements over the standard BLSTM and a recently proposed convolutional recurrent neural network [148]. The encoder contains one 2D convolution, and six

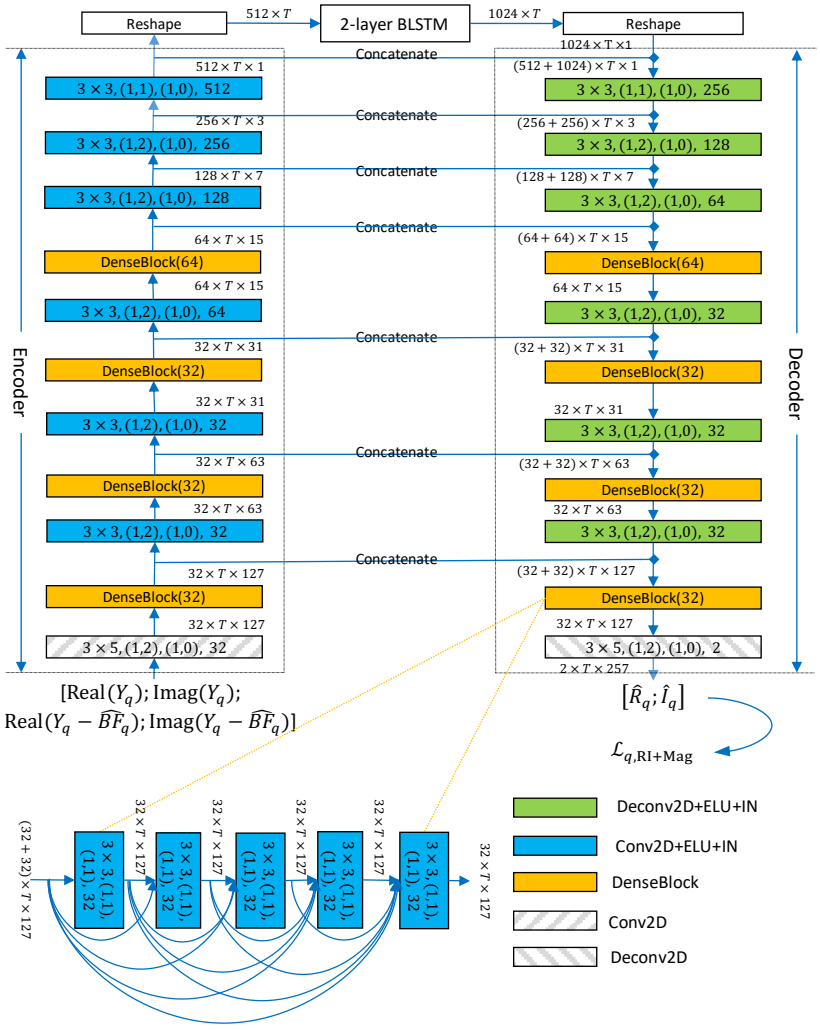


Figure 6-3. Network architecture for predicting the RI components of S_q from the RI components of Y_q and $Y_q - \widehat{BF}_q$. Note that for single-channel processing, the network only takes in single-channel information as its inputs. The tensor shape after each block is in format: *featureMaps*×*timeSteps*×*frequencyChannels*. Each Conv2D, Deconv2D, Conv2D+ELU+IN, and Deconv2D+ELU+IN block is specified in format: *kernelSizeTime* × *kernelSizeFreq, (stridesTime, stridesFreq), (paddingsTime, paddingsFreq), featureMaps*. Each DenseBlock(g) contains five Conv2+ELU+IN blocks with growth rate g.

convolutional blocks, each with 2D convolution, ELUs and instance normalization (IN) [198], for down-sampling. The decoder includes six convolutional blocks, each with 2D deconvolution, ELUs and IN, and one 2D deconvolution, for up-sampling. Each BLSTM

layer has 512 units in each direction. The frontend processing uses 32 ms window length and 8 ms frame shift for STFT. The sampling rate is 16 kHz. A square-root Hann window is used as the analysis window.

Our main evaluation metrics are SI-SDR [88] and PESQ, where the former is a time-domain metric that closely reflects the quality of estimated phase, and the latter strongly correlates with the accuracy of estimated magnitudes. We also consider scale-dependent SDR (SD-SDR) [88] for evaluating the single-channel models. Following REVERB, we also use cepstral distance (CD), log likelihood ratio (LLR), frequency-weighted segmental SNR (fwSegSNR), and speech-to-reverberation modulation energy ratio (SRMR) as the evaluation metrics. Note that the computation of SRMR does not require clean references. WER is used to evaluate ASR performance.

6.4.2. Baseline Systems for Comparison

This section describes the single- and multi-channel baselines considered in our study.

6.4.2.1. Single-Channel Baselines

The first four baselines for dereverberation perform single-channel magnitude-domain masking and mapping based MSA and PSA [161], which are popular approaches in single-channel speech enhancement. We summarize the baselines in Table 6-1. All of them use the same network architecture in Figure 6-3, and the key difference is in the number of input and output feature maps depending on the input features and training targets, output non-linearities and loss functions. $T_a^b(\cdot) = \max\left(\min(\cdot, b), a\right)$ in $\mathcal{L}_{\text{MSA}-\text{Masking}}$ and $\mathcal{L}_{\text{PSA}-\text{Masking}}$ truncates the estimated mask to the range $[a, b]$. α in $\mathcal{L}_{\text{MSA}-\text{Masking}}$ is set to 10.0, and β and γ in $\mathcal{L}_{\text{PSA}-\text{Masking}}$ respectively set to 1.0 and 0.0 in our study.

Table 6-1. Summary of various single-channel models for speech dereverberation.

Method	Input features	Loss function	Network Output	Output activation	Enhancement results
Complex spectral mapping	$\text{Real}(Y_q)$, $\text{Imag}(Y_q)$	$\mathcal{L}_{\text{RI}}$ or $\mathcal{L}_{\text{RI+Mag}}$	$\hat{R}_q, \hat{I}_q$	Linear	$\hat{S}_q = \hat{R}_q + j\hat{I}_q$ $\hat{V}_q = Y_q - \hat{S}_q$
MSA-Masking	$\lvert Y_q \rvert$	$\mathcal{L}_{\text{MSA-Masking}} = \left\| \lvert Y_q \rvert T_0^{\alpha}(\widehat{M}_q^{(s)}) - T_0^{\alpha\lvert Y_q \rvert}(\lvert S_q \rvert) \right\|_1 + \left\| \lvert Y_q \rvert T_0^{\alpha}(\widehat{M}_q^{(v)}) - T_0^{\alpha\lvert Y_q \rvert}(\lvert V_q \rvert) \right\|_1$	$\widehat{M}_q^{(s)}, \widehat{M}_q^{(v)}$	Clipped Softplus	$\hat{S}_q = Y_q T_0^{\alpha}(\widehat{M}_q^{(s)})$ $\hat{V}_q = Y_q T_0^{\alpha}(\widehat{M}_q^{(v)})$
MSA-Mapping		$\mathcal{L}_{\text{MSA-Mapping}} = \left\| \widehat{U}_q^{(s)} - \lvert S_q \rvert \right\|_1 + \left\| \widehat{U}_q^{(v)} - \lvert V_q \rvert \right\|_1$	$\widehat{U}_q^{(s)}, \widehat{U}_q^{(v)}$	Softplus	$\hat{S}_q = \widehat{U}_q^{(s)} e^{j\angle Y_q}$ $\hat{V}_q = \widehat{U}_q^{(v)} e^{j\angle Y_q}$
PSA-Masking		$\mathcal{L}_{\text{PSA-Masking}} = \left\| \lvert Y_q \rvert T_{\gamma}^{\beta}(\hat{Q}_q^{(s)}) - T_{\gamma\lvert Y_q \rvert}^{\beta\lvert Y_q \rvert}(\lvert S_q \rvert \cos(\angle S_q - \angle Y_q)) \right\|_1 + \left\| \lvert Y_q \rvert T_{\gamma}^{\beta}(\hat{Q}_q^{(v)}) - T_{\gamma\lvert Y_q \rvert}^{\beta\lvert Y_q \rvert}(\lvert V_q \rvert \cos(\angle V_q - \angle Y_q)) \right\|_1$	$\hat{Q}_q^{(s)}, \hat{Q}_q^{(v)}$	Sigmoid	$\hat{S}_q = Y_q T_{\gamma}^{\beta}(\hat{Q}_q^{(s)})$ $\hat{V}_q = Y_q T_{\gamma}^{\beta}(\hat{Q}_q^{(v)})$
PSA-Mapping		$\mathcal{L}_{\text{PSA-Mapping}} = \left\| \hat{Z}_q^{(s)} - \lvert S_q \rvert \cos(\angle S_q - \angle Y_q) \right\|_1 + \left\| \hat{Z}_q^{(v)} - \lvert V_q \rvert \cos(\angle V_q - \angle Y_q) \right\|_1$	$\hat{Z}_q^{(s)}, \hat{Z}_q^{(v)}$	Linear	$\hat{S}_q = \hat{Z}_q^{(s)} e^{j\angle Y_q}$ $\hat{V}_q = \hat{Z}_q^{(v)} e^{j\angle Y_q}$

6.4.2.2. TI-MVDR

To show the effectiveness of using estimated complex spectra for covariance matrix computation, we apply the single-channel models to enhance each microphone signal following the last column of Table 6-1, and then compute the covariance matrices based on Eq. (6.5) for TI-MVDR. This method is denoted as $\widehat{BF}_q$. Additionally, we use mask-weighted ways [58], [203] of computing covariance matrices for TI-MVDR, based on the estimated masks produced by the models trained with $\mathcal{L}_{\text{MSA-Masking}}$ and $\mathcal{L}_{\text{PSA-Masking}}$

$$\widehat{\Phi}^{(d)}(f) = \frac{1}{T}\sum_t \eta^{(d)}(t,f)\boldsymbol{Y}(t,f)\,\boldsymbol{Y}(t,f)^H, \tag{6.11}$$

where $d \in \{s, v\}$.

When using $\mathcal{L}_{\text{MSA-Masking}}$, $\eta^{(d)}$ is computed as

$$\eta^{(d)} = \text{median}\left(\frac{T_0^{\alpha}\left(\widehat{M}_1^{(d)}\right)}{T_0^{\alpha}\left(\widehat{M}_1^{(s)}\right) + T_0^{\alpha}\left(\widehat{M}_1^{(v)}\right)}, \ldots, \frac{T_0^{\alpha}(\widehat{M}_P^{(d)})}{T_0^{\alpha}(\widehat{M}_P^{(s)}) + T_0^{\alpha}(\widehat{M}_P^{(v)})}\right), \tag{6.12}$$

where $\widehat{M}_p^{(d)}$ denotes the estimated magnitude mask at microphone p.

When using $\mathcal{L}_{\text{PSA}-\text{Masking}}$, $\eta^{(d)}$ is computed as

$$\eta^{(d)} = \text{median}\left(T_\gamma^\beta\left(\hat{Q}_1^{(d)}\right), \ldots, T_\gamma^\beta\left(\hat{Q}_P^{(d)}\right)\right), \tag{6.13}$$

where $\hat{Q}_p^{(d)}$ denotes the estimated phase-sensitive mask at microphone p.

We also square the mask before median pooling, as the outer product is in the energy domain, while in Eq. (6.13) and (6.12) the mask is in the magnitude domain. $\eta^{(d)}$ is computed as

$$\eta^{(d)} = \text{median}\left(\frac{T_0^\alpha\left(\widehat{M}_1^{(d)}\right)^2}{T_0^\alpha\left(\widehat{M}_1^{(s)}\right)^2 + T_0^\alpha\left(\widehat{M}_1^{(v)}\right)^2}, \ldots, \frac{T_0^\alpha(\widehat{M}_P^{(d)})^2}{T_0^\alpha(\widehat{M}_P^{(s)})^2 + T_0^\alpha(\widehat{M}_P^{(v)})^2}\right) \tag{6.14}$$

for $\mathcal{L}_{\text{PSA}-\text{Masking}}$ and as

$$\eta^{(d)} = \text{median}\left(T_\gamma^\beta\left(\hat{Q}_1^{(d)}\right)^2, \ldots, T_\gamma^\beta\left(\hat{Q}_P^{(d)}\right)^2\right) \tag{6.15}$$

for $\mathcal{L}_{\text{PSA}-\text{Masking}}$. Note that α, β and γ are respectively set to 10.0, 1.0 and 0.0 in our study.

6.4.2.3. Post-filtering (no re-training)

After obtaining $\widehat{BF}_q$, we then apply the single-channel models to $\widehat{BF}_q$ for post-filtering. Note that the phase in $\widehat{BF}_q$ is used as the estimated phase for magnitude-domain masking and mapping based models. We emphasize that $\widehat{BF}_q$ is still very reverberant and is expected to contain low speech distortion. It is therefore reasonable to feed $\widehat{BF}_q$ into a single-channel model trained on unprocessed mixtures for further enhancement. Note that in this method,

only one DNN is trained (i.e. the single-channel model), but it is run twice at run time. This method is denoted as $\widehat{BF}_q$ + *Post-filtering (no re-training)*.

6.4.2.4. Post-filtering (re-training)

As $\widehat{BF}_q$ may contain distortion unseen by the single-channel models, which are trained on unprocessed mixtures. We train a complex spectral mapping based post-filter, which predicts the RI components of S_q based on $\widehat{BF}_q$. Similar to the proposed system shown in Figure 6-1, this method uses two DNNs, while the input to the second DNN is $\widehat{BF}_q$ rather than Y_q and $Y_q - \widehat{BF}_q$. We denote this method as $\widehat{BF}_q$ + Post-filtering (re-training).

6.4.2.5. Single- and Multi-Channel WPE

We follow the script for REVERB in Kaldi, which is based on the open-source *nara-wpe* toolkit [30], to build our offline WPE baselines, where the window size is 32 ms and hop size is 8 ms, the prediction delay is set to 3, the iteration number set to 5, and the order of the regressive model set to 40 for single-channel processing and 10 for multi-channel processing. Note that these hyperparameters are the recommended ones in [76] and [26].

6.5. Evaluation Results

We first report the dereverberation performance of the trained models on Test Set I, and then report their generalization ability on Test Set II and REVERB ASR.

6.5.1. Dereverberation Performance on Test Set I

In Table 6-2, we compare the performance of single-channel magnitude-domain masking and mapping based MSA and PSA, and complex spectral mapping over unprocessed speech and oracle magnitude-domain masks such as the spectral magnitude

Table 6-2. Average SI-SDR (dB), PESQ and SD-SDR (dB) of different methods on single-channel dereverberation (Test Set I). Oracle masking results are marked in gray.

Method	SI-SDR	PESQ	SD-SDR
Unprocessed	-3.7	1.93	-3.7
$\mathcal{L}_{\text{MSA−Masking}}$	0.8	2.91	3.5
$\mathcal{L}_{\text{MSA−Mapping}}$	0.7	2.92	3.5
$\mathcal{L}_{\text{PSA−Masking}}$	2.3	2.55	4.5
$\mathcal{L}_{\text{PSA−Mapping}}$	1.6	2.56	4.2
$\mathcal{L}_{\text{RI}}$	**6.2**	2.80	**7.2**
$\mathcal{L}_{\text{RI+Mag}}$	5.9	**3.07**	7.0
SMM ($T_0^{10}(\lvert S_q\rvert/\lvert Y_q\rvert)$)	1.6	3.40	3.9
PSM ($T_0^1(\lvert S_q\rvert\cos(\angle S_q - \angle Y_q)/\lvert Y_q\rvert)$)	4.5	3.09	5.8

mask [166] and phase-sensitive mask [35]. Note that the unprocessed SI-SDR is closely related to DRR, an important factor characterizing the difficulty of dereverberation along with T60. Comparing $\mathcal{L}_{\text{MSA−Masking}}$, $\mathcal{L}_{\text{MSA−Mapping}}$, $\mathcal{L}_{\text{PSA−Masking}}$ and $\mathcal{L}_{\text{PSA−Mapping}}$ and $\mathcal{L}_{\text{RI}}$, we observe that $\mathcal{L}_{\text{RI}}$ leads to much better SI-SDR than MSA and PSA (6.2 vs. 0.8, 0.7, 2.3 and 1.6 dB), while MSA obtains the best PESQ (2.91 and 2.92 vs. 2.55, 2.56 and 2.80). This is likely because PESQ is closely related to the quality of estimated magnitudes, while time-domain measures such as SI-SDR needs the estimated magnitudes to compensate for the error of phase estimation. Compared with $\mathcal{L}_{\text{RI}}$, $\mathcal{L}_{\text{RI+Mag}}$ substantially improves PESQ from 2.80 to 3.07, slightly degrading SI-SDR from 6.2 to 5.9 dB. In addition, $\mathcal{L}_{\text{RI+Mag}}$ obtains better PESQ than MSA (3.07 vs. 2.91 and 2.92), indicating the effectiveness of phase processing. We observe that SD-SDR results are consistent with SI-SDR. In the following experiments, we use $\mathcal{L}_{\text{RI+Mag}}$ as the loss function to train the two DNNs in Figure 6-1, as it yields a very strong SI-SDR and the highest PESQ.

