Speech Enhancement for Applications in Various Environments Involving Fluctuation of Noise Feature

著者	Kawase Tomoko
year	2018
その他のタイトル	雑音特性の変動を伴う多様な環境で実用可能な音声強調
学位授与大学	筑波大学 (University of Tsukuba)
学位授与年度	2017
報告番号	12102甲第8525号
URL	http://doi.org/10.15068/00152397

Speech Enhancement

for Applications in Various Environments

Involving Fluctuation of Noise Feature

March 2018

Tomoko KAWASE

Speech Enhancement

for Applications in Various Environments

Involving Fluctuation of Noise Feature

Graduate School of Systems and Information Engineering

University of Tsukuba

March 2018

Tomoko Kawase

Abstract

When picking up sound using microphones, noise contaminates the signals observed by microphones. Therefore, technologies to separate the mixture of various sounds into each sound are required. This dissertation particularly focuses on technologies to enhance speech from the mixture. Speech enhancement is required to achieve good remote speech communication or automatic speech recognition (ASR). It is applied to various kinds of devices, e.g., audio conferencing systems, vehicle-mounted microphones, headsets, communication robots, and so on.

Many methods have been studied for practical application of speech enhancement in various environments. To separate speech from various kinds of noise, it is necessary to extract temporal features, spatial features, and spectral features of sounds as cues for estimating each source. There are two common approaches: one is to obtain spatial cues using a set of microphone array, and another is to prepare training sound data and obtain spectral cues by machine learning. Temporal cues and spatial cues are obtained by physically modeling the acoustic environment including the sound sources. On the other hand, it is difficult to obtain spectral cues using the physical models, thus spectral cues are often obtained statistically from training data.

This study aims to integrate microphone array and machine learning based approaches, and to effectively utilize temporal, spatial, and spectral cues. By integrating microphone array and machine learning, speech coming from directions other than the look direction and non-speech noise would be reduced effectively. Recently, studies to integrate speech recognition and speech enhancement based on machine learning has attracted attention, and it is assumed that speech enhancement is also implemented in a high performance computer, and a large machine learning model consisting of an enormous number of parameters is used. However, considering not only ASR but also speech communication as applications, such a large speech enhancement system is often not practical, so methods for integrating microphone array and machine learning with a small machine learning model should also be studied.

As a conventional method using a microphone array, a configuration using a beamforming and a post-filtering is widely used, and there is a practically effective method of estimating a power spectrum density (PSD) of the target and noise for designing a post-filter. Machine learning is not used in this method and instead, parameters for adjusting the property of the post-filter according to the type of noise assumed in each environment are set. By setting the value of the parameters empirically for each frequency band, a post-filter which is effective to the type of noise is calculated. In this study, automatic parameter switching (APS) was proposed to automatically switch the values of the parameters. APS optimizes a function for switching in advance by using training data of noisy observation signals. The APS takes the output of the beamformer and its stationary component as input and outputs the values of the parameters.

In APS, the model expressing the relationship between observation signal and target signal is improved by the switching function. As a second proposed method, the model representing the relationship is further sophisticated. Gaussian mixture model (GMM), which is a machine learning model often used for modeling speech, was integrated into the configuration of beamforming and post-filtering. In this method, GMM of clean speech is learned in advance. PSD of target speech is estimated using GMM, and PSD of noise is estimated based on spatial information as in the conventional method. Only clean speech data is used for learning GMM, there is no need to prepare data for various environments, and GMM does not become large scale like acoustic model of speech recognition. By using GMM, it is possible to maintain spectral features of speech, and as a result, speech enhancement performance can be improved.

Additionally, a method to introduce small-scale neural networks (NNs) into speech enhancement was also proposed. Following the conventional beamforming and post-filtering configurations, NNs are applied to estimate the PSD of the target speech and noise from the output of the beamformers.

This study showed that machine learning is integrated into microphone array speech enhancement in feasible way by expanding the composition of beamforming and post-filtering, without requiring large amount of computation. All of the three proposed methods are experimentally evaluated and it was shown that the enhancement performance was improved by using spectral cues as well as spatial cues.

Abstract in Japanese

本研究は，様々な音の混合音を個々の音に分離する音源分離技術に関するものである．特に，混合音から音声を強調する技術に焦点を当てる．マイクロホンによって観測された音声信号には，一般に雑音が混入する．したがって，良好な遠隔音声通信または自動音声認識を達成するために，音声強調が必要となる．音声強調は，音声会議システム，車載マイクロホン，ヘッドセット，通信ロボットなど，様々な種類の装置に適用される．

音声強調を多様な場面で実用化するために，多くの方法が研究されてきた．音声を様々な種類の雑音から分離するためには，音の時間情報，空間情報およびスペクトル情報を手がかりとして取り出す必要である．主なアプローチとして，マイクロホンアレイを使用して空間情報を得るアプローチと，学習データを用意し，機械学習によりスペクトル情報を得るアプローチがある．時間情報や空間情報は，音源を含む音響環境を物理的にモデル化することによって得られる．一方，物理モデルを用いてスペクトル情報を得ることは難しく，学習データから統計的に情報を得ることが多い．

本研究では，マイクロホンアレイと機械学習を統合し，時間情報や空間情報，スペクトル情報をより効果的に活用した音声強調を目指す．マイクロホンアレイと機械学習に基づくアプローチとを統合することにより，雑音のスペクトル特徴が音声のスペクトル特徴と異なっていれば，目的音と同じ方向から到来する雑音や目的方向以外の方向から到来する音声も効果的に低減できる．最近では，音声認識と機械学習に基づく音声強調を統合する研究が注目されており，音声強調も高性能な計算機に実装する想定で，膨大な数のパラメータからなる大規模な機械学習モデルが用いられている．しかし音声認識だけでなく音声通話も考慮する場合，そのような大規模な音声強調系が実用に沿わないことも多いため，小規模な機械学習モデルを使った機械学習とマイクロホンアレイの統合方法も研究されるべきである．

マイクロホンアレイを用いた従来手法として，ビームフォーマとポストフィルタを用いた構成が広く用いられており，ポストフィルタ設計のために目的音お

よび雑音のパワースペクトル密度（PSD: Power Spectral Density）を推定する方法がある．この方法では機械学習は用いられていない．かわりに，各環境で想定される雑音の種類に応じてポストフィルタの特性を調整するパラメータが設定されている．このパラメータの値を周波数帯域ごとに経験的に設定することで，雑音の特性に合った高性能なポストフィルタが算出される．本研究では，このパラメータの値を自動的に切替える方法（APS: Automatic Parameter Switching）を提案した．APS では，雑音の混入した観測信号の学習データを使用して，事前に切替えのための関数を最適化する．音声強調の段階では，APS はビームフォーマの出力とその定常成分を入力とし，パラメータの値を出力する．

APS では，切替え機能によって観測信号と目的信号の関係をあらわすモデルを高度化している．次の提案手法として，写像を表す関数のさらなる高度化を試みた．二番目の提案手法では，音声のモデル化によく用いられる機械学習モデルである，混合ガウスモデル（GMM: Gaussian Mixture Model）を，ビームフォーミングとポストフィルタの構成に統合した．この方法では，事前にクリーン音声の GMM を学習し，目的音声の PSD は GMM を用いて推定し，雑音の PSD については従来と同じように空間情報に基づいて推定する．学習にはクリーン音声のデータのみを用いており，多様な環境ごとのデータを用意する必要はなく，GMM も音声認識の音響モデルのように大規模にはならない．GMM を用いることで音声らしいスペクトル特徴を保持でき，結果として音声強調性能を改善することができる．

さらに，小規模なニューラルネットワーク（NN: Neural Network）を音声強調に導入する方法も提案した．従来のビームフォーマとポストフィルタの構成を踏襲し，NN をビームフォーマの出力から目的音声と雑音の PSD を推定するために適用した．

本研究では，高性能な計算機でなくても実現可能な方法で，機械学習をマイクロホンアレイ音声強調に統合する方法を示した．3 つの提案手法を実験的に評価し，空間情報だけでなくスペクトル情報も使用することによる音声強調性能の向上効果を確認した．

Acknowledgements

I would like to express my sincere thanks to my advisor, Dr. Shoji Makino, professor of Life Science Center of Tsukuba Advanced Research Alliance, University of Tsukuba, for his guidance and withstanding the enduring task of examining the draft. I would also like to express my thanks to the members of thesis review committee, Dr. Kazuhiro Fukui, Dr. Keisuke Kameyama, Dr. Yoichi Haneda, and Dr. Takeshi Yamada for their valuable advices.

I am grateful to my supervisors at NTT Media Intelligence Laboratories, Mr. Hitoshi Ohmuro now at NTT TechnoCross Corporation and Dr. Noboru Harada, for guiding this research. All members in Acoustic Information Processing group, especially Dr. Kazunori Kobayashi and Dr. Kenta Niwa should also be thanked for many valuable discussions. I wish to acknowledge to the colleagues in other groups for their important comments and helping me to learn the basics of machine learning and speech recognition. Among them, those who should be especially mentioned are Dr. Masakiyo Fujimoto now at National Institute of Information and Communications Technology, Dr. Tomohiro Nakatani, Dr. Shoko Araki, Dr. Yoshinori Kamado now at NTT DOCOMO, Dr. Manabu Okamoto, Mr. Takaaki Fukutomi, and Dr. Taichi Asami.

I would like to express my grateful thanks to Dr. Yusuke Hioka, senior lecturer at Department of Mechanical Engineering, the University of Auckland. His helpful advices and encouragement helped me a lot to carry out this study.

Finally, I appreciate to my family, Takehiko, Yukako, Atsushi, and Kenta for their patients and supports.

Contents

List of Figures

List of Tables

List of Abbreviations

APS	**A**utomatic **P**arameter **S**witching
ASR	**A**utomatic **S**peech **R**ecognition
CNN	**C**onvolutional **N**eural **N**etwork
DCR	**D**egradation **C**ategory **S**cale
DNN	**D**eep **N**eural **N**etwork
DS	**D**elay and **S**um
DUET	**D**egenerate **U**nmixing **E**stimation **T**echnique
EM	**E**xpectation-**M**aximization
FS	**F**ilter and **S**um
GMM	**G**aussian **M**ixture **M**odel
GSC	**G**eneralized **S**idelobe **C**anceller
HMM	**H**idden **M**arkov **M**odel
ICA	**I**ndependent **C**omponent **A**nalysis
ICA	**I**nter**Q**uartile **R**ange
LCMV	**L**inear **C**onstraint **M**inimum **V**ariance
LPSD	**L**ogarithmic compressed **P**ower **S**pectral **D**ensity
MMSE	**M**inimum **M**ean **S**quare **E**rror
MOS	**M**ean **O**pinion **S**core
MVDR	**M**inimum **V**ariance **D**istortionless **R**esponse
NMF	**N**nonnegative **M**atrix **F**actorization
NN	**N**eural **N**etwork
PSD	**P**ower **S**pectral **D**ensity
PSD-BS-BR	**PSD**-estimation-in-**B**eam**S**pace and **B**ackground noise **R**eduction
PSD-GMM	integration of **PSD**-BS-BR and **GMM**
PSD-NN	**PSD** estimation using **NN**
ReLU	**Re**ctified **L**inear **U**nit
RMSE	**R**oot **M**ean **S**quare **E**rror
RNN	**R**ecurrent **N**eural **N**etwork
SD	**S**pectral **D**istortion
SDR	**S**ignal-to **D**istortion **R**atio
SIR	**S**ignal-to **I**nterference **R**atio
SNR	**S**ignal-to **N**oise **R**atio
VTS	**V**ector **T**aylor **S**eriesa
WER	**W**ord **E**rror **R**ate

List of Symbols

$\boldsymbol{b}_{\mathrm{NN}}$	bias of NN	-
c, $\boldsymbol{c}$	centroid	-
C	group	-
$\boldsymbol{d}$, D, $\boldsymbol{D}$	directivity gain	-
d_{O}	distance	-
G	time-frequency mask	-
$\boldsymbol{h}$, H, $\boldsymbol{H}$	transfer function	-
I	number of units of NN	-
I_{GMM}	number of Gaussian components	-
I_{utt}	number of utterances contained in dataset	-
J	number of NN layers	-
J_{APS}	number of pre-adjusted parameter sets for APS	-
J_{utt}	number of HMM states	-
$\mathcal{J}$	objective function	-
K	number of sound sources	-
L	number of beamformers	-
M	number of microphones	-
N_{it}	number of iterations	-
N_{wrd}	number of words	-
o, $\boldsymbol{o}$, O	observed signal	-
$\boldsymbol{R}$	spatial correlation matrix	-
R_{grp}	number of groups for APS	-
R_{NN}	number of NNs	-
$\boldsymbol{s}$, S_{O}	spatially coherent sound signal	-
t	time	s
T	number of time frames	-
$\boldsymbol{u}$	input of NN	-
$\boldsymbol{v}$, V	spatially incoherent noise signal	-
$\boldsymbol{w}$, W, $\boldsymbol{W}$	filter	-
W_{NN}, $\boldsymbol{w}_{\mathrm{NN}}$, $\boldsymbol{W}_{\mathrm{NN}}$	combination weight of NN	-
$\boldsymbol{x}$	output of NN	-
Y	output of beamformers	-
Z	output signal	-
α	forgetting coefficient	-

θ	angle	rad
λ	mixture weight	-
$\lambda_{\mathrm{Lagrange}}$	Lagrange multiplier	-
$\boldsymbol{\mu}$	mean	-
ν, $\boldsymbol{\nu}$	noise power	-
ξ, Ξ	parameter in PSD-BS-BR	-
σ^2, $\boldsymbol{\Sigma}$	variance	-
τ	time frame index	-
ϕ, $\boldsymbol{\phi}$	PSD	-
ω	frequency	Hz
Ω	number of frequency bins	-

Chapter 1

Introduction

1.1 Background

There are various sounds in living environment. Some kinds of sounds are meaningful or informative for us to communicate with each other or know our surroundings. To utilize such sounds in multiple areas, following research area on the basis of acoustic signal processing has been evolved; sound source separation, sound transmission, sound reproduction, acoustic system identification, and so on. This study is about sound source separation, that is, technology for separating the mixture of various sound source signals into each individual sound source signal.

Concretely speaking, this study focuses on speech as the target to be captured. Speech sounds are obviously important for communication. Needs for remote communication is growing in daily life, and speech communications is necessary to remote communications. Additionally, automatic speech recognition (ASR) is also becoming popular as human-machine interface.

The speech signals picked up by microphones are contaminated by distracting sounds. Thus, the speech signals must be enhanced to achieve good remote speech communication or ASR. Speech enhancement is applied to various kinds of devices, e.g., audio conferencing systems, vehicle-mounted microphones, headsets, communication robots, and so on.

Speech enhancement technology aims to clearly extract speech from signals observed by microphones, removing the distracting sounds including followings.

(1) Additive noise

Ambient noise is generally inevitable and exists anywhere at all time. Additionally, there often exists competing sound source, e.g., music reproduced by loudspeakers, speech by people other than the target, and so on.

(2) Reverberation

Reverberation is the result of multipath propagation and caused by an enclosure, such as walls. Although reverberation imparts useful information about the surrounding enclosure and can make instrumental sounds rich, it causes spectral distortion and considered as a destructive factor in speech capture.

(3) Acoustic echo

Acoustic echo occurs due to the coupling between the loudspeakers and the microphones. It occurs in bi-directional communication and makes conversation very difficult.

The technologies which deal with the reverberation and acoustic echo are dereverberation and echo cancelling, respectively. This study focusses on the additive noise reduction and hereinafter, reducing additive noise other than the target speech is narrowly defined as "speech enhancement." Note that technology to reduce the additive noise without restricting the target to speeches is referred to as "noise reduction."

Speech and additive noise have each distinctive feature. Various types of sounds can be separated by capturing the following types of features.

(1) Temporal feature [1]

Sound analysis time is usually fewer than a few tens of seconds. Therefore, sounds which do not change over a few tens of seconds, e.g., road noise or noise of air conditioners are categorized as stationary sound. In contrast, non-stationary sounds change over a few tens of seconds or fewer. Some examples of non-stationary sounds are sounds of doors, music, and speeches. In practice, no sounds are perfectly stationary [2]. Even the road noise or noise of air conditioners changes in the long term.

(2) Spatial feature [3]

Sounds directly coming from a point sound source are categorized as spatially coherent sound. In contrast, incoherent sounds are ones coming from numerous directions. Although there are few sounds which is completely spatially incoherent in practice, background noise such as a buzz is often considered as a spatially incoherent sound.

(3) Spectral feature [1]

Different types of sounds have different distribution of energy in the frequency domain. For example, the main energy of road noise and wind noise is concentrated in the low frequencies, typically below 200 and 500 Hz, respectively. Contrary, sounds such as a buzz, speech and music occupy a wider frequency range.

In stationary noise environment, simple classical methods [4, 5] are effective to enhance speech. However, the noise environment is various and may fluctuate in many practical situations and it makes speech enhancement difficult. For example, noise observed in cars may contain engine noise, background music, interfering speech, and so on. Although engine noise is relatively stationary, the shape of its spectrum fluctuates when running speed changes. Enhancing speech effectively with a uniform method in various environments remains a big challenge.

1.2 Literature review

Many speech enhancement methods have been extensively studied to widen the application areas of speech communication and ASR. To estimate the target speech signals from the observed signals, the acoustic environment including sound sources is somehow modeled and temporal, spatial, or spectral features of sounds are extracted from the observed signal as cues for the estimation. The simple classical methods [4, 5] suppose noise as stationary and separate the target and noise using temporal cues and very simple spectral cues. There are two main approaches to advance speech enhancement technology: using multiple microphones, i.e., microphone array to use spatial cues, and introducing machine learning technology to use sophisticated spectral cues. Spatial cues, as well as temporal cues, become available by physically modeling the acoustic environment, thus hereinafter they are collectively referred as physical cues. On the other hand, it is difficult to obtain sophisticated spectral cues by using physical model, thus machine learning are introduced and spectral cues are statistically obtained using training data of sounds.

1.2.1 Microphone-array speech enhancement

In this section, techniques for enhancing a target speech signals from multiple observation signals collected by a microphone array are described. In microphone array speech enhancement, the sound coming from the direction of the target sound source, referred to as look direction, is enhanced using the fact that the target sound source and the noise source are physically separated in the space. Since this technique uses spatial cue, even if the interference noise is speech, the target speech can be enhanced.

The problem in which the direction of the target sound source is unknown is referred to as blind problem. For the blind problem, independent component analysis (ICA) [6, 7] can be used. In ICA, signals are separated by using the independence. Speech enhancement using ICA cannot be applied to problem where the number of sound sources is larger than the number of microphones. Such a problem is called underdetemined problem.

When the direction of the target sound source is known, a signal processing technique for controlling the directivity, referred to as beamforming, can be used. The system that performs beamforming is called beamformer. There are many applications where the direction of the target sound source is known. It is also possible to estimate the target sound direction in advance by the sound source localization technique [8]. Therefore, beamforming can be said to be useful. The beamformer can be applied to the underdetermined problem. The most basic beamformer is the delay-and-sum (DS) beamformer [9]. In DS beamformer, the differences in observation signals between the channels are modeled only by delay. Then, the multiple observation signals are delayed by an appropriate amount and added together to obtain the target signal.

Beamformers including DS beamformer were first developed with antennas and sonars. Compared with the signals handled in these cases, the frequency band of the acoustic signal has a width of ten times or more [8]. Therefore, in order to apply the beamformer to the acoustic field, a technique for designing a beamformer which have uniform directivity characteristics over a wide band is required. Therefore, a design method in which microphones near the center are arranged densely for high frequency sounds and the other microphones are arranged coarsely in a wide range for low frequency sounds was proposed [10]. However, with this method, the number of microphones becomes very large, and the microphone array becomes very large. Therefore, filter-and-sum (FS) beamformer, which controls the relationship between frequency and directivity by applying a linear filter instead of delay processing to signals observed by each microphone, was proposed [11].