In Table 6-3, we compare the performance of TI-MVDR and post-filtering based on the statistics computed using the single-channel models in Table 6-2. Among all the

Table 6-3. Average SI-SDR (dB) and PESQ of different methods for TI-MVDR and post-filtering using eight microphones (Test Set I).

Method	Model	Covariance Matrices	#mics	SI-SDR	PESQ
$\widehat{BF}_q$	$\mathcal{L}_{\text{MSA-Masking}}$	Eq. (6.5)	8	2.3	2.27
	$\mathcal{L}_{\text{MSA-Mapping}}$			2.3	2.26
	$\mathcal{L}_{\text{PSA-Masking}}$			3.3	2.31
	$\mathcal{L}_{\text{PSA-Mapping}}$			2.8	2.31
	$\mathcal{L}_{\text{RI}}$			5.8	2.34
	$\mathcal{L}_{\text{RI+Mag}}$			5.6	2.34
	$\mathcal{L}_{\text{MSA-Masking}}$	Eq. (6.11), (6.12)		1.7	2.44
	$\mathcal{L}_{\text{PSA-Masking}}$	Eq. (6.11), (6.13)		3.3	2.45
	$\mathcal{L}_{\text{MSA-Masking}}$	Eq. (6.11), (6.14)		3.0	2.44
	$\mathcal{L}_{\text{PSA-Masking}}$	Eq. (6.11), (6.15)		4.2	2.44
$\widehat{BF}_q$+*Post-filtering (no re-training)*	$\mathcal{L}_{\text{MSA-Masking}}$	Eq. (6.5)		4.4	3.01
	$\mathcal{L}_{\text{MSA-Mapping}}$			4.3	3.03
	$\mathcal{L}_{\text{PSA-Masking}}$			5.2	2.85
	$\mathcal{L}_{\text{PSA-Mapping}}$			4.7	2.87
	$\mathcal{L}_{\text{RI}}$			**9.6**	3.10
	$\mathcal{L}_{\text{RI+Mag}}$			9.4	**3.23**
	$\mathcal{L}_{\text{MSA-Masking}}$	Eq. (6.11), (6.12)		3.5	3.10
	$\mathcal{L}_{\text{PSA-Masking}}$	Eq. (6.11), (6.13)		5.3	2.96
	$\mathcal{L}_{\text{MSA-Masking}}$	Eq. (6.11), (6.14)		4.7	3.10
	$\mathcal{L}_{\text{PSA-Masking}}$	Eq. (6.11), (6.15)		6.1	2.95

alternative ways of computing the statistics for TI-MVDR, using the $\mathcal{L}_{\text{RI}}$ and $\mathcal{L}_{\text{RI+Mag}}$ models with Eq. (6.5) obtains the highest SI-SDR (5.8 and 5.6 dB), and the PESQ scores (2.34 and 2.34) are better than using MSA and PSA models with Eq. (6.5) (2.27, 2.26, 2.31 and 2.31) while worse than using MSA and PSA models with Eq. (6.11) (2.44, 2.45, 2.44 and 2.44). Applying post-filtering to $\widehat{BF}_q$ computed using the $\mathcal{L}_{\text{RI}}$ and $\mathcal{L}_{\text{RI+Mag}}$ models and Eq. (6.5) shows the highest SI-SDR scores (9.6 and 9.4 dB), and $\mathcal{L}_{\text{RI+Mag}}$ leads to significantly better PESQ over $\mathcal{L}_{\text{RI}}$ (3.23 vs. 3.10). These results suggest the effectiveness of complex spectral mapping based beamforming and post-filtering. In the following

Table 6-4. Average SI-SDR (dB) and PESQ of different methods on multi-channel dereverberation (Test Set I).

Metrics	#mics	Mixture	Model	$\widehat{BF}_q$+*Postfiltering (no re-training)*	$\widehat{BF}_q$+*Postfiltering (re-training)*	$\hat{S}_q^{(1)}$	$\hat{S}_q^{(2)}$
SI-SDR	1	-3.7	$\mathcal{L}_{\text{RI+Mag}}$ and Eq. (6.5)	-	-	5.9	-
	2			7.3	7.4	-	**7.5**
	3			8.2	8.9	-	**9.1**
	4			8.6	9.7	-	**9.9**
	6			9.2	10.6	-	**10.8**
	8			9.4	11.0	-	**11.2**
PESQ	1	1.93		-	-	3.07	-
	2			3.14	3.17	-	**3.18**
	3			3.20	**3.29**	-	**3.29**
	4			3.22	**3.34**	-	**3.34**
	6			3.23	3.40	-	**3.41**
	8			3.23	**3.44**	-	**3.44**

experiments, we compute $\widehat{BF}_q$ using Eq. (6.5) and $\mathcal{L}_{\text{RI+Mag}}$ if not specified, as this combination obtains the highest PESQ and a very competitive SI-SDR.

In Table 6-4, we show the results of $\hat{S}_q^{(2)}$, obtained by combining Y_q and $Y_q - \widehat{BF}_q$ for dereverberation (see Figure 6-1). Consistently better performance is obtained over $\hat{S}_q^{(1)}$, confirming the effectiveness of multi-channel processing (e.g. 11.2 vs. 5.9 dB in SI-SDR and 3.44 vs. 3.07 in PESQ in the eight-microphone case). $\hat{S}_q^{(2)}$ also obtains better performance than $\widehat{BF}_q$ +*Post-filtering (no re-training)*, especially when the number of microphones is greater than two, for instance 11.2 vs. 9.4 dB in SI-SDR and 3.44 vs. 3.23 in PESQ in the eight-channel case. It is also slightly better than $\widehat{BF}_q$+*Post-filtering (re-training)*. These results demonstrate the gains of combining $Y_q - \widehat{BF}_q$ with Y_q for dereverberation. In the two-channel case, it obtains results slightly better than $\widehat{BF}_q$+*Post-filtering (no re-training)*, likely because $\widehat{BF}_q$ is not accurate enough in such a case. As a

Table 6-5. Average LLR, CD, fwSegSNR, PESQ, and SRMR of different approaches on Test Set II.

Data	Metrics	Unprocessed	#mics	$\hat{S}_q^{(1)}$	$\hat{S}_q^{(2)}$	WPE	WPE+ BeamformIt	WPE+DNN-Based MVDR ($\mathcal{L}_{\text{RI+Mag}}$ and Eq. (6.5))
SimData	CD	5.08	1	**3.16**	-	4.95	-	-
			2	-	**3.01**	4.98	4.66	4.77
			8	-	**2.78**	4.81	3.94	4.45
	LLR	0.67	1	**0.53**	-	0.63	-	-
			2	-	**0.45**	0.61	0.60	0.55
			8	-	**0.39**	0.53	0.49	0.40
	fwSegSNR	8.32	1	**15.61**	-	9.38	-	-
			2	-	**16.94**	9.71	10.20	11.24
			8	-	**18.75**	11.38	12.48	14.20
	PESQ	2.37	1	**3.29**	-	2.51	-	-
			2	-	**3.51**	2.58	2.65	2.77
			8	-	**3.71**	2.82	3.10	3.21
RealData	SRMR	3.18	1	**6.69**	-	3.83	-	-
			2	-	**6.38**	3.99	4.08	4.00
			8	-	**6.30**	5.04	5.53	5.29

result, the quality of $Y_q - \widehat{BF}_q$ is not as good as when more microphones are available, and the trained DNN would focus on dealing with features computed from more than two microphones.

6.5.2. Generalization on Test Set II and REVERB ASR

In Table 6-5, we directly evaluate the performance of the trained dereverberation models on Test Set II. Our models obtain dramatically better performance than WPE, and WPE+BeamformIt which applies weighted delay-and-sum beamforming on the output of WPE, and WPE+DNN-Based MVDR. Note that the first two baselines are available in Kaldi, and the third baseline applies DNN based TI-MVDR beamforming after WPE, where we use the single-channel model trained with $\mathcal{L}_{\text{RI+Mag}}$ and Eq. (6.5) to compute the statistics for MVDR, based on the signals processed after WPE. These comparisons show

Table 6-6. Average WER (%) of different methods on real data of REVERB ASR.

#mics	Method	Validation Set			Test Set		
		Near	Far	Avg	Near	Far	Avg
1	Mixture	16.53	17.22	16.88	17.31	17.05	17.18
	$\hat{S}_q^{(1)}$	**10.61**	**11.35**	**10.98**	**9.26**	**9.28**	**9.27**
	WPE	13.54	15.79	14.66	13.38	14.25	13.82
2	$\widehat{BF}_q$ ($\mathcal{L}_{\text{RI+Mag}}$ and Eq. (6.5))	21.21	22.83	22.02	21.02	18.26	19.64
	$\hat{S}_q^{(2)}$	**9.23**	**9.43**	**9.33**	**7.98**	**8.27**	**8.12**
	WPE	12.98	16.75	14.87	12.46	14.01	13.23
	WPE+BeamformIt	12.41	14.76	13.59	12.49	14.25	13.37
	WPE+DNN-Based MVDR ($\mathcal{L}_{\text{RI+Mag}}$ and Eq. (6.5))	16.91	20.98	18.95	17.18	14.01	15.59
8	$\widehat{BF}_q$ ($\mathcal{L}_{\text{RI+Mag}}$ and Eq. (6.5))	13.41	12.10	12.75	13.13	10.97	12.05
	$\hat{S}_q^{(2)}$	**7.92**	**7.72**	**7.82**	**5.88**	**6.41**	**6.14**
	WPE	12.48	15.31	13.89	11.21	11.75	11.48
	WPE+BeamformIt	9.54	10.59	10.06	8.24	8.61	8.43
	WPE+DNN-Based MVDR ($\mathcal{L}_{\text{RI+Mag}}$ and Eq. (6.5))	9.92	11.00	10.46	9.52	8.34	8.93

that the trained DNN models exhibit good generalization to novel reverberant and noisy conditions, and array configurations.

In Table 6-6, we report the ASR performance of the trained dereverberation models on the REVERB real data. The proposed approach obtains clear WER improvements over WPE, WPE+BeamformIt and WPE+DNN-Based MVDR (9.27% vs. 13.82% in the single-channel case, 8.12% vs. 13.23%, 13.37% and 15.59% in the two-channel case, and 6.14% vs. 11.48%, 8.43% and 8.93% in the eight-channel case). We observe large improvement by using $\hat{S}_q^{(2)}$, which can also be thought of as a variant of post-filtering, over $\widehat{BF}_q$. These results suggest that the trained dereverberation models can suppress reverberation with low speech distortion. We observe that the WPE+DNN-Based MVDR obtains better WER than $\widehat{BF}_q$, suggesting that WPE works as a frontend for DNN based beamforming, but worse WER than WPE+BeamformIt possibly because of the effects of reverberation.

6.6. Conclusion

We have proposed a complex spectral mapping approach for speech dereverberation, where we predict the RI components of direct sound from the mixture. We have extended this approach to multi-channel dereverberation, by incorporating the RI components of cancelled speech for model training. Our single- and multi-channel models show clear improvements over magnitude spectrum and phase-sensitive spectrum based models, and single- and multi-channel WPE. The trained models generalize reasonably well to novel and representative reverberant environments and array configurations.

Chapter 7. Multi-Channel Speech Enhancement and Robust ASR

This chapter investigates multi-channel speech enhancement and its application to robust ASR using deep learning based complex spectral mapping. This work has been published in ICASSP 2017 and 2018 [127], [134], [193], and is under consideration by IEEE/ACM T-ASLP [187] at the time of dissertation writing.

7.1. Introduction

Environmental noise and room reverberation are very detrimental to modern ASR systems and dramatically degrade speech intelligibility and quality [161], [51]. Practical systems typically use multiple microphones to leverage spatial (in addition to spectral) information for speech enhancement and audio source separation. One common approach for multi-channel speech enhancement is beamforming followed by post-filtering [40], [49], where a popular method is to decompose a time-invariant or time-varying multi-channel Wiener filter into a product of an MVDR beamformer and a real-valued post-filter. Conventionally, this approach requires an accurate estimate of target direction, and speech and noise PSD and covariance matrices. Recently, DNN based T-F masking or mapping have been established as a mainstream approach for speech enhancement and source separation [161]. Mask (or magnitude) estimation is dramatically improved using deep

learning. Such real-valued mask estimates have been used to identify T-F units dominated by a single source, where the phase is less corrupted, for accurate source localization [175] and covariance matrix estimation [58], [160]. All the top teams in the recent CHiME-4 challenge adopted T-F masking and deep learning based beamforming in their ASR systems [160].

We investigate single- and multi-channel DNN-based speech enhancement and robust ASR. In addition to mask (or magnitude) estimation, our study explores the effects of phase estimation for multi-channel speech enhancement. We emphasize that current T-F masking based approaches for beamforming typically compute spatial covariance matrices as a summation of mixture outer products weighted by a mask [36], [58], [64], [200], [203], [213]. In environments with strong noise and room reverberation, there may be insufficient T-F units dominated by target speech, and the mixture outer product at each T-F unit inevitably contains noise and reverberation. We believe, in such cases, that it is beneficial to perform phase estimation in addition to magnitude estimation and directly use the estimated complex spectra for covariance matrix computation. In addition, real-valued post-filtering only performs magnitude estimation and would inevitably produce phase inconsistency issues [42], [44], [184]. Although beamforming typically improves phase, its performance heavily depends on the number of microphones and is susceptible to strong room reverberation [40]. Phase estimation would hence be needed for post-filtering in order to further improve the phase produced by beamforming. Although modern ASR systems only consider magnitude-based features, accurate phase estimation can indirectly benefit ASR as better estimated phase leads to better spatial processing such as beamforming and target localization.

Our study performs DNN based phase estimation and investigates its effects on single-channel enhancement, time-invariant and time-varying beamforming, and post-filtering. We perform speech enhancement in the complex domain [192], more specifically via complex spectral mapping [39], [146], which was originally proposed to deal with single-channel speech enhancement in anechoic conditions. This paper goes beyond previous work on complex spectral mapping by using a new loss function and addressing multi-channel speech enhancement and robust ASR. The proposed system advances state-of-the-art enhancement and recognition results on the single-, two- and six-microphone tasks of CHiME-4, without using any model ensemble as employed in the previous best results reported in [32] and [153] that combines multiple frontends and backends.

The rest of this paper is organized as follows. We describe our physical model and objectives in Chapter 7.2, and present the proposed algorithms in 7.3. Experimental setup and evaluation results are presented in Chapter 7.4 and 7.5. Conclusions are made in Chapter 7.6.

7.2. Physical Model and Objectives

The hypothesized physical model is the same as in Eq. (6.1). The $\boldsymbol{N}(t,f)$ we deal with in this chapter are more challenging and realistic recorded noises. Again, we refer to $\boldsymbol{S}(t,f) = \boldsymbol{c}(f;p)S_q(t,f)$ as the target speech to extract, and $\boldsymbol{V}(t,f) = \boldsymbol{H}(t,f) + \boldsymbol{N}(t,f)$ as the non-target signal to remove. See Eq. (6.1) for detailed notation definitions.

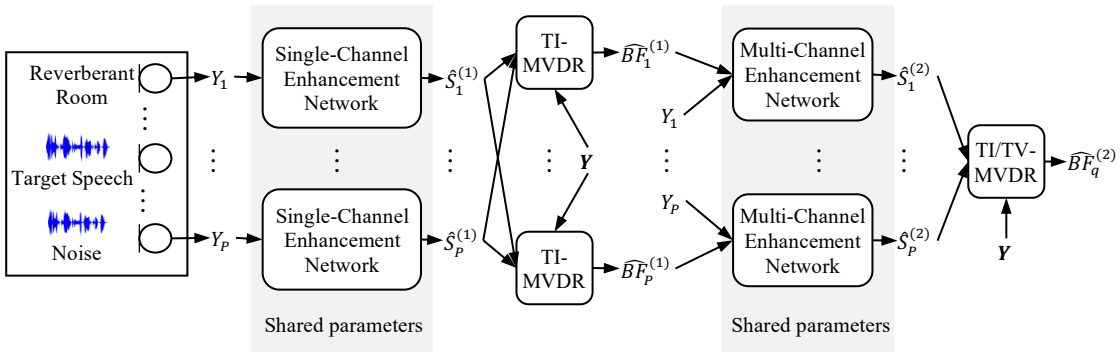


Figure 7-1. System diagram of overall system for single- and multi-channel speech enhancement. There are two DNNs, one taking in single-channel and the other multi-channel information for speech enhancement. The superscripts in $\hat{S}_p^{(1)}$ and $\widehat{BF}_p^{(1)}$, and $\hat{S}_p^{(2)}$ and $\widehat{BF}_p^{(2)}$ for $p \in \{1, \ldots, P\}$ respectively denote whether they are produced by the first and the second DNN. The MVDR beamformer can be time-invariant or time-varying. Detailed DNN architecture is shown in Figure 7-2.