If there are errors in the arrangement and characteristics of the microphone elements or the sound propagation model, the performance of DS and FS beamformers deteriorates. In other words, DS and FS beamformers are not optimal in time-varying acoustic environments. Additionally, it is analytically shown that the gain does not become sufficiently small with respect to some directions other than the target direction, i.e., sidelobe is formed as well as the main beam, thus noise remains in the output. Therefore, adaptive beamformer, in which coefficients are updated according to observation signals, was proposed. Conversely, the DS and FS beamformers are classified as a fixed beamformer because the coefficients are fixed irrespective of observation signals. A typical adaptive beamformer is called linear constraint minimum variance (LCMV) beamformer [12]. LCMV beamformer is designed to pass sound from the target direction and minimize dispersion of the output of the beamformer. The LCMV beamformer is formulated as a linear constrained least square optimization problem and solved by using Lagrangian undetermined multiplier [13] or gradient method [14]. Among the LCMVs, minimum variance distortionless response (MVDR) beamfomer [9], which sets the constraint that the sound from the look direction passes through with a gain of 1, is widely known. When the noise is spatially completely white, the MVDR beamformer is equivalent to the DS beamformer.

The LCMV beamformer can be decomposed into two orthogonal components [15]. The first component is fixed coefficients representing a constraint, and the other component is an unconstrained adaptive coefficients. Fixed coefficients correspond to a fixed beamformer and output a signal in which noise is reduced. By adaptive coefficient, noise components are extracted from observation signals. The noise components are subtracted from the target reference to make the final output. Since this subtraction is interpreted as removing the influence of side lobe of the fixed beamformer, the configuration obtained by decomposing the LCMV beamformer is called generalized side lobe canceller (GSC). Since this configuration reduces computational complexity, it is used in the implementation of LCMV beamformer.

MVDR beamformer, which is a representative of LCMV beamformer, is not the optimal filter in the sense of minimum mean square error (MMSE). It is shown that an optimum filter in the sense of MMSE is composed of MVDR beamformer and single-channel Wiener post-filter [8]. It has been confirmed that the noise suppression performance improves by adding the post-filter, especially when incoherent noise or diffusive noise exists. It is necessary to estimate the power spectral densities (PSDs) of the target sound and noise to design the post-filter. There is a method using

TABLE 1.1: Examples of machine learning model

Generative model	Discriminative model
GMM	Logistic regression
HMM	Support vector machines
Probabilistic context-free grammar	Maximum entropy markov model
Naive Bayes	Conditional random fields
Averaged one-dependence estimators	NN
Latent Dirichlet allocation	
Restricted Boltzmann machine	
Generative adversarial networks	
NMF	

self spectral density and cross spectral density of the observation signal to design a post-filter [16], but this method is based on the assumption that the observed signals are completely uncorrelated. Then many studies were done to improve the post-filter [17–22]. Among these researches, PSD-estimation-in-beamspace method [22] uses the temporal and spatial features of signals as cues to PSD estimation and it is superior to the others in a sense that the post-filter reduce not only incoherent but also coherent signals. However, since the actual acoustic environment is diverse and complicated, speech enhancement performance is limited in the post-filter designed using only temporal and spatial characteristics.

1.2.2 Machine-learning based speech enhancement

To introduce machine-learning based speech enhancement, machine learning itself is firstly stated here. Machine learning is a technology to make information processing system capable of learning or prediction through building machine learning model using data [23]. The machine learning model is divided into generative model, in which data is assumed to result from a probabilistic distribution, and discriminative model, in which probabilistic distributions are not assumed. Typical examples of machine learning model are listed in Table.

Machine learning technology by using multi-layer neural networks (NNs) or deep NNs (DNNs) are especially called deep learning. NN is learned by back propagation [24] and can perform as a non-linear function and it. DNN succeeded in some technical field such as ASR [25], after a method to enable learning of enormous amounts of parameters was discovered [26]. Many studies are carried out to sophisticate network structure. For example, only a few of units are connected between layers in convolutional neural networks (CNN) [27, 28]. CNNs are utilized to learn matrices representing specific pattern of information. As another structure of networks, recurrent neural networks (RNNs) [29] have return path in hidden layers and represent time dependency of information by temporary memorize information.

Following introduces application of machine learning into single-channel speech enhancement. The acoustic signals are transformed into the time-frequency domain and treated as spectra. The speech signals are sparsely present in the time-frequency domain and the property is called "sparse property" [30]. Passing the signal only in the time-frequency bin where the target sound exists, based on the sparseness of the speech signal, is called time-frequency masking and is often used for speech enhancement. The time-frequency mask includes a binary mask [30], Wiener filter [5, 31, 32], and Ephraim-Malah filter [33, 34]. There is an approach to estimate the time-frequency mask itself, or the parameters necessary to calculate the time-frequency mask, applying machine learning to the estimation. In this section, we review single-channel speech enhancement based on machine learning. In the case of single-channel, spatial cue cannot be used and information hidden in spectrum, referred to as spectral cue is used.

In speech enhancement based on machine learning, the problem is following two points:

1. Which type of machine learning model to be used to represent the relationship between the spectrum of the observation signal and the time-frequency mask, or the parameters necessary for calculating the time-frequency mask

2. How to learn model parameters

Regarding the second problem, learning is roughly divided into the following three.

(1) Supervised learning

It is used when a set of data of observation signals and corresponding target speech data is available in advance.

(2) Semi supervised learning

It is used when only the target voice data is available in advance.

(3) Unsupervised learning

It is used when only observation signal data is available.

Since the spectrum of the observation signal can be regarded as a nonnegative combination of the spectra of speech and noise, there are methods of modeling it using nonnegative matrix factorization (NMF) [35, 36]. NMF is an algorithm for decomposing a matrix into two low-dimensional matrices [37] and it is also interpreted as a generative model [38, 39]. Of the two matrices, a matrix called a basis matrix represents a specific spectral pattern of each sound source, and a matrix called an activation matrix represents nonnegative coupling of each sound source. It is necessary to learn the basis matrix and the activation matrix, and all of the supervised [40], semi-supervised [40], and unsupervised [41–43] learning have been proposed.

There are also methods using Gaussian mixture model (GMM) to model the speech-specific spectrum pattern [44–46]. In the methods using GMM, training data of clean speech is used to learn GMM in advance. This GMM is called clean speech GMM. Because the methods have been developed as preprocessing of ASR, they are also called feature enhancement technique and the clean speech GMM is composed in region of input features of ASR. Using the clean speech GMM and noise estimation, the mapping function from observed signal to clean speech is derived, e.g., by vector Taylor series (VTS) method [44, 47]. In the case of semi-supervised learning, expectation-maximization (EM) algorithm [47] and Kalman filter [46] are used for noise estimation. There is also a supervised approach to model noise in advance using training data of noise [45].

In recent years, DNN has been gaining attention also in speech enhancement. It is thought that by using large-scale networks with many parameters, it is possible to express the relationship between the target signal and the observation signal elaborately. On the basis of the concept, an approach to model nonlinear mapping which expresses the relationship between observation signals and target speech using DNN and to obtain its parameters by supervised learning has been studied. The basic methods are using DNN to estimate the time-frequency mask [48, 49] or using the denoising auto encoder (DAE) [50] to estimate clean speech [51]. As extensions of these methods, many variations have been tried, such as those using log-Mel filter banks [52] and MFCC [53–55] as input features, those using CNN [56] and

RNN [57] and those sophisticating objective functions [58, 59]. In addition, an end-to-end configuration, which integrates speech enhancement and speech recognition, has attracted attention [60].

1.2.3 Integration of microphone array and machine-learning for speech enhancement

A few studies recently reported about microphone array machine-learning based speech enhancement. As mentioned above, on the one hand, single-channel machine-learning approach cannot sufficiently separate interfering speech. On the other hand, microphone-array speech enhancement not using spectral cues is capable of reduce spatially coherent noise coming from directions other than the look direction, such as an interfering speech and music, but it cannot sufficiently reduce non-stationary noise coming from the look direction. By integrating the microphone array and machine-learning approaches, both of spatial and spectral cues can be used for enhancement. As a result, speech coming from directions other than the look direction and non-speech noise would be reduced effectively.

In a method called degenerate unmixing estimation technique (DUET) [30, 61–64], the time difference and the level difference of the two-channel observation signals are estimated. Then the sound sources are specified by unsupervised clustering [30, 62, 63] or supervised classification [61] of the estimated time and level differences in each time-frequency bin, and separated by a time-frequency mask. The original DUET uses only spatial cues and does not use spectral cues, therefore those using cepstrum in addition to time difference and level difference were proposed [64].

Multichannel expansion of speech enhancement using NMF or DNN has also been studied. In multichannel NMF [65–67], both spatial information and spectrum information can be used, but there are problems that calculation cost is high and initial value dependence is high because there are many parameters to be determined by learning. As regards speech enhancement using DNN, a method for combining multichannel observation to make input features of DAE [68], and noise-aware training [69], which uses a noise estimation as a feature, were proposed. Also, a beamformer design by DNN and its integration with ASR have been studied [70, 71] and the integrated system is referred as to end-to-end speech recognition.

TABLE 1.2: Examples of speech enhancement in practical use

Product name/institution	Features of technology
Intelligent microphone [22] NIPPON TELEGRAPH AND TELEPHONE CORPORATION	Beamforming and post-filter using spatial cues For vehicle-mounted microphones and headsets
VoCon [72] Nuance Communications, Inc.	Sound localization, beamforming and post-filter For ASR in noisy environments
VoiceDo [73] NEC	Sound localization, beamforming and post-filter For ASR in noisy environments
Alexa [74] Amazon.com, Inc.	Beamforming and post-filter using spectral cues For ASR to command smart speakers
Google Assistant [75] Google	End-to-end speech recognition For ASR to command smart speakers

1.3 The purpose of the studies

Some speech enhancement technology has already been applied in practice. The examples are shown in Table 1.2. In many products, methods composed of beamforming and post-filtering are adopted.

A few latest products adopt microphone array and machine learning. For example, Google Home adopts end-to-end speech recognition using DNN. Google Home receives multichannel observed signals and it does not adopt the composition of beamforming and post-filtering. Because ASR system is implemented in remote servers, not in local processors, to implement the speech enhancement systems followed ASR in the servers is reasonable. However, such implementation is not versatile. As regards development of the system, implementing and training DNN for ASR require immense amount of time and effort. Thus, it is hard to change parts of the system according the usage environment, when once the configuration of whole system is fixed. Note that a speech coding technology usually needs to be implemented into local processors for this configuration.

In many practical cases, we cannot change the remote servers and need to implement speech enhancement system in a local processer. It is important for speech enhancement systems in local processors to work with low computational complexity and small amount of memory, especially when it is used for speech communication, not for ASR. However, most of machine-learning based methods use large-scale machine learning models. Therefore, it is impossibly difficult to introduce machine-learning based approach into speech enhancement in local processor.