7.3. Proposed Algorithms

Figure 7-1 shows two DNNs in the proposed system. The first one performs single-channel complex spectral mapping based enhancement, and the enhancement results are utilized to compute an MVDR beamformer. The beamforming results are combined with the mixture for the second DNN to perform multi-channel complex spectral mapping based speech enhancement so that spectral and spatial information can be integrated during DNN training. A second beamformer is then computed for speech recognition, as the second DNN can produce better signal statistics for beamforming after leveraging spatial information. The single- and multi-channel complex spectral mapping respectively follow Chapter 6.3.1 and 6.3.2. This section describes a novel technique for time-variant MVDR beamforming.

7.3.1. Adaptive Covariance Matrix Computation

Since the target speaker is typically still within each utterance, it is reasonable to estimate RTF from $\widehat{\Phi}^{(s)}(f)$ using all the frames within an utterance. Clearly, more frames in this case lead to more accurate RTF estimation for a still directional source. However, even if the target speaker is still, the spatial coherence of environmental noise and room reverberation can be highly time-varying in real-world environments such as the BUS and CAF conditions in the CHiME-4 corpus. It is hence necessary to estimate noise covariance matrix per T-F unit or per block of units rather than per frequency for more accurate noise suppression.

We follow a recently proposed algorithm [85] to estimate time-varying noise covariance matrices. In [85], per-frequency T-F mask based covariance matrix is considered as a prior, and under a maximum a posterior framework, the time-varying spatial covariance matrix at each T-F unit is computed as a weighted combination of the prior and the summation of the mask-weighted mixture outer products in each non-overlapping block of T-F units. Specifically, we compute the time-varying noise covariance matrix in the following way

$$\widehat{\Phi}^{(v)}(t,f) = (1-\alpha)\frac{\sum_{t-\Delta}^{t+\Delta}\widehat{\boldsymbol{V}}(t,f)\widehat{\boldsymbol{V}}(t,f)^{H}}{trace\left(\sum_{t-\Delta}^{t+\Delta}\widehat{\boldsymbol{V}}(t,f)\widehat{\boldsymbol{V}}(t,f)^{H}\right)/P} + \alpha\frac{\widehat{\Phi}^{(v)}(f)}{trace\left(\widehat{\Phi}^{(v)}(f)\right)/P}, \quad (7.1)$$

where α is empirically set to 0.5, Δ is half the window size in frames. See Chapter 6.3.1 and 6.3.2 for how $\widehat{\boldsymbol{V}}$ and $\widehat{\Phi}^{(v)}(f)$ are computed. Different from [85], we use estimated complex spectra produced by complex spectral mapping, rather than estimated masks in a mask-weighted fashion, for covariance matrix computation. This could result in more

accurate covariance estimation. In addition, we normalize the energy levels before the weighted sum to eliminate the effects of time-varying PSD and focus on the weighted summation of spatial coherences, as noise PSD cancels out in MVDR beamforming. Without the energy normalization, the summation can be easily dominated by one of the two terms, since noise PSD can be highly non-stationary. We emphasize that the first term is computed based on a small context window of $2\Delta + 1$ frames, while the second term based on all the frames. This way, the computation of the noise covariance matrix can leverage long-term stationary information and, at the same time, adapt to sudden changes of noise characteristics. Note that the short-term noise covariance matrix needs an accurate complex spectrum estimate, which is obtained using complex spectral mapping. After cross validation, Δ is set to 0 for the two-microphone task and 3 for the six-microphone task of the CHiME-4 corpus.

A time-varying MVDR (TV-MVDR) beamformer is then computed as

$$\widehat{\boldsymbol{w}}(t,f;q) = \frac{\widehat{\Phi}^{(v)}(t,f)^{-1}\hat{\boldsymbol{c}}(f;q)}{\hat{\boldsymbol{c}}(f;q)^{H}\widehat{\Phi}^{(v)}(t,f)^{-1}\hat{\boldsymbol{c}}(f;q)}, \tag{7.2}$$

where $\hat{\boldsymbol{c}}(f;q)$ is computed as in Chapter 6.3.2. The beamforming result is computed using $\widehat{BF}_q(t,f) = \widehat{\boldsymbol{w}}(t,f;q)^H \boldsymbol{Y}(t,f)$.

7.4. Experimental Setup

We evaluate our algorithms on the enhancement and recognition tasks of the publicly-available CHiME-4 corpus [160], a popular dataset featuring one-, two- and six-microphone tasks designed for robust ASR. Our study always considers the direct outputs from DNN (i.e. $\hat{S}_q^{(1)}$ and $\hat{S}_q^{(2)}$) for speech enhancement, and beamforming results (i.e.

$\widehat{BF}_q^{(1)}$ and $\widehat{BF}_q^{(2)}$) for speech recognition, as it is well-known that beamforming produces less speech distortion, which is important for modern ASR systems, but also less noise reduction, compared to deep learning based masking and mapping. This section details the CHiME-4 dataset, our proposed frontend and several baseline frontends, and our ASR backend.

7.4.1. CHiME-4 Corpus

The CHiME-4 corpus [160] contains six-microphone simulated and real recordings. The microphones are mounted on a tablet, with five of them facing the front and the other one facing the rear. This corpus contains recordings from four real-world environments (including street, pedestrian areas, cafeteria and bus), exhibiting large training and testing mismatches in terms of speaker, noise and spatial characteristics, and around 12% of its real recordings suffer from microphone failures. The training data includes 7,138 simulated and 1,600 recorded utterances, the validation data contains 1,640 simulated and 1,640 recorded utterances, and the test data consists of 1,320 simulated and 1,320 recorded utterances. Each of the three recorded datasets is constructed using four different speakers. It should be noted that reverberation is weak in the CHiME-4 corpus, partly because the considered environments are not very reverberant and the speaker-microphone distance is not large for a hand-held position. The single-channel task uses one of the six microphones for testing. For the two-microphone task, two of the front five channels that do not suffer from microphone failure are selected for each utterance for testing. To address microphone failures in the real recordings of the six-microphone task, we first select a microphone signal that is most correlated with the other five, and then throw away the signals with less than 0.3 correlation coefficients with the selected signal.

7.4.2. Frontend Enhancement System

We use all the simulated signals in the training set to train our frontends, and report the enhancement results on the simulated test set. We consider the clean signal captured by the fifth microphone as the reference for metric computation, since it exhibits the highest signal-to-noise ratio among all the microphones.

The two DNNs in Figure 7-1 are trained sequentially. After training the first DNN, we use it to generate for each microphone a beamformed signal based on TI-MVDR and a random number of microphones, leading to 7,138×6 beamformed signals in total. Each beamformed signal is combined with the mixture signal to train the second DNN. This way, the second DNN can deal with the TI-MVDR results produced by using up to six microphones. In our experiments, we also tried using TV-MVDR to produce beamformed signals for training the second DNN. The performance is however not clearly better.

The network architecture for enhancement is shown in Figure 7-2. The network is a temporal convolutional network (TCN) [7] with encoder-decoder structure similar to U-Net [126], skip connections, and dense blocks [67], [144]. The motivation for this network design is that TCN can model long-term temporal dependencies through large receptive fields achieved via dilated convolution, U-Net can maintain fine-grained local spectral structure as suggested in image semantic segmentation [126], and dense blocks can increase feature reuse and improve the discriminative power of the network [67]. A similar architecture was recently used in a speaker separation algorithm [92]. The encoder contains one 2D convolution, and six convolutional blocks, each with 2D convolution, Swish non-linearity and IN, for down-sampling. The decoder includes six blocks of 2D deconvolution, Swish and IN, and one 2D deconvolution, for up-sampling. The TCN contains two layers,

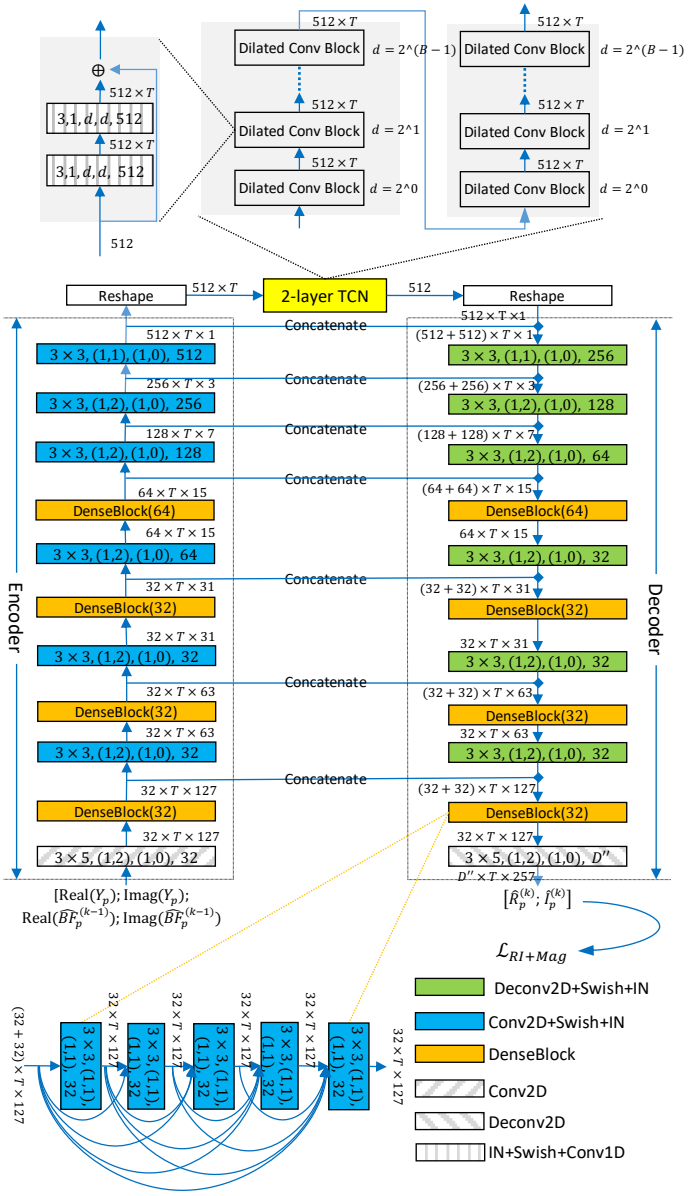


Figure 7-2. Network architecture for predicting the RI components of S_q from the RI components of Y_q and $\widehat{BF}_q$. For single-channel processing, the network only takes single-channel information as its inputs. The tensor shape after each encoder-decoder block is in the format: *featureMaps*×*timeSteps*×*frequencyChannels*. Each of Conv2D, Deconv2D, Conv2D+IN+Swish, and Deconv2D+IN+Swish blocks is specified in the format: *kernelSizeTime* × *kernelSizeFreq, (stridesTime, stridesFreq), (paddingsTime, paddingsFreq), featureMaps*. Each DenseBlock(g) contains five Conv2D+IN+Swish blocks with growth rate g. The tensor shape after each TCN block is in the format: *featureMaps* × *timeSteps*. Each IN+Swish+Conv1D block is specified in the format: *kernelSizeTime, stridesTime, paddingsTime, dilationTime, featureMaps*.

each of which has six dilated convolutional blocks. We use two one-dimensional (1D) depth-wise separable convolution in each dilated convolutional block to reduce the number of parameters.

The frame length is 32 ms and frame shift 8 ms. Square-root Hann window is used as the analysis window. The sampling rate is 16 kHz. A 512-point discrete Fourier transform is used to extract complex STFT spectrograms. No global mean-variance normalization is performed on the input features. For complex spectral mapping, linear activation is used in the output layer to produce estimated RI components. As the CHiME-4 dataset exhibits diverse gains at different microphones, we separately normalize each of the six microphone signals to have unit sample variance before any frontend processing.

We use PESQ, STOI, SI-SDR [88], and *bss-eval* SDR as the evaluation metrics. PESQ and STOI strongly correlate with the accuracy of estimated magnitude. On the other hand, SI-SDR is a time-domain metric closely reflecting the quality of estimated magnitude and phase, meaning that magnitude estimates need to compensate for the inaccuracy of phase estimates in order to produce a high SI-SDR.

7.4.3. Baseline Frontend Systems

We consider four single-channel benchmarks listed in Table 7-1 to demonstrate the effectiveness of single-channel complex spectral mapping based speech enhancement. The four benchmarks are based on masking and mapping based MSA [161] and PSA [35]. All of them use the same network architecture as shown in Figure 7-2. The main differences lie in the number of input and output feature maps, and the activation function in the output layer. In $\mathcal{L}_{\text{MSA}-\text{Masking}}$ and $\mathcal{L}_{\text{PSA}-\text{Masking}}$, $T_a^b(\cdot) = \max\left(\min(\cdot, b), a\right)$ truncates the

Table 7-1. Summary of single-channel frontends.

Method	Input features	Loss function	Network output	Output activation	Enhancement results
Complex Spectral Mapping	$\text{Real}(Y_q)$, $\text{Imag}(Y_q)$	$\mathcal{L}_{\text{RI}}$ or $\mathcal{L}_{\text{RI+Mag}}$	$\hat{R}_q, \hat{I}_q$	Linear	$\hat{S}_q = \hat{R}_q + j\hat{I}_q$ $\hat{V}_q = Y_q - \hat{S}_q$
MSA-Masking	$\lvert Y_q \rvert$	$\mathcal{L}_{\text{MSA-Masking}} = \left\| \lvert Y_q \rvert T_0^{\beta}(\hat{M}_q^{(s)}) - T_0^{\beta\lvert Y_q \rvert}(\lvert S_q \rvert) \right\|_1 + \left\| \lvert Y_q \rvert T_0^{\beta}(\hat{M}_q^{(v)}) - T_0^{\beta\lvert Y_q \rvert}(\lvert V_q \rvert) \right\|_1$	$\hat{M}_q^{(s)}, \hat{M}_q^{(v)}$	Clipped Softplus	$\hat{S}_q = Y_q T_0^{\beta}(\hat{M}_q^{(s)})$ $\hat{V}_q = Y_q T_0^{\beta}(\hat{M}_q^{(v)})$
MSA-Mapping		$\mathcal{L}_{\text{MSA-Mapping}} = \left\| \hat{R}_q^{(s)} - \lvert S_q \rvert \right\|_1 + \left\| \hat{R}_q^{(v)} - \lvert V_q \rvert \right\|_1$	$\hat{R}_q^{(s)}, \hat{R}_q^{(v)}$	Softplus	$\hat{S}_q = \hat{R}_q^{(s)} e^{j\angle Y_q}$ $\hat{V}_q = \hat{R}_q^{(v)} e^{j\angle Y_q}$
PSA-Masking		$\mathcal{L}_{\text{PSA-Masking}} = \left\| \lvert Y_q \rvert T_0^{\gamma}(\hat{Q}_q^{(s)}) - T_0^{\gamma\lvert Y_q \rvert}(\lvert S_q \rvert \cos(\angle S_q - \angle Y_q)) \right\|_1 + \left\| \lvert Y_q \rvert T_0^{\gamma}(\hat{Q}_q^{(v)}) - T_0^{\gamma\lvert Y_q \rvert}(\lvert V_q \rvert \cos(\angle V_q - \angle Y_q)) \right\|_1$	$\hat{Q}_q^{(s)}, \hat{Q}_q^{(v)}$	Sigmoid	$\hat{S}_q = Y_q T_0^{\gamma}(\hat{Q}_q^{(s)})$ $\hat{V}_q = Y_q T_0^{\gamma}(\hat{Q}_q^{(v)})$
PSA-Mapping		$\mathcal{L}_{\text{PSA-Mapping}} = \left\| \hat{Z}_q^{(s)} - \lvert S_q \rvert \cos(\angle S_q - \angle Y_q) \right\|_1 + \left\| \hat{Z}_q^{(v)} - \lvert V_q \rvert \cos(\angle V_q - \angle Y_q) \right\|_1$	$\hat{Z}_q^{(s)}, \hat{Z}_q^{(v)}$	Linear	$\hat{S}_q = \hat{Z}_q^{(s)} e^{j\angle Y_q}$ $\hat{V}_q = \hat{Z}_q^{(v)} e^{j\angle Y_q}$

estimated masks to the range $[a, b]$. β is set to 5.0 in $\mathcal{L}_{\text{MSA-Masking}}$ and γ set to 1.0 in $\mathcal{L}_{\text{PSA-Masking}}$.

In addition, we investigate the effectiveness of the single-channel models for TI-MVDR beamforming. One way is to apply each single-channel model to each microphone signal to obtain $\widehat{\boldsymbol{S}}$ and $\widehat{\boldsymbol{V}}$, perform TI-MVDR beamforming using Eq. (6.5)-(6.9), and compare their ASR performance. This comparison can show the effectiveness of single-channel phase estimation when its result is used for beamforming.

We also evaluate the mask weighting technique for collecting statistics for TI-MVDR beamforming, based on the MSA-Masking and PSA-Masking models. Following [203], [58], [36], [213], we compute the covariance matrices in the following way

$$\widehat{\Phi}^{(d)}(f) = \frac{1}{T}\sum_{t=1}^{T} \eta^{(d)}(t,f)\boldsymbol{Y}(t,f)\,\boldsymbol{Y}(t,f)^{H}, \tag{7.3}$$

where $d \in \{s, v\}$, and $\eta^{(d)}$ is computed as

$$\eta^{(d)} = \text{median}\left(\frac{T_0^{\beta}\left(\widehat{M}_1^{(d)}\right)}{T_0^{\beta}\left(\widehat{M}_1^{(s)}\right) + T_0^{\beta}\left(\widehat{M}_1^{(v)}\right)}, \ldots, \frac{T_0^{\beta}\left(\widehat{M}_1^{(d)}\right)}{T_0^{\beta}\left(\widehat{M}_1^{(s)}\right) + T_0^{\beta}\left(\widehat{M}_1^{(v)}\right)}\right) \tag{7.4}$$

for MSA-Masking and as

$$\eta^{(d)} = \text{median}\left(T_0^{\gamma}\left(\widehat{Q}_1^{(d)}\right), \ldots, T_0^{\gamma}\left(\widehat{Q}_P^{(d)}\right)\right) \tag{7.5}$$

for PSA-Masking. Here β is set to 5.0 and γ set to 1.0 in our study.

7.4.4. Backend Recognition System

Our ASR backend is a DNN-HMM hybrid system built from the Kaldi toolkit. The acoustic model is trained using both simulated and recorded noisy utterances in the training set. The input features to the acoustic model are 80-dimensional logarithmically compressed Mel filterbank feature together with its delta and double delta. The acoustic model is a wide-residual BLSTM network (WRBN) [61] trained with utterance-wise recurrent dropout [164]. At test time, we perform lattice re-scoring using the task-standard trigram, five-gram and RNN language models, and an LSTM language model (LSTMLM) recently proposed in [18]. The LSTMLM re-scored lattice is used for unsupervised speaker adaptation. We apply iterative speaker adaptation proposed in [164] for three iterations, each of which follows the linear input network algorithm [209].

Since the ASR system uses different frame and shift sizes from speech enhancement frontends, we perform signal re-synthesis before extracting features for recognition.

Table 7-2. Average SI-SDR (dB), PESQ, and STOI (%) performance of different methods on channel 5 of CHiME-4 (single-channel).

Methods	SI-SDR	PESQ	STOI
Unprocessed	7.5	2.18	87.0
$\mathcal{L}_{\text{MSA-Masking}}$	13.9	2.94	93.9
$\mathcal{L}_{\text{MSA-Mapping}}$	14.6	3.00	94.5
$\mathcal{L}_{\text{PSA-Masking}}$	14.9	2.84	94.3
$\mathcal{L}_{\text{PSA-Mapping}}$	15.0	2.90	94.3
$\mathcal{L}_{\text{RI}}$	15.5	2.96	95.2
$\mathcal{L}_{\text{RI+Mag}}$	**15.8**	**3.16**	**95.4**
SMM ($T_0^5(\lvert S_q\rvert/\lvert Y_q\rvert)$)	17.2	3.64	98.5
PSM ($T_0^1(\lvert S_q\rvert\cos(\angle S_q - \angle Y_q)/\lvert Y_q\rvert)$)	17.6	3.72	98.1

7.5. Evaluation Results

We first report speech enhancement performance and then recognition results on the CHiME-4 dataset.

7.5.1. Enhancement Performance

Table 7-2 compares the enhancement performance of single-channel complex-domain mapping with single-channel magnitude-domain masking and mapping, along with oracle magnitude-domain masking using the SMM [161] and PSM [35]. We observe better SI-SDR, PESQ and STOI results using the model trained with $\mathcal{L}_{\text{RI}}$ and $\mathcal{L}_{\text{RI+Mag}}$ than with $\mathcal{L}_{\text{MSA-Masking}}$, $\mathcal{L}_{\text{MSA-Mapping}}$, $\mathcal{L}_{\text{PSA-Masking}}$ and $\mathcal{L}_{\text{PSA-Mapping}}$, indicating the effectiveness of complex-domain estimation. Compared with $\mathcal{L}_{\text{RI}}$, $\mathcal{L}_{\text{RI+Mag}}$ yields much better PESQ (3.16 vs. 2.96), slightly better SI-SDR (15.8 vs. 15.5 dB), and marginally better STOI (95.4% vs. 95.2%). This suggests the importance of magnitude estimation for PESQ. The following experiments use $\mathcal{L}_{\text{RI+Mag}}$ as the default loss function.

Table 7-3. Average SI-SDR (dB), PESQ, and STOI (%) of different methods on channel 5 of CHiME-4 (six-channel).

Methods	SI-SDR	PESQ	STOI
Unprocessed	7.5	2.18	87.0
$\widehat{BF}_q^{(1)}$+*post-filtering* ($\mathcal{L}_{\text{MSA-Masking}}$)	18.6	3.32	97.3
$\widehat{BF}_q^{(1)}$+*post-filtering* ($\mathcal{L}_{\text{MSA-Mapping}}$)	19.8	3.38	97.9
$\widehat{BF}_q^{(1)}$+*post-filtering* ($\mathcal{L}_{\text{PSA-Masking}}$)	19.8	3.32	97.8
$\widehat{BF}_q^{(1)}$+*post-filtering* ($\mathcal{L}_{\text{PSA-Mapping}}$)	19.4	3.30	97.5
$\widehat{BF}_q^{(1)}$+*post-filtering* ($\mathcal{L}_{\text{RI}}$)	19.3	3.46	98.0
$\widehat{BF}_q^{(1)}$+*post-filtering* ($\mathcal{L}_{\text{RI+Mag}}$)	20.0	3.54	98.1
$\hat{S}_q^{(2)}$ ($\mathcal{L}_{\text{RI+Mag}}$)	**22.0**	**3.68**	**98.6**

Table 7-4. Comparison of average SI-SDR (dB), SDR (dB), PESQ, and STOI (%) of different approaches on channel 5 of CHiME-4 (six-channel).

Methods	SI-SDR	SDR	PESQ	STOI
Unprocessed	7.5	7.6	2.18	87.0
$\hat{S}_q^{(2)}$ ($\mathcal{L}_{\text{RI+Mag}}$)	**22.0**	**22.4**	**3.68**	**98.6**
Bu *et al.* [13]	-	-	2.69	93.9
Tu *et al.* [154]	-	-	2.71	94.0
Shimada *et al.* [139]	-	16.2	2.70	94.0

Table 7-3 reports the performance of multi-channel enhancement. One straightforward approach, denoted as $\widehat{BF}_q^{(1)}$+*post-filtering*, is to first utilize a single-channel model listed in Table 4-1 to obtain $\widehat{BF}_q^{(1)}$ via Eq. (6.5)-(6.9) (see also Figure 7-1), and then apply the single-channel model again on $\widehat{BF}_q^{(1)}$ for post-filtering. Since $\widehat{BF}_q^{(1)}$ is expected to contain low speech distortion, it can be used as the input to the single-channel model for post-filtering, although the model is trained on noisy mixtures. Clearly, using $\widehat{BF}_q^{(1)}$+*post-filtering* obtained via the model trained with $\mathcal{L}_{\text{RI+Mag}}$ leads to the best performance. This is consistent with the single-channel results in Table 7-1. Another approach, denoted as $\hat{S}_q^{(2)}$ (see Figure 7-1), combines $\widehat{BF}_q^{(1)}$ and Y_q to train another DNN for multi-channel

Table 7-5. Comparison of ASR performance (%WER) with other approaches (single-channel).

Approaches	Dev. Set		Test Set	
	Simu.	Real	Simu.	Real
Mixtures + Trigram	8.24	6.67	12.98	10.70
+ Five-gram and RNNLM	6.58	4.84	11.17	8.38
+ LSTMLM	5.65	4.06	10.58	8.12
+ Iterative Speaker Adaptation	**4.99**	**3.54**	**9.41**	**6.82**
Kaldi baseline [18]	6.81	5.58	12.15	11.42
Du *et al*. [32]	6.61	4.55	11.81	9.15
Wang and Wang [164] (No LSTMLM)	6.77	4.99	11.14	8.28

complex spectral mapping. Clearly better results are observed over $\widehat{BF}_q^{(1)}$+*post-filtering*, but at the expense of using one more DNN. Note that both of them show clear improvements over single-channel enhancement.

Table 7-4 compares the proposed approach with other competitive approaches in the literature. Bu *et al.* [13] utilize estimated masks produced by BLSTM based single-channel masking to compute the signal statistics for MVDR beamforming and magnitude-domain post-filtering. Tu *et al.* [154] combine the estimated mask produced by complex Gaussian mixture models (CGMM) with the estimated ideal ratio mask (IRM) provided by an LSTM for masking-based block-wise MVDR, and use another LSTM for monaural magnitude mapping based post-filtering for further noise reduction. In [139], Shimada *et al.* combine CGMM based spatial clustering and multi-channel non-negative matrix factorization based spectral modeling to estimate time-varying speech and noise covariance matrices for time-varying beamforming. As can be observed from Table 7-4, substantially better enhancement results are obtained by our approach over the comparison approaches.

7.5.2. Recognition Performance

Table 7-5 reports ASR performance on the single-channel task of CHiME-4. Our single-channel system directly uses unprocessed noisy signals for recognition and obtains 6.82% WER after lattice-rescoring and iterative speaker adaptation. This result is significantly better than the previous best WERs reported by Du et al. [32], and Wang and Wang [164]. This result suggests that our backend is a strong one and can be very indicative at measuring the effectiveness of frontend enhancement for recognition. It should be noted that we tried to use the enhancement results of our single-channel frontends for recognition. The ASR performance is however worse than using unprocessed mixtures. This is likely due to the speech distortion introduced by DNN based enhancement and the large mismatch between the training and test conditions of CHiME-4.

Table 7-6 presents the ASR results of TI- and TV-MVDR using single- and multi-channel models, based on the trigram language model for decoding. We explain the results by using the two-channel task as an example. Entries 1-8 are obtained by using various single-channel models to compute the statistics for TI-MVDR, either by using Eq. (6.5) and Eq. (6.6) or Eq. (6.11) for covariance matrix computation. Among these entries, we found that entry 8 obtains the highest score, which indicates the effectiveness of DNN based phase estimation for beamforming. Entry 9 is obtained by using multi-channel complex spectral mapping to compute $\hat{S}_p^{(2)}$, and then deriving a TI-MVDR (see Figure 7-1 for more details). Slightly better WER is observed over entry 8, suggesting that the second DNN leads to better signal statistics for beamforming than the first one. Entry 10 uses $\hat{S}_p^{(2)}$

Table 7-6. ASR Performance (%WER) of using various single- and multi-channel models for TI- and TV-MVDR, and trigram language model for decoding.

#mics	Entry	Methods	$\widehat{\Phi}^{(s)}$, $\widehat{\Phi}^{(v)}$	Dev. Set		Test Set	
				Simu.	Real	Simu.	Real
2	1	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{MSA-Masking}}$)	Eq. (6.11), (7.4)	6.23	5.58	8.45	8.44
	2	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{PSA-Masking}}$)	Eq. (6.11), (6.13)	6.23	5.48	8.43	8.58
	3	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{MSA-Masking}}$)	Eq. (6.5), (6.6)	6.06	5.54	7.83	8.63
	4	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{MSA-Mapping}}$)		6.06	5.48	7.86	8.46
	5	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{PSA-Masking}}$)		6.05	5.50	7.85	8.37
	6	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{PSA-Mapping}}$)		6.15	5.50	8.18	8.36
	7	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{RI}}$)		5.98	5.52	7.82	8.23
	8	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{RI+Mag}}$)		5.93	5.48	7.68	8.29
	9	$\widehat{BF}_q^{(2)}$ ($\mathcal{L}_{\text{RI+Mag}}$)		5.91	5.42	7.74	8.11
	10	$\widehat{BF}_q^{(2)}$ ($\mathcal{L}_{\text{RI+Mag}}$)	Eq. (6.5), (7.1)	**5.32**	**5.03**	**6.85**	**7.72**
6	11	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{MSA-Masking}}$)	Eq. (6.11), (7.4)	4.16	4.24	5.16	5.75
	12	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{PSA-Masking}}$)	Eq. (6.11), (6.13)	4.04	4.15	4.87	5.55
	13	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{MSA-Masking}}$)	Eq. (6.5), (6.6)	3.98	4.24	4.75	6.08
	14	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{MSA-Mapping}}$)		3.97	4.20	4.64	5.95
	15	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{PSA-Masking}}$)		3.97	4.19	4.66	5.78
	16	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{PSA-Mapping}}$)		4.05	4.28	5.02	6.10
	17	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{RI}}$)		3.79	4.16	4.47	5.59
	18	$\widehat{BF}_q^{(1)}$ ($\mathcal{L}_{\text{RI+Mag}}$)		3.91	4.15	4.55	5.69
	19	$\widehat{BF}_q^{(2)}$ ($\mathcal{L}_{\text{RI+Mag}}$)		3.86	4.12	4.42	5.34
	20	$\widehat{BF}_q^{(2)}$ ($\mathcal{L}_{\text{RI+Mag}}$)	Eq. (6.5), (7.1)	**3.58**	**3.99**	**4.22**	**5.18**

to compute a TV-MVDR. Clearly better WER is observed over entry 9, indicating the effectiveness of using estimated complex spectra to compute time-varying noise covariance matrices for beamforming. Similar trend is observed on the six-channel task.

Table 7-7 and Table 7-8 further apply five-gram, RNN and LSTM language models for lattice re-scoring and perform iterative speaker adaptation for the two- and six-channel tasks, based respectively on the TV-MVDR frontends produced in the entry 10 and entry 20 of Table 7-6.

Table 7-7. Comparison of ASR performance (%WER) with other approaches (two-channel).

Approaches	Dev. Set		Test Set	
	Simu.	Real	Simu.	Real
$\widehat{BF}_q^{(2)}$ ($\mathcal{L}_{\text{RI+Mag}}$, Eq. (6.5) and (7.1)) + Trigram	5.32	5.03	6.85	7.72
+ Five-gram and RNNLM	3.74	3.32	4.84	5.54
+ LSTMLM	2.52	2.15	3.28	3.80
+ Iterative Speaker Adaptation	**2.17**	**1.99**	**2.53**	**3.19**
Kaldi baseline [18]	3.94	2.85	5.03	5.40
Du *et al*. [32]	3.46	2.33	5.74	3.91

Table 7-8. Comparison of ASR performance (%WER) with other approaches (six-channel).

Approaches	Dev. Set		Test Set	
	Simu.	Real	Simu.	Real
$\widehat{BF}_q^{(2)}$ ($\mathcal{L}_{\text{RI+Mag}}$, Eq. (6.5) and (7.1)) + Trigram	3.58	3.99	4.22	5.18
+ Five-gram and RNNLM	2.44	2.58	2.97	3.73
+ LSTMLM	1.43	1.69	1.80	2.34
+ Iterative Speaker Adaptation	**1.26**	**1.51**	**1.46**	**2.04**
Kaldi baseline [18]	2.10	1.90	2.66	2.74
Du *et al*. [32]	1.78	1.69	2.12	2.24

Table 7-5, Table 7-7 and Table 7-8 also compare the proposed system with other state-of-the-art systems. Our system advances state-of-the-art ASR results on all the tasks. The system in Du *et al*. [32] (and their journal version [153]) was the winning solution in the CHiME-4 challenge, and produces the best WER results reported to date. It ensembles one DNN and four CNN based acoustic models as the backend, using a combination of log Mel filterbank, fMLLR and i-vectors as the input features. Their frontend uses T-F masking based MVDR beamforming, where the estimated masks are combined on the basis of an unsupervised CGMM model, a supervised LSTM based IRM estimator, and frame-level voice activity detection results produced by a speech recognizer. An LSTM language model is used for lattice re-scoring. As can be seen, their frontend and backend are both ensembles of multiple models. In contrast, our system does not use any model ensemble,

and obtains better ASR results on all the three tasks (6.82% vs. 9.15%, 3.19% vs. 3.91%, and 2.04% vs. 2.24% WER). These amount to 25.5%, 18.4%, and 8.9% relative WER reductions for the single-, two-, and six-microphone tasks, respectively. The improvement is especially large on the simulated test data of the two- and six-microphone tasks (2.53% vs. 5.74%, and 1.46% vs. 2.12% WER), indicating that the proposed system is particularly effective when training and testing conditions are not very different. Another system worth mentioning is a recently-proposed CHiME-4 baseline [18] available in Kaldi. The frontend is a masking based generalized eigenvector beamformer based on a BLSTM, the acoustic model is a time-delay DNN trained with a lattice-free version of the maximum mutual information criterion, and an LSTM language model, which is the one we use in our study, is trained for lattice re-scoring. Our system obtains much better ASR results, demonstrating the effectiveness of the proposed frontend and backend.