There are few examples of practical applications in which machine learning is integrated to microphone array speech enhancement, not requiring high performance computer. This study focusses on the composition of beamforming and post-filtering. There are few studies which integrate microphone array and machine learning on the basis of the composition of beamforming and post-filtering. Although both microphone array and machine learning are used in VoiceDo as shown in Table 1.2, these are just simply cascaded and we may fail to take advantage of using both of spatial and spectral cues. The physical model for the composition of beamforming and post-filtering has been well investigated and verified theoretically and experimentally. Therefore, we could efficiently introduce machine learning models as a part of the composition to compensate for weakness of the physical model. By integrating physical model and machine learning model efficiently, the performance would be improved even if a small-scale machine learning model is used.

1.4 Overview of dissertation

The organization of this dissertation is outlined in Fig. 1.1. Outline about conventional technologies particularly related with this study is given in Chapter 2. In each of Chapter 3–5, a novel speech enhancement method is proposed.

Chapter 3 Automatic parameter switching (APS)

Hioka et al. has introduced a method to estimates PSDs of the target and noise, referred to as *PSD-estimation-in-beamspace,* in order to analytically design Wiener post-filter to be applied to the beamformer's output [22, 76]. While the original *PSD-estimation-in-beamspace* approximates the mapping from PSDs of the beamformers' output to PSDs of incoherent sound sources by a linear function, background noise reduction was provided later. The method using *PSD-estimation-in-beamspace* and the background noise reduction is abbreviated as PSD-BS-BR hereafter. In PSD-BS-BR, machine learning is not used to obtain spectral cues. Instead, parameters to adjust speech enhancement function are introduced to the PSD estimation and Wiener filter calculation, and the values of the parameters are tuned empirically for every frequency bands according to noise types supposed in each environment. In this

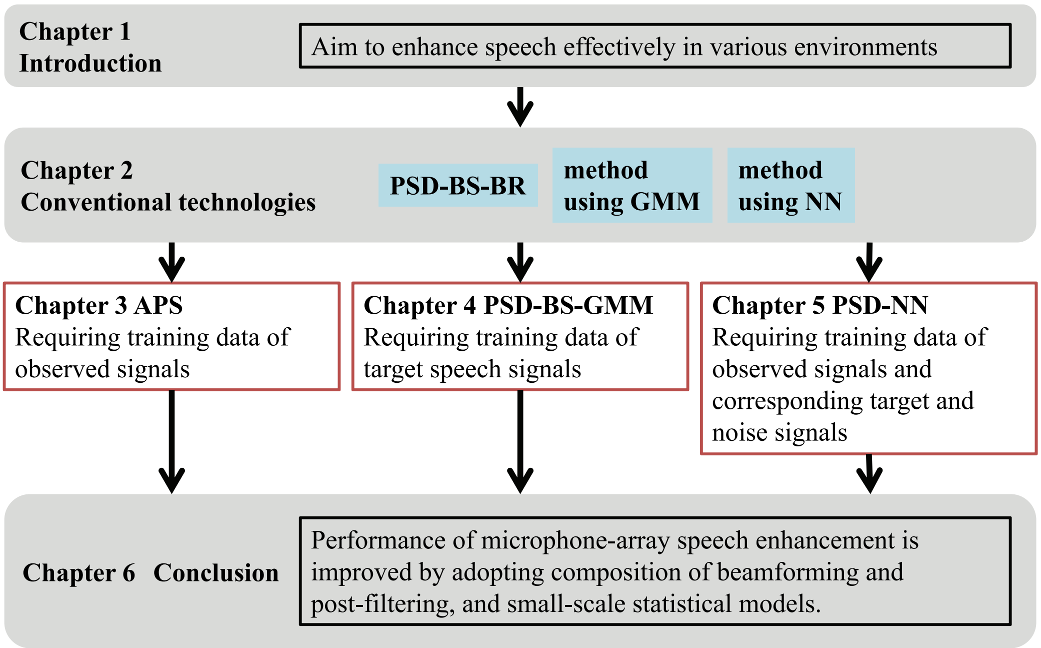


FIGURE 1.1: Overview of the dissertation

study, automatically switching parameters used for PSD estimation and Wiener filter calculation is proposed to improve performance in various environments. Automatic parameter switching (APS) receives the stationary components of the beamformer's output as input features, and the switching function itself is optimized in advance using training data of noisy speech.

Chapter 4 Integration of PSD-BS-BR and GMM (PSD-GMM)

APS approximates the mapping between the input and output by switching function, which is just an extension of PSD-BS-BR. Subsequently we expect that sophisticating the function which represents the mapping would further improve the performance. In this study, the target PSD is estimated using GMM, which is a machine learning model often used to model speech, trained in advance using clean speech data, and the noise PSD is estimated using PSD-BS-BR. Introducing GMM to represent target speech could preserve the features of speech and consequently improve the performance. It should be noted that target speech could be modeled using a reasonably small amount of training data because environmental noise little influences the target speech. By integrating microphone array approach and machine-learning approach in this way, the observed signals can be modeled using both of the physical and statistical cues effectively.

Chapter 5 PSD estimation using NN (PSD-NN)

NN is capable of estimating both of the target and noise from input feature in unified manner. While many studies adopt DNNs, we strive to introduce smaller-scale NNs into speech enhancement. The study on automatic parameter switching indicates that the beamformings' output and their stationary components are effective as input features; therefore the NNs are trained to map these features to source PSDs.

Finally, Chapter 6 concludes this dissertation.

Chapter 2

Fundamental Technologies of Speech Enhancement

2.1 Introduction

This chapter summarizes conventional technologies of speech enhancement. Section 2.2 discuss a way to model the sound sources and the relationship between the sound sources and observed signals. After explaining the problem setting, Section 2.3 explains one of the most basic speech enhancement methods. This method assumes noise as stationary. As approaches for separating non-stationary noise as well as stationary noise, microphone array speech enhancement and machine-learning based speech enhancement are introduced. Section 2.4 explains microphone array speech enhancement, which uses spatial cues. A method for designing a beamformer is described in Section 2.4.1 and PSD-estimation-in-beamspace is introduced as a method for deriving post-filter in Section 2.4.2. Additionally, two methods categorized into machine-learning based speech enhancement is introduced in Section 2.5.

2.2 Modeling of speech signal and noise

2.2.1 Time-frequency analysis

To save or transmit sounds, they are observed using microphones and usually represented by samples. By using Ω_{DFT}-point discrete Fourier transform to the observed signal $o(t)$, Fourier spectrum of the observed signal $O(\omega)$ is exploited, where t and ω denote the time index and frequency bins, respectively. The absolute value of

Fourier spectrum is referred to as amplitude spectrum and the square of the amplitude spectrum is referred to as power spectrum. The amplitude spectrum $|O(\omega)|$ and the power spectrum $|O(\omega)|^2$ are calculated as Eqs. (2.1) and (2.2), respectively.

$$|O(\omega)| = \sqrt{\{\mathfrak{Re}(O(\omega))\}^2 + \{\mathfrak{Im}(O(\omega))\}^2} \tag{2.1}$$

$$|O(\omega)|^2 = O(\omega)\bar{O}(\omega) \tag{2.2}$$

The distribution of the power of signals with respect to frequency is defined as PSD [77]. PSD $\phi_O(\omega)$ is of observed signal $O(\omega)$ is calculated by Eq. (2.3).

$$\phi_O(\omega) = E\left[|O(\omega)|^2\right] \tag{2.3}$$

It is hard to process long signal because properties of sound signals often varies over time. Thus, signals in short time are clipped and the temporal interval is referred as to frame. Defining τ as the frame index, the Fourier spectrum of $o(t)$ at frame τ is described as $O(\omega, \tau)$.

To calculate PSD for framed signals, Eq. (2.3) is approximated by Eq. (2.4), where T denotes the number of frames for average.

$$\phi_O(\omega) \approx \frac{1}{T}\sum_{\tau=0}^{T-1}|O(\omega,\tau)|^2 \tag{2.4}$$

If the signals are assumed to be non-stationary, Eq. (2.4) is further approximated by Eq. (2.5), where α_{PSD} is a forgetting coefficient.

$$\phi_O(\omega,\tau) \approx (1-\alpha_{\mathrm{PSD}})\sum_{\tau_d=0}^{\tau}\alpha_{\mathrm{PSD}}^{\tau_d}|O(\omega,\tau-\tau_d)|^2 \tag{2.5}$$

An analysis filterbank [77], which is described as Eq. (2.6), and a logarithmic arithmetic is applied to obtain the logarithmic compressed PSD (LPSD), as Eqs. (2.7) and (2.8), especially for analyzing speech, where the Ω_{SB} denotes the number of sub-bands.

$$\boldsymbol{W}_{\mathrm{AFB}} = \begin{bmatrix} \boldsymbol{w}_{\mathrm{AFB},1} \\ \boldsymbol{w}_{\mathrm{AFB},2} \\ \vdots \\ \boldsymbol{w}_{\mathrm{AFB},\Omega_{\mathrm{SB}}} \end{bmatrix} \tag{2.6}$$

$$\boldsymbol{\phi}^{\ln}(\tau) = \begin{bmatrix} \phi_1^{\ln}(\tau) \\ \phi_2^{\ln}(\tau) \\ \vdots \\ \phi_{\Omega_{\mathrm{SB}}}^{\ln}(\tau) \end{bmatrix} \tag{2.7}$$

$$\phi_{\omega_{\mathrm{SB}}}^{\ln}(\tau) = \ln\left(\boldsymbol{w}_{\mathrm{AFB},\omega_{\mathrm{SB}}} \begin{bmatrix} \phi(\omega,\tau)\,|_{\omega=0} \\ \phi(\omega,\tau)\,|_{\omega=1} \\ \vdots \\ \phi(\omega,\tau)\,|_{\omega=\Omega_{\mathrm{DFT}}-1} \end{bmatrix} \right) \tag{2.8}$$

The analysis filter bank outputs a sub-band signal. When it is necessary to restore the original signal from the sub-band signal, a synthesis filter bank described by Eq. (2.9) is applied.

$$\boldsymbol{W}_{\mathrm{SFB}} = \begin{bmatrix} \boldsymbol{w}_{\mathrm{SFB},1} \\ \boldsymbol{w}_{\mathrm{SFB},2} \\ \vdots \\ \boldsymbol{w}_{\mathrm{SFB},\Omega_{\mathrm{DFT}}} \end{bmatrix} \tag{2.9}$$