7.6. Conclusion

We have proposed a complex spectral mapping approach for single- and multi-channel speech enhancement. Experiments on the CHiME-4 corpus show that complex spectral mapping leads to better single-channel enhancement, beamforming and post-filtering, over magnitude-domain masking and mapping. Our adaptive noise covariance matrix estimation yields further ASR improvements over TI-MVDR, especially on the two-channel task. State-of-the-art results have been obtained on the enhancement and recognition tasks of the CHiME-4 corpus.

Chapter 8. Multi-Microphone Complex Spectral Mapping for Speech Dereverberation

This chapter investigates multi-channel speech dereverberation on fixed-geometry arrays, where we train DNNs using multi-microphone inputs based on complex spectral mapping. This work has been published in ICASSP 2020 [188].

8.1. Introduction

The multi-channel systems in Chapter 6 and Chapter 7 assume a relatively blind setup, where the trained models are designed to be directly applicable to arrays with any number of microphones arranged in an unknown geometry. Although this flexibility is desirable, in applications such as Amazon Echo and Google Home, the device only has a fixed microphone array with a known number of microphones and geometry. How to leverage this fixed geometry for robust speech processing is therefore an interesting research problem to investigate.

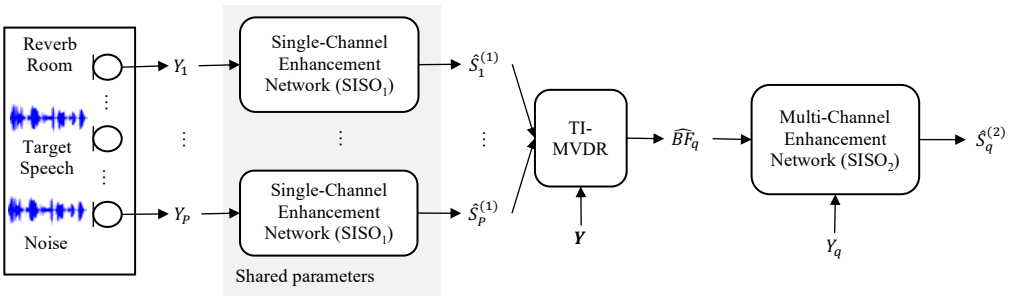
This chapter proposes a multi-microphone complex spectral mapping approach for speech dereverberation based on a fixed array geometry, where the real and imaginary (RI) components of multiple microphones are concatenated as input features for a DNN to predict the RI components of the direct-path signal(s) captured at a reference microphone or at all the microphones. The initially estimated target speech can be utilized to compute

a beamformer, and the RI components of the beamforming results can be further combined with the RI components of all the microphone signals for post-filtering.

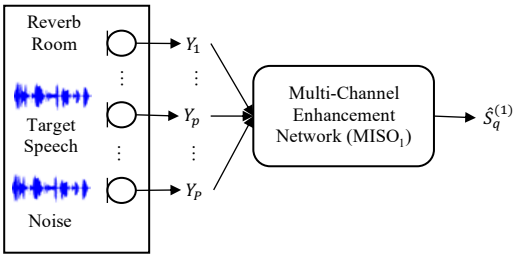
Why should this approach work? We believe that, for a fixed-geometry array, the neural network could learn to enhance the speech arriving from a specific direction by exploiting the spatial information contained in multiple microphones. This approach is in a way similar to recent studies of classification-based sound source localization for arrays with fixed geometry, where a DNN is trained to learn a one-to-one mapping from the inter-channel phase patterns of multiple microphones to the target direction [16], [38], [99], [199]. Based on deep learning, the proposed approach has the potential to model the non-linear spatial information contained in multi-microphone inputs, while conventional beamforming is only linear and typically utilizes second-order statistics [40] within each frequency.

Although there are time-domain approaches that use multi-microphone modeling for speech enhancement and source separation [90], [141], [150], their effectiveness in environments with moderate to strong reverberation is not yet established [96]. In addition, our study tightly integrates multi-microphone complex spectral mapping with beamforming and post-filtering.

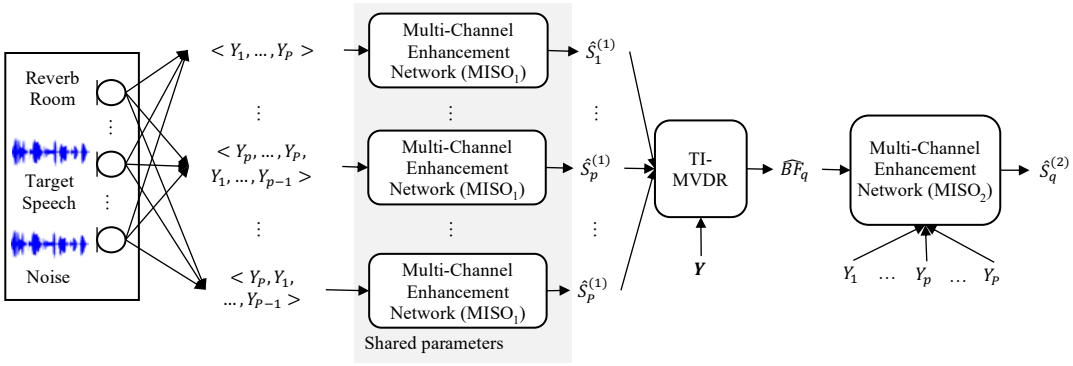
The rest of this paper presents the physical model and proposed algorithms in Chapter 8.2 and 8.3, experimental setup and evaluation results in Chapter 8.4 and 8.5, and conclusions in Chapter 8.6.



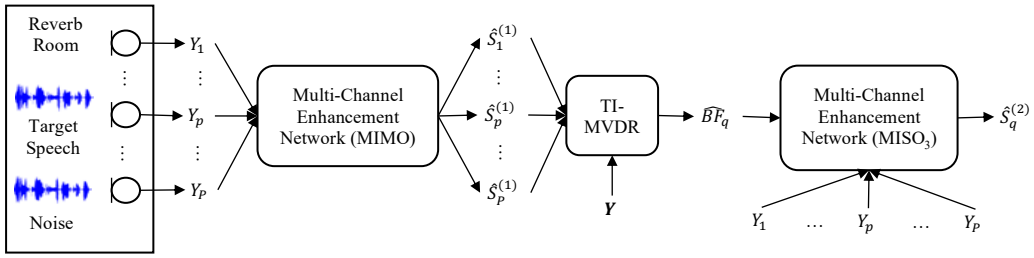
(a) SISO$_1$-BF-SISO$_2$ system.



(b) MISO$_1$ system.



(c) MISO$_1$-BF-MISO$_2$ system.



(d) MIMO-BF-MISO$_3$ system.

Figure 8-1. System overview.

8.2. Physical Model and Objectives

The hypothesized physical model and objectives are the same as in Chapter 6.2. Different from Chapter 6, we assume that the same microphone array is used for both training and testing.

8.3. Proposed Algorithms

We propose four approaches (denoted as $SISO_1$-BF-$SISO_2$, $MISO_1$, $MISO_1$-BF-$MISO_2$, and MIMO-BF-$MISO_3$, see Figure 8-1) for multi-channel speech dereverberation. This section discusses each one of them and their combination with beamforming and post-filtering. All the TI-MVDR beamforming results are computed based on Eq. (6.5)-(6.10).

8.3.1. $SISO_1$-BF-$SISO_2$ System

The $SISO_1$-BF-$SISO_2$ system contains two single-input and single-output (SISO) networks. The first one ($SISO_1$) performs single-channel complex spectral mapping at each microphone. The enhanced speech is used to compute a TI-MVDR beamformer. The beamforming result $\widehat{BF}_q$ is then combined with the mixture at the reference microphone Y_q as the input to the second SISO network ($SISO_2$) for complex spectral mapping based post-filtering.

This system is essentially similar to the one described in Chapter 6.

8.3.2. $MISO_1$ System

The multiple-input and single-output system (denoted as $MISO_1$) stacks the RI components of the mixtures at all the microphones and predicts the RI components of the direct-path signal at a reference microphone. This algorithm essentially trains a DNN for

non-linear time-varying beamforming. It is simple, fast, and can be easily modified for real-time processing. The model is trained using $\mathcal{L}_{q,\mathrm{RI+Mag}}$.

We emphasize that conventional multi-channel Wiener filtering computes a linear filter per frequency or per T-F unit to project the mixture $\boldsymbol{Y}(t,f)$ onto $S_q(t,f)$, typically based on second-order statistics [40]. In contrast, we utilize a DNN to learn a highly non-linear function to map $\boldsymbol{Y}$ to S_q. Although this seems challenging for arrays with arbitrary geometry, for a fixed geometry, this could work as the inter-channel phase patterns are almost fixed for the signal arriving from a specific direction.

8.3.3. MISO$_1$-BF-MISO$_2$ System

The MISO$_1$-BF-MISO$_2$ system includes a MISO network, an MVDR beamformer, and another MISO network. This system is similar to SISO$_1$-BF-SISO$_2$, but we use two MISO networks rather than two SISO networks, since MISO is expected to be better than SISO by doing multi-microphone modeling.

We circularly shift the microphones to estimate the direct-path signal at each microphone. For example, we stack an ordered microphone sequence $< Y_1, \dots, Y_P >$ as the inputs to MISO$_1$ to obtain $\hat{S}_1^{(1)}$, and feed in $< Y_p, \dots, Y_P, Y_1, \dots, Y_{p-1} >$ to obtain $\hat{S}_p^{(1)}$. This strategy would work as we use a circular array with uniformly spaced microphones.

An MVDR beamformer is then computed using $\widehat{\boldsymbol{S}}$. The beamforming result $\widehat{BF_q}$ is combined with $\boldsymbol{Y}$ to predict S_q using a MISO network (denoted as MISO$_2$) via complex spectral mapping. This way, post-filtering can also leverage multi-microphone modeling.

8.3.4. MIMO-BF-MISO$_3$ System

The MIMO-BF-MISO$_3$ system consists of a multiple-input and multiple-output (MIMO) network, an MVDR beamformer, and a MISO network. The MIMO network takes in the mixture RI components of all the microphones to predict the RI components of the direct-path signals at all the microphones. This way, we can get an estimate of $\boldsymbol{S}$ for beamforming by performing feed-forwarding only once, rather than P times as in SISO$_1$-BF-SISO$_2$ and MISO$_1$-BF-MISO$_2$. The amount of computation is therefore dramatically reduced. The loss function for the MIMO network is

$$\mathcal{L}_{1,\ldots,P,\text{RI+Mag+PhaseDiff}} = \frac{1}{P}\sum_{p=1}^{P}\mathcal{L}_{p,\text{RI+Mag}} + \frac{1}{P^2-P}\sum_{p'=1}^{P}|S_{p'}|\sum_{p''=1}^{P}\left(1-\cos\left(\angle\hat{S}_{p'}-\angle\hat{S}_{p''}-\left(\angle S_{p'}-\angle S_{p''}\right)\right)\right)/2 \tag{8.1}$$

where the first term is defined as in Eq. (6.3), and the second term is a magnitude-weighted cosine distance between the predicted phase differences and the actual phase differences of all the microphone pairs. In our experiments, the second term leads to faster convergence and better performance over using the first term alone.

After obtaining $\widehat{\boldsymbol{S}}$, we compute an MVDR beamformer. The beamforming result $\widehat{BF_q}$ is combined with $\boldsymbol{Y}$ to predict S_q using a MISO network (denoted as MISO$_3$) via complex spectral mapping.

Input: WSJCAM0;
Output: spatialized reverberant (and noisy) WSJCAM0;
For *dataset, REP* in {*train:5*, *validation:4*, *test:3}* set of WSJCAM0 **do**
  For each anechoic speech signal s in *dataset* **do**
    Repeat *REP* times **do**
      - Draw room length r_x and width r_y from [5,10] m, and height r_z from [3,4] m;
      - Sample mic array height a_z from $[1,2]$ m;
      - Sample array displacement n_x and n_y from $[-0.5,0.5]$ m;
      - Place array center at $\langle \frac{r_x}{2} + n_x, \frac{r_y}{2} + n_y, a_z \rangle$ m;
      - Set array radius a_r to 0.1 m;
      - Sample angle of first mic ϑ from $[0, \frac{\pi}{4}]$;
      - Place $P(=8)$ mics uniformly on the circle, starting from angle ϑ;
      - Sample target speaker locations: $\langle s_x, s_y, s_z(=a_z) \rangle$ such that distance from target speaker to array center is in between $[0.75,2.5]$ m, and target speaker is at least 0.5 m from each wall;
      - Sample T60 from $[0.2,1.3]$ s;
      - Generate multi-channel impulse responses and convolve them with *s*;
      If *dataset* in {*train*, *validation*} do
       -Sample a *P*-channel noise signal n from REVERB training noise;
      Else
       - Sample a *P*-channel noise signal n from REVERB testing noise;
      End
      - Concatenate channels of reverberated *s* and n respectively, scale them to an SNR randomly sampled from $[5,25]$ dB, and mix them;
    End
  End
End

Algorithm 8-1. Data spatialization process.

8.4. Experimental Setup

We use the WSJ0CAM corpus and a large set of simulated RIRs (in total 39,305 eight-channel RIRs) to simulate room reverberation. See Algorithm 8-1 for the detailed simulation procedure. For each utterance, we randomly generate a room with different room characteristics, microphone and speaker locations, array configurations, and noise levels. Our study considers an eight-microphone circular array with the radius fixed at 10 cm. The target speaker is in the same plane as the array, at a distance sampled from [0.75,2.5] m. The training and testing noise (mostly air-conditioning noise) used in

REVERB [77] is utilized to simulate noisy-reverberant mixtures for training and testing, respectively. The reverberation time (T60) is randomly drawn from the range [0.2,1.3] s. The average direct-to-reverberation energy ratio is -3.7 dB with 4.4 dB standard deviation. There are 39,305 (7,861×5, ~80 h), 2,968 (742×4, ~6 h) and 3,264 (1,088×3, ~7 h) eight-channel utterances in the training, validation and test set, respectively.

We validate our algorithms on speech dereverberation using one, two and four microphones. We use the first microphone for the single-microphone task, the first and fifth for the two-microphone task, and the first, third, fifth and seventh for the four-microphone task. Note that the two- and four-microphone setups both have an aperture size of 20 cm. The first microphone is considered as the reference microphone for metric computation.

To evaluate the generalization ability of the trained models, we directly apply them to the recorded data of REVERB [77] for ASR. The recording device is an eight-microphone circular array with 10 cm radius. Note that the array geometry is subject to manufacturing error, which introduces a geometry mismatch between training and testing. The T60 is around 0.7 s and the speaker-to-array distance is 1 m in the near-field case and 2.5 m in the far-field case. We always consider the first microphone as the reference microphone. The ASR backend is built using the most recent Kaldi toolkit.

The network architectures follow the one depicted in Figure 6-3. The RI components of multiple microphones are stacked as feature maps for the network input and output. The window size is 32 ms and hop size 8 ms. The sampling rate is 16 kHz. A 512-point DFT is performed to extract 257-dimensional STFT features at each microphone.

Table 8-1. Average SI-SDR and PESQ of different methods on monaural dereverberation.

Methods	SI-SDR (dB)	PESQ
Unprocessed	-3.8	1.93
Estimated SMM	0.6	2.92
Estimated PSM	2.2	2.54
$\mathcal{L}_{\mathrm{RI}}$	6.1	2.79
$\mathcal{L}_{\mathrm{RI+Mag}}$	**6.5**	**3.10**
Oracle SMM ($T_0^{10}(\lvert S_q\rvert/\lvert Y_q\rvert)$)	1.5	3.39
Oracle PSM ($T_0^1(\lvert S_q\rvert\cos(\angle S_q - \angle Y_q)/\lvert Y_q\rvert)$)	4.4	3.09

Table 8-2. Average SI-SDR and PESQ of various methods on two- and four-channel dereverberation using simulated test data, and average WER (%) on REVERB real test data.

Metrics	SI-SDR (dB)			PESQ			WER on REVERB		
#mics	1	2	4	1	2	4	1	2	4
SISO_1	**6.5**	-	-	**3.10**	-	-	**9.62**	-	-
SISO_1-BF-SISO_1	-	8.0	9.4	-	3.20	3.29	-	8.37	7.63
SISO_1-BF-SISO_2	-	8.2	10.6	-	3.22	3.38	-	7.96	7.25
MISO_1	-	7.6	9.0	-	3.22	3.33	-	**7.38**	6.88
MISO_1-BF-MISO_2	-	8.6	**10.9**	-	3.24	**3.43**	-	**7.38**	**6.30**
MIMO	-	7.2	7.8	-	3.23	3.33	-	7.46	6.74
MIMO-BF-MISO_3	-	**8.7**	10.6	-	**3.28**	3.41	-	7.92	6.62
WPE	-	-	-	-	-	-	14.01	13.14	11.45
WPE+BeamformIt	-	-	-	-	-	-	-	12.64	9.30

8.5. Evaluation Results

Table 8-1 compares the performance of complex spectral mapping with real-valued masking on monaural dereverberation. Much better SI-SDR is obtained using complex spectral mapping based models over using estimated SMM and PSM. In addition, $\mathcal{L}_{\mathrm{RI+Mag}}$ leads to much better PESQ than $\mathcal{L}_{\mathrm{RI}}$, and slightly better SI-SDR. This indicates the importance of magnitude estimation when PESQ is used as the evaluation metric. The magnitude loss is always included for complex spectral mapping in the following experiments.