2.2.2 Sound propagation model

Assume that there are a target source, $K-1$ coherent interferences arriving from different angles, and incoherent background noise. The vectors $\boldsymbol{s}(\omega,\tau)$ denote the coherent sound sources including the target, i.e. $k=1$, defined in Eq. (2.10).

$$\boldsymbol{s}(\omega,\tau) = \begin{bmatrix} S_1(\omega,\tau) \\ S_2(\omega,\tau) \\ \vdots \\ S_K(\omega,\tau) \end{bmatrix} \tag{2.10}$$

They mix in the air and a mixture of all the signals is picked up by microphones as observed signals. Single-channel observed signal, i.e., signal observed by a single microphone, is expressed in frequency domain by Eq. (2.11), where the target signal arrives from a known direction θ_1.

$$\begin{aligned} O(\omega, \tau) &= \begin{bmatrix} H_1(\omega) & H_2(\omega) & \dots & H_K(\omega) \end{bmatrix} \boldsymbol{s}(\omega, \tau) + V(\omega, \tau) \\ &= O_\mathrm{S}(\omega, \tau) + O_\mathrm{N}(\omega, \tau) \end{aligned} \tag{2.11}$$

$$O_\mathrm{S}(\omega, \tau) = H_1(\omega) S_1(\omega, \tau) \tag{2.12}$$

$$O_\mathrm{N}(\omega, \tau) = \begin{bmatrix} H_2(\omega) & H_3(\omega) & \dots & H_K(\omega) \end{bmatrix} \begin{bmatrix} S_2(\omega, \tau) \\ S_3(\omega, \tau) \\ \vdots \\ S_K(\omega, \tau) \end{bmatrix} + V(\omega, \tau) \tag{2.13}$$

The V and H_k denote incoherent background noise and transfer function between the k-th sound source and the microphone, respectively. The subscript $_\mathrm{S}$ and $_\mathrm{N}$ denotes the target signal and the noise.

The power spectrum of the observed signal is derived as Eq. (2.14).

$$\begin{aligned} |O(\omega, \tau)|^2 =& |O_\mathrm{S}(\omega, \tau)|^2 + |O_\mathrm{N}(\omega, \tau)|^2 \\ &+ 2|O_\mathrm{S}(\omega, \tau)||O_\mathrm{N}(\omega, \tau)| \cos(\arg O_\mathrm{S}(\omega, \tau) - \arg O_\mathrm{N}(\omega, \tau)) \end{aligned} \tag{2.14}$$

Hereafter, all coherent sound sources and incoherent background noise are assumed to be mutually uncorrelated. Thus, the value of the third term in Eq. (2.14) is zero.

The microphone array observation $\boldsymbol{o}(\omega, \tau)$ is represented by a vector form, where the microphone array is composed of M microphones.

$$\begin{aligned} \boldsymbol{o}(\omega, \tau) &= \begin{bmatrix} O_1(\omega, \tau) \\ O_2(\omega, \tau) \\ \vdots \\ O_M(\omega, \tau) \end{bmatrix} \\ &= \boldsymbol{H}(\omega) \boldsymbol{s}(\omega, \tau) + \boldsymbol{v}(\omega, \tau) \end{aligned} \tag{2.15}$$

The $\boldsymbol{H}(\omega)$ consists of transfer functions between the k-th sound source and m-th microphone as described in Eqs. (2.16) and (2.17).

$$\boldsymbol{H}(\omega) = \begin{bmatrix} \boldsymbol{h}_1(\omega) & \boldsymbol{h}_2(\omega) & \dots & \boldsymbol{h}_K(\omega) \end{bmatrix} \tag{2.16}$$

$$\boldsymbol{h}_k(\omega) = \begin{bmatrix} H_{1,k}(\omega) \\ H_{2,k}(\omega) \\ \vdots \\ H_{M,k}(\omega) \end{bmatrix} \tag{2.17}$$

The vector $\boldsymbol{v}(\omega, \tau)$ denotes incoherent background noise, defined in Eq. (2.18).

$$\boldsymbol{v}(\omega, \tau) = \begin{bmatrix} V_1(\omega, \tau) \\ V_2(\omega, \tau) \\ \vdots \\ V_M(\omega, \tau) \end{bmatrix} \tag{2.18}$$

2.3 Speech enhancement using temporal cues

The enhanced speech signal is obtained by multiplying time-frequency mask $G(\omega, \tau)$ by the observed signal $O(\omega, \tau)$, as Eqs. (2.19) and (2.20), where Z is the ideal target source signal.

$$Z(\omega, \tau) = G(\omega, \tau)\, O(\omega, \tau) \tag{2.19}$$

$$G(\omega, \tau) \in [0, 1] \tag{2.20}$$

The $G(\omega, \tau)$ or parameters to design $G(\omega, \tau)$ should be derived. Wiener filter is the optimal filter in the MMSE sense and it is derived by minimizing the mean squared valued of the error described by Eq. (2.21).

$$E\left[|O_{\mathrm{S}}(\omega, \tau) - Z(\omega, \tau)|^2\right] = E\left[|O_{\mathrm{S}}(\omega, \tau) - G(\omega, \tau)\, O(\omega, \tau)|^2\right] \tag{2.21}$$

The error is differentiated with respect to the filter as Eq. (2.22).

$$\begin{aligned}
&\frac{\partial E\left[|O_{\mathrm{S}}(\omega,\tau)-Z(\omega,\tau)|^2\right]}{\partial G(\omega,\tau)} \\
&\quad = 2G(\omega,\tau)\,E\left[|O(\omega,\tau)|^2\right]-2E\left[O(\omega,\tau)\,\bar{O_{\mathrm{S}}}(\omega,\tau)\right] \\
&\quad \approx 2G(\omega,\tau)\,\phi_O(\omega,\tau)-2E\left[O(\omega,\tau)\,\bar{O_{\mathrm{S}}}(\omega,\tau)\right]
\end{aligned} \tag{2.22}$$

The $E\left[O(\omega,\tau)\,\bar{O_{\mathrm{S}}}(\omega,\tau)\right]$ is cross-spectrum of the observed signal and the target speech signal.

Eqs. (2.23) and (2.24) are derived by Eq. (2.22) to minimize the error.

$$2G(\omega,\tau)\,\phi_O(\omega,\tau)-2E\left[O(\omega,\tau)\,\bar{O_{\mathrm{S}}}(\omega,\tau)\right]=0 \tag{2.23}$$

$$G(\omega,\tau)=\frac{E\left[O(\omega,\tau)\,\bar{O_{\mathrm{S}}}(\omega,\tau)\right]}{\phi_O(\omega,\tau)} \tag{2.24}$$

The filter described by Eq. (2.24) is called Wiener filter.

If the $O_{\mathrm{S}}(\omega,\tau)$ and $O_{\mathrm{N}}(\omega,\tau)$ are uncorrelated each other, the PSD of $O(\omega,\tau)$ is expressed as Eq. (2.25).

$$\phi_O(\omega,\tau)=\phi_{O_{\mathrm{S}}}(\omega,\tau)+\phi_{O_{\mathrm{N}}}(\omega,\tau) \tag{2.25}$$

Additionally, the cross-spectrum of $O(\omega,\tau)$ and $O_{\mathrm{S}}(\omega,\tau)$ is described as Eq. (2.26).

$$\begin{aligned}
E\left[O(\omega,\tau)\,\bar{O_{\mathrm{S}}}(\omega,\tau)\right] &= E\left[(S(\omega,\tau)+N(\omega,\tau))\,\bar{O_{\mathrm{S}}}(\omega,\tau)\right] \\
&= E\left[|O_{\mathrm{S}}(\omega,\tau)|^2\right] \\
&\approx \phi_{O_{\mathrm{S}}}(\omega,\tau)
\end{aligned} \tag{2.26}$$

Wiener filter is described as Eq. (2.27) by substitute Eqs. (2.25) and (2.26) to Eq. (2.24).

$$G(\omega,\tau)=\frac{\phi_{O_{\mathrm{S}}}(\omega,\tau)}{\phi_{O_{\mathrm{S}}}(\omega,\tau)+\phi_{O_{\mathrm{N}}}(\omega,\tau)} \tag{2.27}$$

Wiener filter is designed using PSDs of speech and noise.

The $\phi_{O_{\mathrm{S}}}(\omega,\tau)$ and $\phi_{O_{\mathrm{N}}}(\omega,\tau)$ need to be derived to calculate Wiener filter. If noise is assumed to be stationary, the $\phi_{O_\mathrm{N}}(\omega,\tau)$ is often approximated by taking the average of $O(\omega)$ in the last some frames before the start of the speech. The $\phi_{O_\mathrm{S}}(\omega,\tau)$ is estimated using the principle of spectral subtraction method [4], as Eq. (2.28).

$$\phi_{O_\mathrm{S}}(\omega,\tau) = \phi_O(\omega,\tau) - \phi_{O_\mathrm{N}}(\omega,\tau) \tag{2.28}$$

2.4 Microphone-array speech enhancement

2.4.1 Beamforming

Beamforming is basic of array signal processing and applied to source localization and source separation. Beamformer is multichannel filter, described Eq. (2.29), and the output is described as Eq. (2.30).

$$\boldsymbol{w}(\omega) = \begin{bmatrix} W_1(\omega) \\ W_2(\omega) \\ \vdots \\ W_M(\omega) \end{bmatrix} \tag{2.29}$$

$$Y(\omega,\tau) = \boldsymbol{w}^{\mathrm{H}}(\omega)\,\boldsymbol{o}(\omega,\tau) \tag{2.30}$$

MVDR beamformer [9, 78] is trained using the observed signal $\boldsymbol{o}(\omega,\tau)$. MVDR beamformer is derived on the basis of constrained optimization. It minimizes the noise power without rejecting the target signals by minimizing variance of beamformer' s output and setting a constraint that the filter passes signals arriving from the target direction over all frequency bands. The constraint is described as Eq. (2.31).

$$\boldsymbol{w}^{\mathrm{H}}(\omega)\,\boldsymbol{h}_1(\omega) = 1 \tag{2.31}$$

The expectation of the power of the beamformers' output is described as Eqs. (2.32) and (2.33).