Table 8-2 first reports the enhancement performance of various multi-channel approaches. $SISO_1$ represents a baseline of monaural complex spectral mapping. In $SISO_1$-BF-$SISO_1$, we apply monaural complex spectral mapping on $\widehat{BF}_q$ to estimate target speech S_q, while in $SISO_1$-BF-$SISO_2$, complex spectral mapping is applied on the combination of $\widehat{BF}_q$ and Y_q to estimate S_q as in Figure 8-1(a). $SISO_1$-BF-$SISO_2$ produces better performance than $SISO_1$-BF-$SISO_1$ and $SISO_1$. We emphasize that $SISO_1$-BF-$SISO_1$ represents a typical beamforming followed by post-filtering approach in DNN based multi-channel speech enhancement [189]. In addition, both $MISO_1$ and MIMO are better than $SISO_1$. This indicates that concatenating multiple microphones for complex spectral mapping clearly helps. MIMO is worse than $MISO_1$, because producing multiple outputs is a harder task. Overall, $MISO_1$-BF-$MISO_2$ and MIMO-BF-$MISO_3$ perform the best. This is likely because MISO networks used for post-filtering can benefit from multi-microphone modeling.

In Table 8-2 we also evaluate the trained models in terms of ASR performance directly on the real test set of REVERB. Both $MISO_1$-BF-$MISO_2$ and MIMO-BF-$MISO_3$ exhibit strong generalization ability, and better ASR performance than $SISO_1$-BF-$SISO_1$ and $SISO_1$-BF-$SISO_2$, which are not sensitive to geometry mismatch. Clear improvements are also observed using the trained models over the baseline WPE [77] and WPE followed by BeamformIt algorithms, both available in Kaldi.

8.6. Conclusion

We have proposed a multi-microphone complex spectral mapping approach for speech dereverberation, and integrated it with beamforming and post-filtering into a unified

system. Experimental results suggest that on a fixed geometry, concatenating multiple microphone signals for complex spectral mapping is a simple and effective way of combining spectral and spatial information for speech dereverberation.

Chapter 9. Conclusions and Outlook

9.1. Contributions

Microphone array processing is essential in modern hands-free speech communitation such as speech enhacnement, speaker separation and robust ASR. In this dissertation, we have employed deep learning to improve robust speaker localization, acoustic beamforming, post-filtering, phase estimation, speech separation and robust ASR.

In Chapter 2, we have proposed to jointly train an frontend, a mel-filterbank and an acoustic model for robust ASR. We have explored several representitive noise- and reverberation-robust features for acoustic modeling, applied sequence-discriminative training for better seqeunce modeling, and conducted run-time unsupervised adaption to address the mismatches between training and testing. At the time of publication, these techniques together achieved the state-of-the-art performance on CHiME-2.

In Chapter 3, we have proposed three algorithms to utilize deep learning based T-F masking for robust speaker localization. Experimental results suggest that these algorithms dramatically improve conventional cross-correlation, beamforming and subspace based approaches for speaker localization in noisy-reverberant environments. In addition, our study finds that the ideal ratio mask can serve as a strong training target for robust speaker localization.

In Chapter 4, we have proposed a *Separate-Localize-Enhance* approach for deep learning based multi-channel blind speaker separation, where spatial features are combined with spectral features for DNN to extract target speech from an estimated direction and with particular spectral structure. This novel approach leads to large improvements over conventional methods and other DNN based algorithms that do not leverage spatial features for model training.

In Chapter 5, we have proposed multiple algorithms for monaural phase reconstruction based on magnitude estimates, based on a trigonometric perspective. The obtained state-of-the-art speaker separation results at the time of publication indicate that DNN based magnitude estimation can clearly help phase reconstruction. The proposed geometric constraint affords a mechanism to confine the possible solutions of phase. It could play a fundamental role in future research on phase estimation.

In Chapter 6, we have investigated a complex spectral mapping approach for phase estimation and proposed a target cancellation algorithm for multi-channel speech dereverberation. The trained single- and multi-channel models show clear improvements over single- and multi-channel WPE and other DNN based models. The improved phase produced by complex spectral mapping also leads to better beamforming. The trained models exhibit strong generalization ability to new and representative reverberant environments and array configurations.

In Chapter 7, we have applied single- and multi-channel complex spectral mapping for multi-channel speech enhancement. We have proposed a new and effective approach for time-varying beamforming. State-of-the-art performance has been obtained on the enhancement and recognition tasks of CHiME-4.

In Chapter 8, we have proposed a multi-microphone complex spectral mapping approach for speech dereverberation, and integrated it with acoustic beamforming and post-filtering. Experimental results indicate that, on a fixed geometry, concatenating multiple microphone signals for complex spectral mapping is an effective and simple way of integrating spectral and spatial information for robust speech processing.

Perhaps the most valuable insight I have gained in this dissertation study is that DNN based single-channel processing provides reliable signal statistics for spatial processing, even in environments with very strong noise and reverberation. By further combining such spatial processing and spectral processing using a DNN, we can integrate spectral and spatial cues for much better speech separation and recognition.

9.2. Future Work

This dissertation achieves large speech separation and ASR improvements over conventional and other DNN based algorithms. The proposed algorithms represent comprehensive solutions by exploiting spatial informaiton for modern speech communication, and have the potential to benefit numerous commercial speech applications. Here we put forth serveral directions for future research.

- *Online and time-varying beamforming*. This dissertation assumes offline processing scenarios and that the speakers are still within each utterance. To make the algorithms online, one can consider modifying DNN architectures causal by making them look at past observations only. In addition, the beamforming components can be made online by simply collecting statistics from past and current frames. This strategy could potentially deal with moving speakers.

- *Multi-channel multi-speaker separation in noisy and reverberant conditions.* Strong room reverberation and encironmental noise can drastically increase the difficulty of speaker separation. Future research could consider simultaneous speaker separation, denoising and dereverberation, which could be approached by using direct sound as the training target. An end-to-end system that optimizes all the modules could further elevate performance.
- *Phase estimation*. Phase estimation is a notoriously difficult but useful task in speech enhancement and dereverberation, and speaker separation. Using supervised learning where a model is trained to predict clean speech from a corrupted version, be it in the complex or time domain, might be fundamentally limited. Generative modeling could be a possible direction to produce more natural sounding, enhanced speech [101], [23].

Bibliography

[1] T. Afouras, J.S. Chung, and A. Zisserman, "The Conversation: Deep Audio-Visual Speech Enhancement," in *Proceedings of Interspeech*, 2018, pp. 3244–3248.

[2] X. Anguera and C. Wooters, "Acoustic Beamforming for Speaker Diarization of Meetings," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, pp. 2011–2022, 2007.

[3] S. Araki, H. Sawada, R. Mukai, and S. Makino, "DOA Estimation for Multiple Sparse Sources with Normalized Observation Vector Clustering," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2006, pp. 33–36.

[4] S. Araki, T. Hayashi, M. Delcroix, M. Fujimoto, K. Takeda, and T. Nakatani, "Exploring Multi-Channel Features for Denoising-Autoencoder-Based Speech Enhancement," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015, pp. 116–120.

[5] F. Bach and M. Jordan, "Learning Spectral Clustering, with Application to Speech Separation," *The Journal of Machine Learning Research*, vol. 7, pp. 1963–2001, 2006.

[6] D. Bagchi, M.I. Mandel, Z. Wang, Y. He, A. Plummer, and E. Fosler-Lussier, "Combining Spectral Feature Mapping and Multi-Channel Model-Based Source Separation for Noise-Robust Automatic Speech Recognition," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2016, pp. 496–503.

[7] S. Bai, J.Z. Kolter, and V. Koltun, "An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling," in *arXiv preprint arXiv:1803.01271*, 2018.

[8] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The Third 'CHiME' Speech Separation and Recognition Challenge: Dataset, Task and Baselines," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2015, pp. 504–511.

[9] C. Blandin, A. Ozerov, and E. Vincent, "Multi-Source TDOA Estimation in Reverberant Audio using Angular Spectra and Clustering," *Signal Processing*, vol. 92, pp. 1950–1960, 2012.

[10] S. Braun, B. Schwartz, S. Gannot, and E.A.P. Habets, "Late Reverberation PSD Estimation for Single-Channel Dereverberation using Relative Convolutive Transfer Functions," in *Proceedings of IWAENC*, 2016, pp. 1–5.

[11] S. Braun, A. Kuklasinski, O. Schwartz, O. Thiergart, E.A.P. Habets, S. Gannot, S. Doclo, and J. Jensen, "Evaluation and Comparison of Late Reverberation Power Spectral Density Estimators," *IEEE/ACM Transactions on Audio Speech and*

Language Processing, vol. 26, pp. 1056–1071, 2018.
[12] A.S. Bregman, *Auditory scene analysis: the perceptual organization of sound*. Cambridge, MA: MIT Press, 1990.
[13] S. Bu, Y. Zhao, M.-Y. Hwang, and S. Sun, "A Robust Nonlinear Microphone Array Postfilter for Noise Reduction," in *Proceedings of IWAENC*, 2018, pp. 206–210.
[14] G.C. Carter, A.H. Nuttall, and P.G. Cable, "The Smoothed Coherence Transform," *Proceedings of the IEEE*, vol. 61, pp. 1497–1498, 1973.
[15] S. Chakrabarty and E.A.P. Habets, "Broadband DOA Estimation using Convolutional Neural Networks Trained with Noise Signals," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2017, pp. 136–140.
[16] S. Chakrabarty and E.A.P. Habets, "Multi-Speaker DOA Estimation using Deep Convolutional Networks Trained with Noise Signals," *IEEE Journal on Selected Topics in Signal Processing*, vol. 13, pp. 8–21, 2019.
[17] J. Chen, Y. Wang, and D.L. Wang, "A Feature Study for Classification-Based Speech Separation at Low Signal-to-Noise Ratios," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, pp. 1993–2002, 2014.
[18] S.-J. Chen, A. S. Subramanian, H. Xu, and S. Watanabe, "Building State-of-The-Art Distant Speech Recognition using The CHiME-4 Challenge with A Setup of Speech Enhancement Baseline," in *Proceedings of Interspeech*, 2018, pp. 1571–1575.
[19] Z. Chen, S. Watanabe, H. Erdogan, and J.R. Hershey, "Speech Enhancement and Recognition using Multi-task Learning of Long Short-term Memory Recurrent Neural Networks," in *Proceedings of Interspeech*, 2015.
[20] Z. Chen, Y. Luo, and N. Mesgarani, "Deep Attractor Network for Single-Microphone Speaker Separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 246–250.
[21] Z. Chen, J. Li, X. Xiao, T. Yoshioka, H. Wang, Z. Wang, and Y. Gong, "Cracking The Cocktail Party Problem by Multi-Beam Deep Attractor Network," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2017, pp. 437–444.
[22] Z. Chen, T. Yoshioka, X. Xiao, J. Li, M.L. Seltzer, and Y. Gong, "Efficient Integration of Fixed Beamformers and Speech Separation Networks for Multi-Channel Far-Field Speech Separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 5384–5388.
[23] M. Chinen, W.B. Kleijn, F.S.C. Lim, and J. Skoglund, "Generative Speech Enhancement Based on Cloned Networks," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2019.
[24] C. Darwin, "Listening to Speech in The Presence of Other Sounds," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 363, pp. 1011–1021, 2008.
[25] M. Delcroix, Y. Kubo, T. Nakatani, and A. Nakamura, "Is Speech Enhancement Pre-Processing Still Relevant When using Deep Neural Networks for Acoustic Modeling?," in *Proceedings of Interspeech*, 2013, pp. 2992–2996.
[26] M. Delcroix, T. Yoshioka, A. Ogawa, Y. Kubo, M. Fujimoto, N. Ito, K. Kinoshita, M. Espi, S. Araki, T. Hori, and T. Nakatani, "Strategies for Distant Speech Recognition in Reverberant Environments," *Eurasip Journal on Advances in Signal*

Processing, 2015.
[27] L. Deng, D. Yu, and J. Platt, "Scalable Stacking and Learning for Building Deep Architectures," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2012, pp. 2133–2136.
[28] J. DiBiase, H. Silverman, and M. Brandstein, "Robust Localization in Reverberant Rooms," in *Microphone Arrays*, Berlin Heidelberg: Springer, 2001, pp. 157–180.
[29] L. Drude and R. Haeb-Umbach, "Tight Integration of Spatial and Spectral Features for BSS with Deep Clustering Embeddings," in *Proceedings of Interspeech*, 2017, pp. 2650–2654.
[30] L. Drude, J. Heymann, C. Boeddeker, and R. Haeb-Umbach, "NARA-WPE: A Python Package for Weighted Prediction Error Dereverberation in Numpy and Tensorflow for Online and Offline Processing," in *ITG conference on Speech Communication*, 2018.
[31] J. Du, Q. Wang, T. Gao, and Y. Xu, "Robust Speech Recognition with Speech Enhanced Deep Neural Networks," in *Proceedings of Interspeech*, 2014, pp. 616–620.
[32] J. Du, Y. Tu, L. Sun, F. Ma, H. Wang, and J. Pan, "The USTC–iFlytek System for CHiME-4 Challenge," in *Proceedings of CHiME-4*, 2016, pp. 36–38.
[33] N.Q.K. Duong, E. Vincent, and R. Gribonval, "Under-Determined Reverberant Audio Source Separation using A Full-Rank Spatial Covariance Model," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, pp. 1830–1840, 2010.
[34] J. Eaton, N.D. Gaubitch, A.H. Moore, and P.A. Naylor, "The ACE Challenge - Corpus Description and Performance Evaluation," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2015.
[35] H. Erdogan, J.R. Hershey, S. Watanabe, and J. Le Roux, "Phase-Sensitive and Recognition-Boosted Speech Separation using Deep Recurrent Neural Networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015, pp. 708–712.
[36] H. Erdogan, J.R. Hershey, S. Watanabe, M.I. Mandel, and J. Le Roux, "Improved MVDR Beamforming using Single-Channel Mask Prediction Networks," in *Proceedings of Interspeech*, 2016, vol. 08-12-Sept, pp. 1981–1985.
[37] O. Ernst, S.E. Chazan, S. Gannot, and J. Goldberger, "Speech Dereverberation using Fully Convolutional Networks," in *European Signal Processing Conference*, 2018, pp. 390–394.
[38] E.L. Ferguson, S.B. Williams, and C.T. Jin, "Sound Source Localization in a Multipath Environment using Convolutional Neural Networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 2386–2390.
[39] S.-W. Fu, T.-Y. Hu, Y. Tsao, and X. Lu, "Complex Spectrogram Enhancement By Convolutional Neural Network with Multi-Metrics Learning," in *IEEE International Workshop on Machine Learning for Signal Processing*, 2017, pp. 1–6.
[40] S. Gannot, E. Vincent, S. Markovich-Golan, and A. Ozerov, "A Consolidated Perspective on Multi-Microphone Speech Enhancement and Source Separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, pp.