$$\begin{aligned} E\left[\left|\boldsymbol{w}^{\mathrm{H}}(\omega)\,\boldsymbol{o}(\omega,\tau)\right|^2\right] &= \boldsymbol{w}^{\mathrm{H}} E\left[\boldsymbol{o}(\omega,\tau)\,\boldsymbol{o}^{\mathrm{H}}(\omega,\tau)\right]\boldsymbol{w} \\ &= \boldsymbol{w}(\omega)^{\mathrm{H}}\,\boldsymbol{R}(\omega)\,\boldsymbol{w}(\omega) \qquad (2.32) \\ \boldsymbol{R}(\omega) &:= E\left[\boldsymbol{o}(\omega,\tau)\,\boldsymbol{o}^{\mathrm{H}}(\omega,\tau)\right] \qquad (2.33) \end{aligned}$$

Thus, the constrained optimization is represented as Eq. (2.34).

$$\min_{\boldsymbol{w}} \boldsymbol{w}(\omega)^{\mathrm{H}}\,\boldsymbol{R}(\omega)\,\boldsymbol{w}(\omega) \qquad \text{subject to} \qquad \hat{\boldsymbol{h}}_1^{\mathrm{H}}(\omega)\,\boldsymbol{w}(\omega) = 1 \qquad (2.34)$$

The $\hat{\boldsymbol{h}}_k(\omega)$ is array manifold vector [9], which models the transfer function $\boldsymbol{h}_k(\omega)$. Eq. (2.34) is solved using method of Lagrange multiplier. The objective function is set as Eq. (2.35).

$$\mathcal{J}_{\mathrm{MVDR}} = \boldsymbol{w}(\omega)^{\mathrm{H}}\,\boldsymbol{R}(\omega)\,\boldsymbol{w}(\omega) + 2\Re\mathfrak{e}\left(\lambda_{\mathrm{Lagrange}}\left(\hat{\boldsymbol{h}}_1^{\mathrm{H}}(\omega)\,\boldsymbol{w}(\omega) - 1\right)\right) \qquad (2.35)$$

The solution which minimizes the objective function is derived as Eq. (2.36).

$$\boldsymbol{w}(\omega) = \frac{\boldsymbol{R}^{-1}(\omega)\,\hat{\boldsymbol{h}}_1(\omega)}{\hat{\boldsymbol{h}}_1^{\mathrm{H}}(\omega)\,\boldsymbol{R}^{-1}(\omega)\,\hat{\boldsymbol{h}}_1(\omega)} \qquad (2.36)$$

2.4.2 Post-filter and PSD estimation

As mentioned in Chapter 1, applying Wiener filter to beamformer's output is optimal way in the sense of MMSE and the composition is widely used in practice. *PSD-estimation-in-beamspace* is a method for estimating PSDs of the target $\phi_S(\omega,\tau)$ and noise $\phi_N(\omega,\tau)$ in the beamformer's output for the following Wiener filter calculation. Fig. 2.1 summarizes the method explained in this section.

Let L $(\geq K)$ beamformers which focus their directivity on different angles be applied for microphone array observation. The output signal of the l-th beamformer is given by as Eq. (2.37).

$$Y_l(\omega,\tau) = \boldsymbol{w}_l^{\mathrm{H}}(\omega)\,\boldsymbol{o}(\omega,\tau) \qquad (2.37)$$

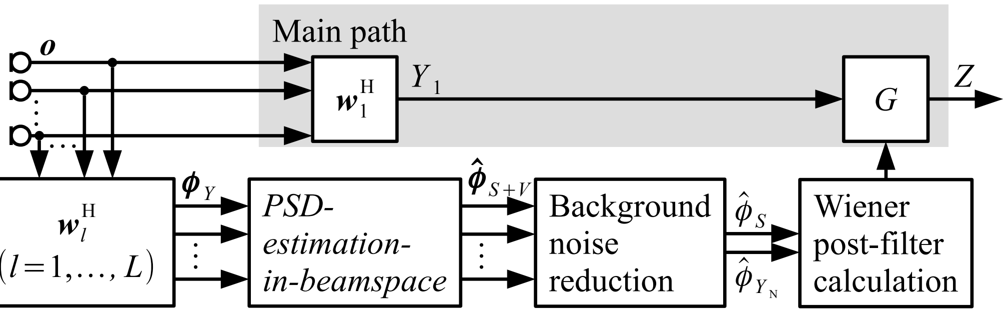


FIGURE 2.1: PSD-BS-BR

Hereafter, it is assumed that the directivity of the first beamformer ($l = 1$) points to the angle of the target source. Taking into account the sound propagation model described in Eq. (2.15), $Y_1(\omega, \tau)$ is represented also as Eqs. (2.38)–(2.40).

$$\begin{aligned} Y_1(\omega,\tau) &= \boldsymbol{w}_1^{\mathrm{H}}(\omega)\left(\boldsymbol{H}(\omega)\boldsymbol{s}(\omega,\tau)+\boldsymbol{v}(\omega,\tau)\right) \\ &= \boldsymbol{w}_1^{\mathrm{H}}(\omega)\boldsymbol{H}(\omega)\boldsymbol{s}(\omega,\tau)+\tilde{V}_1(\omega,\tau) \\ &= Y_{\mathrm{S}}(\omega,\tau)+Y_{\mathrm{N}}(\omega,\tau) \end{aligned} \tag{2.38}$$

$$Y_{\mathrm{S}}(\omega,\tau) = \boldsymbol{w}_1^{\mathrm{H}}(\omega)\boldsymbol{h}_1(\omega)S_1(\omega,\tau) \tag{2.39}$$

$$Y_{\mathrm{N}}(\omega,\tau) = \boldsymbol{w}_1^{\mathrm{H}}(\omega)\begin{bmatrix}\boldsymbol{h}_2 & \boldsymbol{h}_3 & \ldots & \boldsymbol{h}_K\end{bmatrix}\begin{bmatrix}S_2\\ S_3\\ \vdots\\ S_K\end{bmatrix}+\tilde{V}_1(\omega,\tau) \tag{2.40}$$

The $\tilde{V}_l(\omega,\tau)$ denotes the background noise component in l-th beamformer's output.

Although signal-to-noise ratio (SNR) is improved by applying beamforming, Y_{N} should be further reduced by applying post-filtering. On the basis of MMSE criterion, Wiener post-filter is basically calculated by Eq. (2.41).

$$G(\omega,\tau) = \frac{\hat{\phi}_{Y_{\mathrm{S}}}(\omega,\tau)}{\hat{\phi}_{Y_{\mathrm{S}}}(\omega,\tau)+\hat{\phi}_{Y_{\mathrm{N}}}(\omega,\tau)} \tag{2.41}$$

The PSD of $Y_1(\omega,\tau)$ is represented by Eq. (2.42), similarly to Eq. (2.25).

$$\phi_{Y_1}(\omega,\tau) = \phi_{Y_{\mathrm{S}}}(\omega,\tau)+\phi_{Y_{\mathrm{N}}}(\omega,\tau) \tag{2.42}$$

The $\phi_{Y_1}(\omega, \tau)$ can be approximated as Eq. (2.43) on the basis of the constraint described by Eq. (2.31), where $\phi_S(\omega, \tau)$ is PSD of the target signal $S_1(\omega, \tau)$, thus Eq. (2.41) is deformed to Eq. (2.44).

$$\phi_{Y_1}(\omega, \tau) \approx \phi_S(\omega, \tau) + \phi_{Y_N}(\omega, \tau) \tag{2.43}$$

$$G(\omega, \tau) = \frac{\hat{\phi}_S(\omega, \tau)}{\hat{\phi}_S(\omega, \tau) + \hat{\phi}_{Y_N}(\omega, \tau)} \tag{2.44}$$

The PSD of the l-th beamformer output $\phi_{Y_l}(\omega)$ can be approximated by an affine transformation of the PSDs of each source $\phi_{S_k}(\omega)$ with the directivity gain of the l-th beamformer to the k-th source direction $|D_{l,k}(\omega)|^2$ and noise PSD, as shown in Eqs. (2.45)–(2.47).

$$\begin{aligned} \boldsymbol{\phi}_Y(\omega, \tau) = &\begin{bmatrix} \phi_{Y_1}(\omega, \tau) \\ \phi_{Y_2}(\omega, \tau) \\ \vdots \\ \phi_{Y_L}(\omega, \tau) \end{bmatrix} \\ =&\boldsymbol{D}(\omega)\,\boldsymbol{\phi}_S(\omega, \tau) + \boldsymbol{\phi}_{\tilde{V}}(\omega, \tau) \end{aligned} \tag{2.45}$$

$$\boldsymbol{\phi}_S(\omega, \tau) = \begin{bmatrix} \phi_{S_1}(\omega, \tau) \\ \phi_{S_2}(\omega, \tau) \\ \vdots \\ \phi_{S_K}(\omega, \tau) \end{bmatrix} \tag{2.46}$$

$$\boldsymbol{\phi}_{\tilde{V}}(\omega, \tau) = \begin{bmatrix} \phi_{\tilde{V}_1}(\omega, \tau) \\ \phi_{\tilde{V}_2}(\omega, \tau) \\ \vdots \\ \phi_{\tilde{V}_L}(\omega, \tau) \end{bmatrix} \tag{2.47}$$

The $\boldsymbol{\phi}_{\tilde{V}}(\omega, \tau)$ describes PSD of incoherent background noise in the l-th beamformer's output. The directivity gain $\boldsymbol{D}(\omega)$ can be given in advance as described in Eqs. (2.48)–(2.49).

$$\boldsymbol{D}(\omega) = \begin{bmatrix} \boldsymbol{d}_1(\omega) & \boldsymbol{d}_2(\omega) & \dots & \boldsymbol{d}_K(\omega) \end{bmatrix} \tag{2.48}$$

$$\boldsymbol{d}_k(\omega) = \begin{bmatrix} |D_{1,k}(\omega)|^2 \\ |D_{2,k}(\omega)|^2 \\ \vdots \\ |D_{L,k}(\omega)|^2 \end{bmatrix} \tag{2.49}$$

$$D_{l,k}(\omega) = \boldsymbol{w}_l^{\mathrm{H}}(\omega)\,\hat{\boldsymbol{h}}_k(\omega) \tag{2.50}$$

The PSD of each coherent sound source can be separated by Eq. (2.51), where $^+$ denotes pseudo inverse.

$$\begin{aligned} \hat{\boldsymbol{\phi}}_{S+V}(\omega,\tau) &= \begin{bmatrix} \hat{\phi}_{S_1+V}(\omega,\tau) \\ \hat{\phi}_{S_2+V}(\omega,\tau) \\ \vdots \\ \hat{\phi}_{S_K+V}(\omega,\tau) \end{bmatrix} \\ &= \boldsymbol{D}^+(\omega)\,\boldsymbol{\phi}_Y(\omega,\tau) \\ &\approx \boldsymbol{\phi}_S(\omega,\tau) + \boldsymbol{D}^+(\omega)\,\boldsymbol{\phi}_{\tilde{V}}(\omega,\tau) \\ &\approx \boldsymbol{\phi}_S(\omega,\tau) + \boldsymbol{\phi}_V(\omega,\tau) \end{aligned} \tag{2.51}$$

However, as can be seen in the second term of (2.51), components originating from the spatially incoherent background noise are still included in the estimated PSD $\hat{\boldsymbol{\phi}}_{S+V}(\omega,\tau)$. Thus, *PSD-estimation-in-beamspace* is extended to estimate and remove the background noise by using the temporally stationary property of the background noise [79]. The composition of beamforming and post-filtering with the extended estimation method is referred to as method using *PSD-estimation-in-beamspace* and background noise reduction (PSD-BS-BR). Provided all spatially coherent sources including the target source are nonstationary, the PSD of background noise can be estimated by measuring the power of stationary components. The estimation of the stationary components $\hat{\phi}_{V_k}(\omega,\tau)$ in $\hat{\phi}_{S_k+V}(\omega,\tau)$ can be roughly obtained by calculating minimum statistics [80, 81] of $\hat{\phi}_{S_k+V}(\omega,\tau)$, as Eq. (2.52).

$$\begin{aligned}\hat{\phi}_{V_k}(\omega,\tau) &\approx f_{\mathrm{MS}}\left(\hat{\phi}_{S_k+V}(\omega,\tau)\right)\\ &= \min_{\tau\in T}\left\{\sum_{q=0}^{\tau-1}\beta(\omega)\,(1-\beta(\omega))^{q}\,\hat{\phi}_{S_k+V}(\omega,\tau)\right\}\end{aligned}\tag{2.52}$$

The T and $\beta(\omega)$ are a time interval and a forgetting factor, respectively. Similarly, $\hat{\phi}_{\tilde{V}_l}(\omega,\tau)$ can be obtained as Eq. (2.53).

$$\hat{\phi}_{\tilde{V}_l}(\omega,\tau) \approx f_{\mathrm{MS}}\left(\phi_{Y_l}(\omega,\tau)\right)\tag{2.53}$$

The PSD of target source $\phi_S(\omega,\tau)$ is calculated by Eq. (2.54).

$$\hat{\phi}_S(\omega,\tau) = \hat{\phi}_{S_1+V}(\omega,\tau) - \hat{\phi}_{V_1}(\omega,\tau)\tag{2.54}$$

Likewise, the PSD of noise can be calculated using the PSD of other coherent sources and background noise as Eq. (2.55), where $\xi_1(\omega)$ is a weighting parameter.

$$\begin{aligned}\hat{\phi}_{Y_N}(\omega,\tau) =& \underbrace{\xi_1(\omega)\sum_{k=2}^{K}\left\{\phi_{S_k+V}(\omega,\tau)-\hat{\phi}_{V_k}(\omega,\tau)\right\}}_{\text{PSD of interference sources}}\\ &+ \underbrace{\hat{\phi}_{\tilde{V}_1}(\omega,\tau)}_{\text{PSD of background noise}}\end{aligned}\tag{2.55}$$

The *PSD-estimation-in-beamspace* approximates the mapping from the beamformers' outputs to the source PSDs by a linear function, and error caused by the approximation sometimes causes musical noise. To reduce the musical noise, Wiener filter is often reshaped in PSD-BS-BS, as follows. Wiener filter is smoothed in the time domain as Eq. (2.56), where $\xi_2(\omega)$ is a forgetting coefficient.

$$G_{\mathrm{smooth}}(\omega,\tau) = \xi_2(\omega)\sum_{\tau_d=0}^{\tau}\{1-\xi_2(\omega)\}^{\tau_d}\,G(\omega,\tau-\tau_d)\tag{2.56}$$

Then it is floored in the time-frequency domain as Eq. (2.57), where $\xi_3(\omega)$ is a flooring parameter.

$$G_{\text{floor}} = \begin{cases} 1 & (G_{\text{smooth}}(\omega,\tau) \geq 1) \\ G_{\text{smooth}}(\omega,\tau) & (\xi_3(\omega) \leq G_{\text{smooth}}(\omega,\tau) < 1) \\ \xi_3(\omega) & (G_{\text{smooth}}(\omega,\tau) < \xi_3(\omega)) \end{cases} \tag{2.57}$$

Finally, the output signal $Z(\omega,\tau)$ is obtained by applying a Wiener filter $G_{\text{floor}}(\omega,\tau)$ to the output of the first beamformer $Y_1(\omega,\tau)$, which points its directivity to the target source, as follows.

$$Z(\omega,\tau) = G_{\text{floor}}(\omega,\tau)\,Y_1(\omega,\tau) \tag{2.58}$$

2.5 Machine-learning based speech enhancement

2.5.1 VTS method using GMM

This section explains a method to compose adaptive models, called VTS method. By using a clean speech model and a noise model, VTS method approximates the parameters of model which express observed signals, referred to as observation model. The relationship between the clean speech signals and the observed signals are derived in the LPSD domain in this section. Eq. (2.59) is derived by using Eqs. (2.8) and (2.25).

$$\exp\left(\phi^{\ln}_{O,\omega_{\text{SB}}}(\tau)\right) = \exp\left(\phi^{\ln}_{S,\omega_{\text{SB}}}(\tau)\right) + \exp\left(\phi^{\ln}_{N,\omega_{\text{SB}}}(\tau)\right) \tag{2.59}$$

Deforming Eq. (2.59), the LPSD of the observed signal is expressed using a non-linear function, as Eq. (2.60).

$$\begin{aligned} \phi^{\ln}_{O,\omega_{\text{SB}}}(\tau) &= \phi^{\ln}_{S,\omega_{\text{SB}}}(\tau) + \ln\left\{1 + \exp\left(\phi^{\ln}_{N,\omega_{\text{SB}}}(\tau) - \phi^{\ln}_{S,\omega_{\text{SB}}}(\tau)\right)\right\} \\ &= \phi^{\ln}_{S,\omega_{\text{SB}}}(\tau) + f_O\left(\phi^{\ln}_{S,\omega_{\text{SB}}}, \phi^{\ln}_{N,\omega_{\text{SB}}}\right) \end{aligned} \tag{2.60}$$

In VTS method, the relationship between $\phi^{\ln}_{O,\omega_{\text{SB}}}(\tau)$ and $\phi^{\ln}_{S,\omega_{\text{SB}}}(\tau)$ is expressed using Taylor expansion. Then, the probability density function of $\phi^{\ln}_{O,\omega_{\text{SB}}}(\tau)$ is estimated from that of $\phi^{\ln}_{S,\omega_{\mathrm{SB}}}(\tau)$.

The LPSDs of the clean speech and noise are assumed to follow Gaussian distributions, as Eqs. (2.61) and (2.62).

$$p\left(\boldsymbol{\phi}^{\ln}_{S}(\tau)\right) = \mathcal{N}\left(\boldsymbol{\phi}^{\ln}_{S}(\tau)\,;\boldsymbol{\mu}_S, \boldsymbol{\Sigma}_S\right) \tag{2.61}$$

$$p\left(\boldsymbol{\phi}^{\ln}_{N}(\tau)\right) = \mathcal{N}\left(\boldsymbol{\phi}^{\ln}_{N}(\tau)\,;\boldsymbol{\mu}_N, \boldsymbol{\Sigma}_N\right) \tag{2.62}$$

The $\mathcal{N}$, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ denote the probability density function of the Gaussian distribution, mean vector and variance matrix, respectively, which are calculated using training data. The LPSDs are supposed to be uncorrelated between filterbank channels, and the variance matrix is approximated by a diagonal matrix to reduce computational complexity, as Eqs. (2.63) and (2.64).

$$\boldsymbol{\Sigma}_S = \mathrm{diag}\left(\sigma^2_{S,1}, \sigma^2_{S,2}, \ldots, \sigma^2_{S,\Omega_{\mathrm{SB}}}\right) \tag{2.63}$$

$$\boldsymbol{\Sigma}_N = \mathrm{diag}\left(\sigma^2_{N,1}, \sigma^2_{N,2}, \ldots, \sigma^2_{N,\Omega_{\mathrm{SB}}}\right) \tag{2.64}$$

The Eq. (2.60) is approximated using a Taylor-series expansion around μ_S and μ_N as Eq. (2.65).

$$\begin{aligned}
f_O\left(\phi^{\ln}_{S,\omega_{\mathrm{SB}}}, \phi^{\ln}_{N,\omega_{\mathrm{SB}}}\right) &\approx \frac{1}{0!} f_O\left(\mu_{S,\omega_{\mathrm{SB}}}, \mu_{N,\omega_{\mathrm{SB}}}\right) \\
&+ \frac{1}{1!}\left\{\left(\phi^{\ln}_{S,\omega_{\mathrm{SB}}} - \mu_{S,\omega_{\mathrm{SB}}}\right)\frac{\partial}{\partial \phi^{\ln}_{S,\omega_{\mathrm{SB}}}} + \left(\phi^{\ln}_{N,\omega_{\mathrm{SB}}} - \mu_{N,\omega_{\mathrm{SB}}}\right)\frac{\partial}{\partial \phi^{\ln}_{N,\omega_{\mathrm{SB}}}}\right\} \\
&\quad f_O\left(\mu_{S,\omega_{\mathrm{SB}}}, \mu_{N,\omega_{\mathrm{SB}}}\right) \\
&+ \frac{1}{2!}\left\{\left(\phi^{\ln}_{S,\omega_{\mathrm{SB}}} - \mu_{S,\omega_{\mathrm{SB}}}\right)\frac{\partial}{\partial \phi^{\ln}_{S,\omega_{\mathrm{SB}}}} + \left(\phi^{\ln}_{N,\omega_{\mathrm{SB}}} - \mu_{N,\omega_{\mathrm{SB}}}\right)\frac{\partial}{\partial \phi^{\ln}_{N,\omega_{\mathrm{SB}}}}\right\}^2 \\
&\quad f_O\left(\mu_{S,\omega_{\mathrm{SB}}}, \mu_{N,\omega_{\mathrm{SB}}}\right) \\
&+ \cdots
\end{aligned} \tag{2.65}$$