692–730, 2017.
[41] T. Gao, J. Du, L.-R. Dai, and C.-H. Lee, "Joint Training of Front-End and Back-End Deep Neural Networks for Robust Speech Recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015, pp. 4375–4379.
[42] T. Gerkmann, M. Krawczyk-Becker, and J. Le Roux, "Phase Processing for Single-Channel Speech Enhancement: History and Recent Advances," *IEEE Signal Processing Magazine*, vol. 32, pp. 55–66, 2015.
[43] X. Glorot, A. Bordes, and Y. Bengio, "Deep Sparse Rectifier Neural Networks," in *International Conference on Artificial Intelligence and Statistics*, 2011, pp. 315–323.
[44] D.W. Griffin and J.S. Lim, "Signal Estimation from Modified Short-Time Fourier Transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, pp. 236–243, 1984.
[45] F. Grondin and F. Michaud, "Time Difference of Arrival Estimation based on Binary Frequency Mask for Sound Source Localization on Mobile Robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2015, pp. 6149–6154.
[46] D. Gunawan and D. Sen, "Iterative Phase Estimation for the Synthesis of Separated Sources from Single-Channel Mixtures," in *IEEE Signal Processing Letters*, 2010, pp. 421–424.
[47] E.A.P. Habets, "Room Impulse Response Generator," 2010.
[48] E.A.P. Habets, S. Gannot, and I. Cohen, "Late Reverberant Spectral Variance Estimation Based on A Statistical Model," *IEEE Signal Processing Letters*, vol. 16, pp. 770–774, 2009.
[49] E.A.P. Habets and P.A. Naylor, "Dereverberation," in *Audio Source Separation and Speech Enhancement*, E. Vincent, T. Virtanen, and S. Gannot, Eds. Wiley, 2018, pp. 317–343.
[50] E. Hadad, F. Heese, P. Vary, and S. Gannot, "Multichannel Audio Database in Various Acoustic Environments," in *International Workshop on Acoustic Signal Enhancement*, 2014, pp. 313–317.
[51] R. Haeb-Umbach, S. Watanabe, T. Nakatani, M. Bacchiani, B. Hoffmeister, M.L. Seltzer, H. Zen, and M. Souden, "Speech Processing for Digital Home Assistants: Combining Signal Processing with Deep-Learning Techniques," *IEEE Signal Processing Magazine*, vol. 36, pp. 111–124, 2019.
[52] K. Han, Y. Wang, D.L. Wang, W.S. Woods, and I. Merks, "Learning Spectral Mapping for Speech Dereverberation and Denoising," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, pp. 982–992, 2015.
[53] K. Han, Y. He, D. Bagchi, E. Fosler-lussier, and D.L. Wang, "Deep Neural Network Based Spectral Feature Mapping for Robust Speech Recognition," in *Proceedings of Interspeech*, 2015, pp. 2484–2488.
[54] W.M. Hartmann, "How We Localize Sound," *Physics Today*, vol. 52, pp. 24–29, 1999.
[55] E. Healy, S. Yoho, Y. Wang, and D.L. Wang, "An Algorithm to Improve Speech Recognition in Noise for Hearing-Impaired Listeners," *The Journal of the Acoustical Society of America*, vol. 23, pp. 3029–3038, 2013.
[56] H. Hermansky and N. Morgan, "RASTA Processing of Speech," *IEEE Transactions on Speech and Audio Processing*, vol. 2, pp. 578–589, 1994.

[57] J.R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep Clustering: Discriminative Embeddings for Segmentation and Separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016, pp. 31–35.
[58] J. Heymann, L. Drude, A. Chinaev, and R. Haeb-Umbach, "BLSTM Supported GEV Beamformer Front-End for the 3rd CHiME Challenge," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2015, pp. 444–451.
[59] J. Heymann, L. Drude, R. Haeb-Umbach, K. Kinoshita, and T. Nakatani, "Frame-Online DNN-WPE Dereverberation," in *Proceedings of IWAENC*, 2018, pp. 466–470.
[60] J. Heymann, L. Drude, R. Haeb-Umbach, K. Kinoshita, and T. Nakatani, "Joint Optimization of Neural Network-Based WPE Dereverberation and Acoustic Model for Robust Online ASR," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 6655–6659.
[61] J. Heymann and R. Haeb-Umbach, "Wide Residual BLSTM Network with Discriminative Speaker Adaptation for Robust Speech Recognition," in *Proceedings of CHiME-4*, 2016.
[62] T. Higuchi, K. Kinoshita, M. Delcroix, K. Zmolkova, and T. Nakatani, "Deep Clustering-Based Beamforming for Separation with Unknown Number of Sources," in *Proceedings of Interspeech*, 2017, pp. 1183–1187.
[63] T. Higuchi, N. Ito, S. Araki, T. Yoshioka, M. Delcroix, and T. Nakatani, "Online MVDR Beamformer Based on Complex Gaussian Mixture Model with Spatial Prior for Noise Robust ASR," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, pp. 780–793, 2017.
[64] T. Higuchi, N. Ito, T. Yoshioka, and T. Nakatani, "Robust MVDR Beamforming using Time-frequency Masks for Online/Offline ASR in Noise," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016, pp. 5210–5214.
[65] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups," *IEEE Signal Processing Magazine*, vol. 29, pp. 82–97, 2012.
[66] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, pp. 1735–1780, 1997.
[67] G. Huang, Z. Liu, L.V.D. Maaten, and K.Q. Weinberger, "Densely Connected Convolutional Networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
[68] IEEE, "IEEE Recommended Practice for Speech Quality Measurements," *IEEE Transactions on Audio and Electroacoustics*, vol. 17, pp. 225–246, 1969.
[69] Y. Isik, J. Le Roux, Z. Chen, S. Watanabe, and J.R. Hershey, "Single-Channel Multi-Speaker Separation using Deep Clustering," in *Proceedings of Interspeech*, 2016, pp. 545–549.
[70] N. Ito, S. Araki, and T. Nakatani, "Recent Advances in Multichannel Source Separation and Denoising Based on Source Sparseness," in *Audio Source Separation*, 2018, pp. 279–300.
[71] A. Jansson, E. Humphrey, N. Montecchio, R. Bittner, A. Kumar, and T. Weyde,

"Singing Voice Separation with Deep U-Net Convolutional Networks," in *Proceedings of ISMIR*, 2017, pp. 745–751.

[72] J. Jensen and C.H. Taal, "An Algorithm for Predicting The Intelligibility of Speech Masked by Modulated Noise Maskers," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, pp. 2009–2022, 2016.

[73] Y. Jiang, D.L. Wang, R. Liu, and Z. Feng, "Binaural Classification for Reverberant Speech Segregation using Deep Neural Networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, pp. 2112–2121, 2014.

[74] C. Kim and R.M. Stern, "Power-Normalized Cepstral Coefficients (PNCC) for Robust Speech Recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2012, pp. 4101–4104.

[75] B. Kingsbury, "Lattice-Based Optimization of Sequence Classification Criteria for Neural-Network Acoustic Modeling," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009, pp. 3761–3764.

[76] K. Kinoshita, M. Delcroix, H. Kwon, T. Mori, and T. Nakatani, "Neural Network-Based Spectrum Estimation for Online WPE Dereverberation," in *Proceedings of Interspeech*, 2017, pp. 384–388.

[77] K. Kinoshita, M. Delcroix, S. Gannot, E.A.P. Habets, R. Haeb-Umbach, W. Kellermann, V. Leutnant, R. Maas, T. Nakatani, B. Raj, A. Sehr, and T. Yoshioka, "A Summary of the REVERB Challenge: State-of-The-Art and Remaining Challenges in Reverberant Speech Processing Research," *Eurasip Journal on Advances in Signal Processing*, pp. 1–19, 2016.

[78] D. Kitamura, N. Ono, H. Sawada, H. Kameoka, and H. Saruwatari, "Determined Blind Source Separation with Independent Low-Rank Matrix Analysis," in *Audio Source Separation*, 2018, pp. 125–155.

[79] U. Kjems and J. Jensen, "Maximum Likelihood Based Noise Covariance Matrix Estimation for Multi-Microphone Speech Enhancement," in *European Signal Processing Conference*, 2012, pp. 295–299.

[80] C. Knapp and G. Carter, "The Generalized Correlation Method for Estimation of Time Delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, pp. 320–327, 1976.

[81] M. Kolbæk, D. Yu, Z.-H. Tan, and J. Jensen, "Multi-Talker Speech Separation with Utterance-Level Permutation Invariant Training of Deep Recurrent Neural Networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, pp. 1901–1913, 2017.

[82] B. Kollmeier and R. Koch, "Speech Enhancement Based on Physiological and Psychoacoustical Models of Modulation Perception and Binaural Interaction," *The Journal of the Acoustical Society of America*, vol. 95, pp. 1593–1602, 1994.

[83] H. Krim and M. Viberg, "Two Decades of Array Signal Processing Research: the Parametric Approach," *IEEE Signal Processing Magazine*, vol. 13, pp. 67–94, 1996.

[84] V. Krishnaveni, T. Kesavamurthy, and B. Aparna, "Beamforming for Direction-of-Arrival (DOA) Estimation - A Survey," *International Journal of Computer Applications*, vol. 61, pp. 975–8887, 2013.

[85] Y. Kubo, T. Nakatani, M. Delcroix, K. Kinoshita, and S. Araki, "Mask-Based MVDR Beamformer for Noisy Multisource Environments: Introduction of Time-Varying Spatial Covariance Model," in *IEEE International Conference on*

Acoustics, Speech and Signal Processing, 2019, pp. 6855–6859.
[86] J. Le Roux, N. Ono, and S. Sagayama, "Explicit Consistency Constraints for STFT Spectrograms and Their Application to Phase Reconstruction," *Proceedings of SAPA*, 2008.
[87] J. Le Roux, G. Wichern, S. Watanabe, A. Sarroff, and J.R. Hershey, "Phasebook and Friends: Leveraging Discrete Representations for Source Separation," *IEEE Journal on Selected Topics in Signal Processing*, vol. 13, pp. 370–382, 2019.
[88] J. Le Roux, S. Wisdom, H. Erdogan, and J.R. Hershey, "SDR – Half-Baked or Well Done?," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 626–630.
[89] F. Li, P. Nidadavolu, and H. Hermansky, "A Long, Deep and Wide Artificial Neural Net for Robust Speech Recognition in Unknown Noise," in *Proceedings of Interspeech*, 2014.
[90] C.-L. Liu, S.-W. Fu, Y.-J. Li, J.-W. Huang, H.-M. Wang, and Y. Tsao, "Multichannel Speech Enhancement by Raw Waveform-Mapping using Fully Convolutional Networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2020.
[91] Y. Liu, A. Ganguly, K. Kamath, and T. Kristjansson, "Neural Network Based Time-Frequency Masking and Steering Vector Estimation for Two-Channel MVDR Beamforming," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 6717–6721.
[92] Y. Liu and D.L. Wang, "Divide and Conquer: A Deep CASA Approach to Talker-Independent Monaural Speaker Separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, pp. 2092–2102, 2019.
[93] P.C. Loizou, *Speech enhancement: theory and practice*. CRC press, 2013.
[94] Y. Luo, Z. Chen, and N. Mesgarani, "Speaker-Independent Speech Separation with Deep Attractor Network," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, pp. 787–796, 2018.
[95] Y. Luo and N. Mesgarani, "TasNet: Surpassing Ideal Time-Frequency Masking for Speech Separation," *arXiv preprint arXiv:1809.07454v2*, 2018.
[96] Y. Luo and N. Mesgarani, "Real-Time Single-Channel Dereverberation and Separation with Time-Domain Audio Separation Network," in *Proceedings of Interspeech*, 2018, pp. 342–346.
[97] Y. Luo and N. Mesgarani, "TasNet: Time-Domain Audio Separation Network for Real-Time, Single-Channel Speech Separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 697–700.
[98] N. Ma, G.J. Brown, and T. May, "Exploiting Deep Neural Networks and Head Movements for Binaural Localisation of Multiple Speakers in Reverberant Conditions," in *Proceedings of Interspeech*, 2015, pp. 160–164.
[99] N. Ma, T. May, and G.J. Brown, "Exploiting Deep Neural Networks and Head Movements for Robust Binaural Localization of Multiple Sources in Reverberant Environments," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, pp. 2444–2453, 2017.
[100] W. Mack, S. Chakrabarty, F.-R. Stöter, S. Braun, B. Edler, and E.A.P. Habets, "Single-Channel Dereverberation using Direct MMSE Optimization and Bidirectional LSTM Networks," in *Proceedings of Interspeech*, 2018, pp. 1314–

1318.
[101] S. Maiti and M.I. Mandel, “Parametric Resynthesis with Neural Vocoders,” in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2019.
[102] M.I. Mandel, R.J. Weiss, and D.P.W. Ellis, “Model-Based Expectation-Maximization Source Separation and Localization,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, pp. 382–394, 2010.
[103] M.I. Mandel and J.P. Barker, “Multichannel Spatial Clustering using Model-Based Source Separation,” in *New Era for Robust Speech Recognition Exploiting Deep Learning*, 2017, pp. 51–78.
[104] M. Mimura, S. Sakai, and T. Kawahara, “Speech Dereverberation using Long Short-Term Memory,” in *Proceedings of Interspeech*, 2015, pp. 2435–2439.
[105] S. Mirsamadi and J. Hansen, “A Study on Deep Neural Network Acoustic Model Adaptation for Robust Far-Field Speech Recognition,” in *Proceedings of Interspeech*, 2015, pp. 2430–2434.
[106] S. Mohan, M.E. Lockwood, M.L. Kramer, and D.L. Jones, “Localization of Multiple Acoustic Sources with Small Arrays using a Coherence Test,” *The Journal of the Acoustical Society of America*, vol. 123, pp. 2136–2147, 2008.
[107] P. Mowlaee, R. Saeidi, and R. Martin, “Phase Estimation for Signal Reconstruction in Single-Channel Speech Separation,” in *Proceedings of Interspeech*, 2012.
[108] P. Mowlaee and R. Saeidi, “Time-Frequency Constraints for Phase Estimation in Single-Channel Speech Enhancement,” in *Proceedings of IWAENC*, 2014, pp. 337–341.
[109] P. Mowlaee, R. Saeidi, and Y. Stylianou, “Advances in Phase-Aware Signal Processing in Speech Communication,” *Speech Communication*, vol. 81, pp. 1–29, 2016.
[110] H.A. Murthy and B. Yegnanarayana, “Speech Processing using Group Delay Functions,” *Signal Processing*, vol. 22, pp. 259–267, 1991.
[111] J. Muth, S. Uhlich, N. Perraudin, T. Kemp, F. Cardinaux, and Y. Mitsufuji, “Improving DNN-based Music Source Separation using Phase Features,” in *arXiv preprint arXiv:1807.02710*, 2018.
[112] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H.F. Juang, “Speech Dereverberation Based on Variance-Normalized Delayed Linear Prediction,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, pp. 1717–1731, 2010.
[113] A. Narayanan and D.L. Wang, “Ideal Ratio Mask Estimation using Deep Neural Networks for Robust Speech Recognition,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 7092–7096.
[114] A. Narayanan and D.L. Wang, “Investigation of Speech Separation as a Front-end for Noise Robust Speech Recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, pp. 826–835, 2014.
[115] A. Narayanan and D.L. Wang, “Joint noise adaptive training for robust automatic speech recognition,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 2504–2508.
[116] A. Narayanan and D.L. Wang, “Improving Robustness of Deep Neural Network Acoustic Models via Speech Separation and Joint Adaptive Training,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, pp. 92–101,

2015.
[117] A. Pandey and D.L. Wang, "A New Framework for CNN-Based Speech Enhancement in the Time Domain," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, pp. 1179–1188, 2019.
[118] P. Pertilä and J. Nikunen, "Distant Speech Separation using Predicted Time-Frequency Masks from Spatial Features," *Speech Communication*, vol. 68, pp. 97–106, 2015.
[119] P. Pertilä and E. Cakir, "Robust Direction Estimation with Convolutional Neural Networks based Steered Response Power," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 6125–6129.
[120] W. Ping, K. Peng, and J. Chen, "ClariNet: Parallel Wave Generation in End-to-End Text-to-Speech," in *Proceedings of ICLR*, 2019.
[121] D. Povey, A. Ghoshal, and G. Boulianne, "The Kaldi Speech Recognition Toolkit," 2011.
[122] Y.-M. Qian, C. Weng, X. Chang, S. Wang, and D. Yu, "Past Review, Current Progress, and Challenges Ahead on the Cocktail Party Problem," *Frontiers of Information Technology & Electronic Engineering*, pp. 40–63, 2018.
[123] A.W. Rix, J.G. Beerends, M.P. Hollier, and A.P. Hekstra, "Perceptual Evaluation of Speech Quality (PESQ)-A New Method for Speech Quality Assessment of Telephone Networks and Codecs," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2001, vol. 2, pp. 749–752.
[124] N. Roman, D.L. Wang, and G.J. Brown, "Speech Segregation Based on Sound Localization," *The Journal of the Acoustical Society of America*, vol. 114, pp. 2236–2252, 2003.
[125] N. Roman, S. Srinivasan, and D.L. Wang, "Binaural Segregation in Multisource Reverberant Environments," *The Journal of the Acoustical Society of America*, vol. 120, pp. 4040–4051, 2006.
[126] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Proceedings of MICCAI*, 2015.
[127] Y. Rui and D. Florencio, "Time Delay Estimation in the Presence of Correlated Noise and Reverberation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2004, pp. 133–136.
[128] T.N. Sainath, B. Kingsbury, A. Mohamed, and B. Ramabhadran, "Learning Filterbanks within A Deep Neural Network Framework," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2013, pp. 297–302.
[129] G. Saon and H. Soltau, "Speaker Adaptation of Neural Network Acoustic Models using I-Vectors," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2013, pp. 55–59.
[130] A. Sarrof, "Complex Neural Networks for Audio," Dartmouth College, 2018.
[131] H. Sawada, S. Araki, and S. Makino, "Underdetermined Convolutive Blind Source Separation via Frequency Bin-Wise Clustering and Permutation Alignment," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, pp. 516–527, 2011.
[132] H. Sawada, S. Araki, and S. Makino, "A Two-Stage Frequency-Domain Blind Source Separation Method for Underdetermined Convolutive Mixtures," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2007, pp.

139–142.
[133] J. Schmidhuber, “Deep Learning in Neural Networks: An Overview,” *Neural Networks*, vol. 61, pp. 85–117, 2015.
[134] R. Schmidt, “Multiple Emitter Location and Signal Parameter Estimation,” *IEEE Transactions on Antennas and Propagation*, vol. 34, pp. 276–280, 1986.
[135] F. Seide, G. Li, X. Chen, and D. Yu, “Feature Engineering in Context-dependent Deep Neural Networks for Conversational Speech Transcription,” in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2011, pp. 24–29.
[136] M.L. Seltzer, I. Tashev, and T. Ivan, “A Log-MMSE Adaptive Beamformer using A Nonlinear Spatial Filter,” in *Proceedings of IWAENC*, 2008.
[137] M.L. Seltzer, D. Yu, and Y. Wang, “An Investigation of Deep Neural Networks for Noise Robust Speech Recognition,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 7398–7402.
[138] U. Shaham, K. Stanton, H. Li, B. Nadler, R. Basri, and Y. Kluger, “SpectralNet: Spectral Clustering using Deep Neural Networks,” in *Proceedings of ICLR*, 2018.
[139] K. Shimada, Y. Bando, M. Mimura, K. Itoyama, K. Yoshii, and T. Kawahara, “Unsupervised Speech Enhancement Based on Multichannel NMF-Informed Beamforming for Noise-Robust Automatic Speech Recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, pp. 960–971, 2019.
[140] S. Shimauchi, S. Kudo, Y. Koizumi, and K. Furuya, “On Relationships Between Amplitude and Phase of Short-Time Fourier Transform,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 676–680.
[141] D. Stoller, S. Ewert, and S. Dixon, “Wave-U-Net: A Multi-Scale Neural Network for End-to-End Audio Source Separation,” in *Proceedings of ISMIR*, 2018, pp. 334–340.
[142] F.-R. Stöter, A. Liutkus, and N. Ito, “The 2018 Signal Separation Evaluation Campaign,” in *International Conference on Latent Variable Analysis and Signal Separation*, 2018, pp. 293–305.
[143] A.S. Subramanian, X. Wang, S. Watanabe, T. Taniguchi, D. Tran, and Y. Fujita, “An Investigation of End-to-End Multichannel Speech Recognition for Reverberant and Mismatch Conditions,” in *arXiv preprint arXiv:1904.09049*, 2019.
[144] N. Takahashi, N. Goswami, and Y. Mitsufuji, “MMDenseLSTM: An Efficient Combination of Convolutional and Recurrent Neural Networks for Audio Source Separation,” in *Proceedings of IWAENC*, 2018, pp. 106–110.
[145] S. Takamichi, Y. Saito, N. Takamune, D. Kitamura, and H. Saruwatari, “Phase Reconstruction from Amplitude Spectrograms Based on von-Mises-Distribution Deep Neural Network,” in *Proceedings of IWAENC*, 2018.
[146] K. Tan and D.L. Wang, “Learning Complex Spectral Mapping With Gated Convolutional Recurrent Networks for Monaural Speech Enhancement,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 380–390, 2020.
[147] K. Tan and D.L. Wang, “A Convolutional Recurrent Neural Network for Real-Time Speech Enhancement,” in *Proceedings of Interspeech*, 2018, pp. 3229–3233.
[148] K. Tan and D.L. Wang, “Complex Spectral Mapping with A Convolutional Recurrent Network for Monaural Speech Enhancement,” in *IEEE International*

Conference on Acoustics, Speech and Signal Processing, 2019, vol. 2019-May, pp. 6865–6869.

[149] I. Tashev and A. Acero, "Microphone Array Post-Processor using Instantaneous Direction of Arrival," in *Proceedings of IWAENC*, 2006.

[150] N. Tawara, T. Kobayashi, and T. Ogawa, "Multi-Channel Speech Enhancement using Time-Domain Convolutional Denoising Autoencoder," in *Proceedings of Interspeech*, 2019, pp. 86–90.

[151] N.T.N. Tho, S. Zhao, and D.L. Jones, "Robust DOA Estimation of Multiple Speech Sources," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 2287–2291.

[152] J. Traa, M. Kim, and P. Smaragdis, "Phase and Level Difference Fusion for Robust Multichannel Source Separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 6687–6691.

[153] Y.-H. Tu, J. Du, L. Sun, F. Ma, H.-K. Wang, J.-D. Chen, and C.-H. Lee, "An Iterative Mask Estimation Approach to Deep Learning Based Multi-Channel Speech Recognition," *Speech Communication*, vol. 106, pp. 31–43, 2019.

[154] Y.-H. Tu, J. Du, N. Zhou, and C.-H. Lee, "Online LSTM-Based Iterative Mask Estimation for Multi-Channel Speech Enhancement and ASR," in *Annual Summit and Conference on Signal and Information Processing*, 2018, pp. 362–366.

[155] J.-M. Valin, F. Michaud, J. Rouat, and D. Letourneau, "Robust Sound Source Localization using A Microphone Array on A Mobile Robot," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2003, vol. 2, pp. 1228–1233.

[156] J.-M. Valin, F. Michaud, and J. Rouat, "Robust Localization and Tracking of Simultaneous Moving Sound Sources using Beamforming and Particle Filtering," *Robotics and Autonomous Systems*, vol. 55, pp. 216–228, 2007.

[157] B.D. Van Veen and K.M. Buckley, "Beamforming: A Versatile Approach to Spatial Filtering," *IEEE ASSP Magazine*, vol. 5, pp. 4–24, 1988.

[158] K. Veselý, A. Ghoshal, L. Burget, and D. Povey, "Sequence-Discriminative Training of Deep Neural Networks," in *Proceedings of Interspeech*, 2013, pp. 2345–2349.

[159] E. Vincent, J.P. Barker, S. Watanabe, J. Le Roux, F. Nesta, and M. Matassoni, "The Second 'CHiME' Speech Separation and Recognition Challenge: An Overview of Challenge Systems and Outcomes," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2013, pp. 162–167.

[160] E. Vincent, S. Watanabe, A.A. Nugraha, J. Barker, and R. Marxer, "An Analysis of Environment, Microphone and Data Simulation Mismatches in Robust Speech Recognition," *Computer Speech and Language*, vol. 46, pp. 535–557, 2017.

[161] D.L. Wang and J. Chen, "Supervised Speech Separation Based on Deep Learning: An Overview," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, pp. 1702–1726, 2018.

[162] D.L. Wang, "On Ideal Binary Mask as the Computational Goal of Auditory Scene Analysis," in *Speech Separation by Humans and Machines*, P. Divenyi, Eds., Spinger, 2005, pp. 181–197.

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

[164] P. Wang and D.L. Wang, "Utterance-Wise Recurrent Dropout and Iterative Speaker Adaptation for Robust Monaural Speech Recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 4814–4818.
[165] Y. Wang and D.L. Wang, "Towards Scaling Up Classification-Based Speech Separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, pp. 1381–1390, 2013.
[166] Y. Wang, A. Narayanan, and D.L. Wang, "On Training Targets for Supervised Speech Separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, pp. 1849–1858, 2014.
[167] Y. Wang, A. Misra, and K. Chin, "Time-Frequency Masking for Large Scale Robust Speech Recognition," in *Proceedings of Interspeech*, 2015, pp. 2469–2473.
[168] Y. Wang, J. Chen, and D.L. Wang, "Deep Neural Network Based Supervised Speech Segregation Generalizes to Novel Noises Through Large-Scale Training," *OSU-CISRC-3/15-TR02*, 2015.
[169] Y. Wang, K. Han, and D.L. Wang, "Exploring Monaural Features for Classification-based Speech Segregation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, pp. 270–279, 2013.
[170] Y. Wang and D.L. Wang, "A Deep Neural Network for Time-Domain Signal Reconstruction," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015, pp. 4390–4394.
[171] Z.-Q. Wang and D.L. Wang, "Joint Training of Speech Separation, Filterbank and Acoustic Model for Robust Automatic Speech Recognition," in *Proceedings of Interspeech*, 2015, pp. 2839–2843.
[172] Z.-Q. Wang and D.L. Wang, "A Joint Training Framework for Robust Automatic Speech Recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, pp. 796–806, 2016.
[173] Z.-Q. Wang, Y. Zhao, and D.L. Wang, "Phoneme-Specific Speech Separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016, pp. 146–150.
[174] Z.-Q. Wang, X. Zhang, and D.L. Wang, "Robust TDOA Estimation Based on Time-Frequency Masking and Deep Neural Networks," in *Proceedings of Interspeech*, 2018, pp. 322–326.
[175] Z.-Q. Wang, X. Zhang, and D.L. Wang, "Robust Speaker Localization Guided by Deep Learning Based Time-Frequency Masking," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, pp. 178–188, 2019.
[176] Z.-Q. Wang and D.L. Wang, "Recurrent Deep Stacking Networks for Supervised Speech Separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 71–75.
[177] Z.-Q. Wang, J. Le Roux, and J.R. Hershey, "Alternative Objective Functions for Deep Clustering," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 686–690.
[178] Z.-Q. Wang, J. Le Roux, and J.R. Hershey, "Multi-Channel Deep Clustering: Discriminative Spectral and Spatial Embeddings for Speaker-Independent Speech Separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 1–5.
[179] Z.-Q. Wang and D.L. Wang, "Integrating Spectral and Spatial Features for Multi-

Channel Speaker Separation," in *Proceedings of Interspeech*, 2018, pp. 2718–2722.
[180] Z.-Q. Wang and D.L. Wang, "Combining Spectral and Spatial Features for Deep Learning Based Blind Speaker Separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, pp. 457–468, 2019.
[181] Z.-Q. Wang, J. Le Roux, D.L. Wang, and J.R. Hershey, "End-to-End Speech Separation with Unfolded Iterative Phase Reconstruction," in *Proceedings of Interspeech*, 2018, pp. 2708–2712.
[182] Z.-Q. Wang and D.L. Wang, "Mask-Weighted STFT Ratios for Relative Transfer Function Estimation and Its Application to Robust ASR," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 5619–5623.
[183] Z.-Q. Wang and D.L. Wang, "On Spatial Features for Supervised Speech Separation and Its Application to Beamforming and Robust ASR," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 5709–5713.
[184] Z.-Q. Wang, K. Tan, and D.L. Wang, "Deep Learning Based Phase Reconstruction for Speaker Separation: A Trigonometric Perspective," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 71–75.
[185] Z.-Q. Wang and D.L. Wang, "Deep Learning Based Target Cancellation for Speech Dereverberation," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 2020.
[186] Z.-Q. Wang and D.L. Wang, "Unsupervised Speaker Adaptation of Batch Normalized Acoustic Models for Robust ASR," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 4890–4894.
[187] Z.-Q. Wang, P. Wang, and D.L. Wang, "Complex Spectral Mapping for Single- and Multi-Channel Speech Enhancement and Robust ASR," *in submission to IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2019.
[188] Z.-Q. Wang and D.L. Wang, "Multi-Microphone Complex Spectral Mapping for Speech Dereverberation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 486–490.
[189] Z.-Q. Wang and D.L. Wang, "All-Neural Multi-Channel Speech Enhancement," in *Proceedings of Interspeech*, 2018, pp. 3234–3238.
[190] C. Weng, D. Yu, S. Watanabe, and B.-H.F. Juang, "Recurrent Deep Neural Networks for Robust Speech Recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 5532–5536.
[191] 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 *International Conference on Latent Variable Analysis and Signal Separation*, 2015, pp. 91–99.
[192] D.S. Williamson, Y. Wang, and D.L. Wang, "Complex Ratio Masking for Monaural Speech Separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pp. 483–492, 2016.
[193] D.S. Williamson and D.L. Wang, "Time-Frequency Masking in The Complex Domain for Speech Dereverberation and Denoising," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, pp. 1492–1501, 2017.
[194] S. Wisdom, J.R. Hershey, K. Wilson, J. Thorpe, M. Chinen, B. Patton, and R.A. Saurous, "Differentiable Consistency Constraints for Improved Deep Speech Enhancement," in *IEEE International Conference on Acoustics, Speech and Signal*

Processing, 2019, vol. 2019, pp. 900–904.
[195] S.U.N. Wood, J. Rouat, S. Dupont, and G. Pironkov, "Blind Speech Separation and Enhancement with GCC-NMF," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, pp. 745–755, 2017.
[196] J. Woodruff and D.L. Wang, "Binaural Localization of Multiple Sources in Reverberant and Noisy Environments," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, pp. 1503–1512, 2012.
[197] B. Wu, M. Yang, K. Li, Z. Huang, S.M. Siniscalchi, T. Wang, and C.H. Lee, "A Reverberation-Time-Aware DNN Approach Leveraging Spatial Information for Microphone Array Dereverberation," *Eurasip Journal on Advances in Signal Processing*, 2017.
[198] Y. Wu and K. He, "Group Normalization," in *European Conference on Computer Vision*, 2018, pp. 3–19.
[199] X. Xiao, S. Zhao, X. Zhong, D.L. Jones, E.S. Chng, and H. Li, "A Learning-Based Approach to Direction of Arrival Estimation in Noisy and Reverberant Environments," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015, pp. 2814–2818.
[200] X. Xiao, S. Zhao, D.L. Jones, E.S. Chng, and H. Li, "On Time-Frequency Mask Estimation for MVDR Beamforming with Application in Robust Speech Recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 3246–3250.
[201] C. Xu, X. Xiao, S. Sun, W. Rao, E.S. Chng, and H. Li, "Weighted Spatial Covariance Matrix Estimation for MUSIC based TDOA Estimation of Speech Source," in *Proceedings of Interspeech*, 2017, pp. 1894–1898.
[202] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "A Regression Approach to Speech Enhancement Based on Deep Neural Networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, pp. 7–19, 2015.
[203] T. Yoshioka, N. Ito, M. Delcroix, A. Ogawa, K. Kinoshita, M. Fujimoto, C. Yu, W.J. Fabian, M. Espi, T. Higuchi, S. Araki, and T. Nakatani, "The NTT CHiME-3 System: Advances in Speech Enhancement and Recognition for Mobile Multi-Microphone Devices," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2015, pp. 436–443.
[204] T. Yoshioka, H. Erdogan, Z. Chen, and F. Alleva, "Multi-Microphone Neural Speech Separation for Far-Field Multi-Talker Speech Recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 5739–5743.
[205] T. Yoshioka, A. Sehr, M. Delcroix, K. Kinoshita, R. Maas, T. Nakatani, and W. Kellermann, "Making Machines Understand Us in Reverberant Rooms: Robustness against Reverberation for Automatic Speech Recognition," *IEEE Signal Processing Magazine*, vol. 29, pp. 114–126, 2012.
[206] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation Invariant Training of Deep Models for Speaker-Independent Multi-talker Speech Separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 241–245.
[207] D. Yu, M.L. Seltzer, J. Li, J.-T. Huang, and F. Seide, "Feature Learning in Deep Neural Networks - Studies on Speech Recognition Tasks," *Proceedings of ICLR*,

2013.
[208] D. Yu, K. Yao, H. Su, G. Li, and F. Seide, “KL-Divergence Regularized Deep Neural Network Adaptation for Improved Large Vocabulary Speech Recognition,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 7893–7897.
[209] D. Yu and L. Deng, *Automatic speech recognition: a deep learning approach*. Springer, 2014.
[210] C. Zhang, D. Florêncio, and Z. Zhang, “Why Does PHAT Work Well in Lownoise, Reverberative Environments?,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008, pp. 2565–2568.
[211] W. Zhang and B.D. Rao, “A Two Microphone-Based Approach for Source Localization of Multiple Speech Sources,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, pp. 1913–1928, 2010.
[212] X.-L. Zhang and D.L. Wang, “A Deep Ensemble Learning Method for Monaural Speech Separation,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pp. 967–977, 2016.
[213] X. Zhang, Z.-Q. Wang, and D.L. Wang, “A Speech Enhancement Algorithm by Iterating Single- and Multi-Microphone Processing and Its Application to Robust ASR,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 276–280.
[214] X. Zhang and D.L. Wang, “Deep Learning Based Binaural Speech Separation in Reverberant Environments,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, pp. 1075–1084, 2017.
[215] Y. Zhao, Z.-Q. Wang, and D.L. Wang, “Two-Stage Deep Learning for Noisy-Reverberant Speech Enhancement,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, pp. 53–62, 2019.
[216] W. Zheng, Y. Zou, and C. Ritz, “Spectral Mask Estimation using Deep Neural Networks for Inter-Sensor Data Ratio Model Based Robust DOA Estimation,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015.
[217] M. Zohourian, G. Enzner, and R. Martin, “Binaural Speaker Localization Integrated in An Adaptive Beamformer for Hearing Aids,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, pp. 515–528, 2017.