The mean of $\phi^{\ln}_{O,\omega_{\mathrm{SB}}}(\tau)$ is obtained from Eq. (2.65) as Eq. (2.66).

$$\mu_{O,\omega_{\mathrm{SB}}} \approx \mu_{S,\omega_{\mathrm{SB}}} + f_O\left(\mu_{S,\omega_{\mathrm{SB}}}, \mu_{N,\omega_{\mathrm{SB}}}\right) \tag{2.66}$$

The variance matrix is obtained as Eqs. (2.67) and (2.68) by truncate Eq to zeroth and first order, respectively.

$$\sigma^2_{O,\omega_{\mathrm{SB}}} \approx \sigma^2_{S,\omega_{\mathrm{SB}}} \tag{2.67}$$

$$\begin{aligned}\sigma^2_{O,\omega_{\mathrm{SB}}} \approx & \left\{1 + \frac{\partial}{\partial \phi^{\ln}_{S,\omega_{\mathrm{SB}}}} f_O\left(\mu_{S,\omega_{\mathrm{SB}}}, \mu_{N,\omega_{\mathrm{SB}}}\right)\right\}^2 \sigma^2_{S,\omega_{\mathrm{SB}}} \\ & + \left\{\frac{\partial}{\partial \phi^{\ln}_{N,\omega_{\mathrm{SB}}}} f_O\left(\mu_{S,\omega_{\mathrm{SB}}}, \mu_{N,\omega_{\mathrm{SB}}}\right)\right\}^2 \sigma^2_{N,\omega_{\mathrm{SB}}}\end{aligned} \tag{2.68}$$

Note that $\phi^{\ln}_{O,\omega_{\mathrm{SB}}}(\tau)$ can be approximated accurately even if the Taylor expansion is truncated to low order when the LPSD of the clean speech and noise are within a relatively narrow region around the mean. The following explains the case of zeroth order.

The LPSD is often modeled using GMM as Eq. (2.69), instead of single Gaussian distribution.

$$p\left(\boldsymbol{\phi}^{\ln}_S(\tau)\right) = \sum_{i=1}^{I_{\mathrm{GMM}}} \lambda_{S,i} \mathcal{N}\left(\boldsymbol{\phi}^{\ln}_S(\tau); \boldsymbol{\mu}_{S,i}, \boldsymbol{\Sigma}_{S,i}\right) \tag{2.69}$$

The $lambda$ denotes the mixture weight. The parameters of the GMM are learned using EM algorithm. Because the model of $\phi^{\ln}_{O,\omega_{\mathrm{SB}}}(\tau)$ is composed of that of $\phi^{\ln}_{S,\omega_{\mathrm{SB}}}(\tau)$, the probability density function of $\phi^{\ln}_{O,\omega_{\mathrm{SB}}}(\tau)$ is also expressed using GMM. When GMM is used to model $\phi^{\ln}_{O,\omega_{\mathrm{SB}}}(\tau)$, the parameters are obtained as Eqs (2.70)–(2.72).

$$\lambda_{O,i} = \lambda_{S,i} \tag{2.70}$$

$$\mu_{O,i,\omega_{\mathrm{SB}}} \approx \mu_{S,i,\omega_{\mathrm{SB}}} + f_O\left(\mu_{S,i,\omega_{\mathrm{SB}}}, \mu_{N,\omega_{\mathrm{SB}}}\right) \tag{2.71}$$

$$\sigma^2_{O,i,\omega_{\mathrm{SB}}} \approx \sigma^2_{S,i,\omega_{\mathrm{SB}}} \tag{2.72}$$

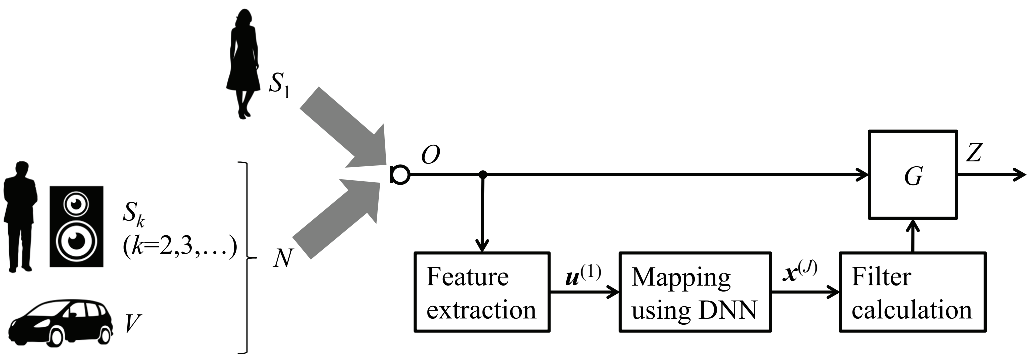


FIGURE 2.2: Overview of speech enhancement using DNN

The clean speech signal is estimated using the probability density function of $\phi^{\ln}_{O,\omega_{\mathrm{SB}}}(\tau)$ as Eq. (2.73).

$$\begin{aligned}
\hat{\boldsymbol{\phi}}^{\ln}_{S}(\tau) &= E\left(\boldsymbol{\phi}^{\ln}_{S}(\tau) \mid \boldsymbol{\phi}^{\ln}_{O}(\tau)\right) \\
&= \int \boldsymbol{\phi}^{\ln}_{S}(\tau)\, p\left(\boldsymbol{\phi}^{\ln}_{S}(\tau) \mid \boldsymbol{\phi}^{\ln}_{O}(\tau)\right) d\boldsymbol{\phi}^{\ln}_{S} \\
&= \int \left\{\boldsymbol{\phi}^{\ln}_{Z}(\tau) - f_O\left(\phi^{\ln}_{S,\omega_{\mathrm{SB}}}, \phi^{\ln}_{N,\omega_{\mathrm{SB}}}\right)\right\} p\left(\boldsymbol{\phi}^{\ln}_{S}(\tau) \mid \boldsymbol{\phi}^{\ln}_{O}(\tau)\right) d\boldsymbol{\phi}^{\ln}_{S} \\
&= \boldsymbol{\phi}^{\ln}_{Z}(\tau) - \sum_{i=1}^{I_{\mathrm{GMM}}} p\left(i \mid \boldsymbol{\phi}^{\ln}_{O}(\tau)\right) f_O\left(\mu_{S,i,\omega_{\mathrm{SB}}}, \mu_{N,\omega_{\mathrm{SB}}}\right)
\end{aligned} \tag{2.73}$$

The posterior probability in Eq.(2.73) is calculated using Bayes' theorem as Eq. (2.74).

$$p\left(i \mid \boldsymbol{\phi}^{\ln}_{O}(\tau)\right) = \frac{\lambda_i \mathcal{N}\left(\boldsymbol{\phi}^{\ln}_{O}(\tau); \boldsymbol{\mu}_{O,i}, \boldsymbol{\Sigma}_{O,i}\right)}{\sum_{i=1}^{I_{\mathrm{GMM}}} \lambda_i \mathcal{N}\left(\boldsymbol{\phi}^{\ln}_{O}(\tau); \boldsymbol{\mu}_{O,i}, \boldsymbol{\Sigma}_{O,i}\right)} \tag{2.74}$$

2.5.2 Method using DNN

In this section, speech enhancement using DNN is explained, whose overview is showed in Fig. 2.2. Some studies have applied deep learning as a way for extracting implicit spectral cues to speech enhancement. NNs are used as a nonlinear function to map parameters for designing time-frequency mask from the observed signal. Because the parameters for designing time-frequency mask are real variables such as amplitude spectrum or power spectrum, NNs are used as a regression function.

NN is composed of perceptron, or units. The units are layered and connected to propagate information of input and calculate output. NN with a lot of layers are called DNN.

Time-frequency mask design using DNN is basic method using deep learning. The observed signal $\{O_\tau \mid \tau = 1, \ldots, T\}$ and label data $\{\boldsymbol{x}_{\mathrm{d},\tau} \mid \tau = 1, \ldots, T\}$ is generated to train DNN. The observed signal in training data is generated using clean speech signals and noise signals, by Eq. (2.15).

Fig. 2.3 shows a schematic diagram of DNN. The output for τ-th sample $\boldsymbol{x}_\tau$ is calculated by feedforward DNN as Eqs. (2.75) and (2.76), where $\boldsymbol{u}$, $\boldsymbol{W}_{\mathrm{NN}}$, $\boldsymbol{b}_{\mathrm{NN}}$, φ, J, and I_j denote the input, combination weight, bias, and activation function, number of layers, and number of units in j-th layer, respectively.

$$\begin{aligned}\boldsymbol{u}_\tau^{(j)} &:= \begin{bmatrix} u_{\tau,1}^{(j)} \\ u_{\tau,2}^{(j)} \\ \vdots \\ u_{\tau,I_j}^{(j)} \end{bmatrix} \\ &= \boldsymbol{W}_{\mathrm{NN}}^{(j)} \boldsymbol{x}_\tau^{(j-1)} + \boldsymbol{b}_{\mathrm{NN}}^{(j)} \qquad (j = 2, \ldots, J)\end{aligned} \tag{2.75}$$

$$\begin{aligned}\boldsymbol{x}_\tau^{(j)} &= \begin{bmatrix} x_{\tau,1}^{(j)} \\ x_{\tau,2}^{(j)} \\ \vdots \\ x_{\tau,I_j}^{(j)} \end{bmatrix} \\ &= \begin{bmatrix} \varphi\left(u_{\tau,1}^{(j)}\right) \\ \varphi\left(u_{\tau,2}^{(j)}\right) \\ \vdots \\ \varphi\left(u_{\tau,I_j}^{(j)}\right) \end{bmatrix} \qquad (j = 2, \ldots, J)\end{aligned} \tag{2.76}$$

The activation function is often omitted when the NN is used to solve regression problems. The relationship between the input $\boldsymbol{u}^{(1)}$ and output $\boldsymbol{x}^{(J)}$ is expressed as Eq. (2.77), where the indices for the number of layers are omitted for simplicity.

$$\boldsymbol{x}^{(J)} = f\left(\boldsymbol{u}^{(1)}; \boldsymbol{W}_{\mathrm{NN}}, \boldsymbol{b}_{\mathrm{NN}}\right) \tag{2.77}$$

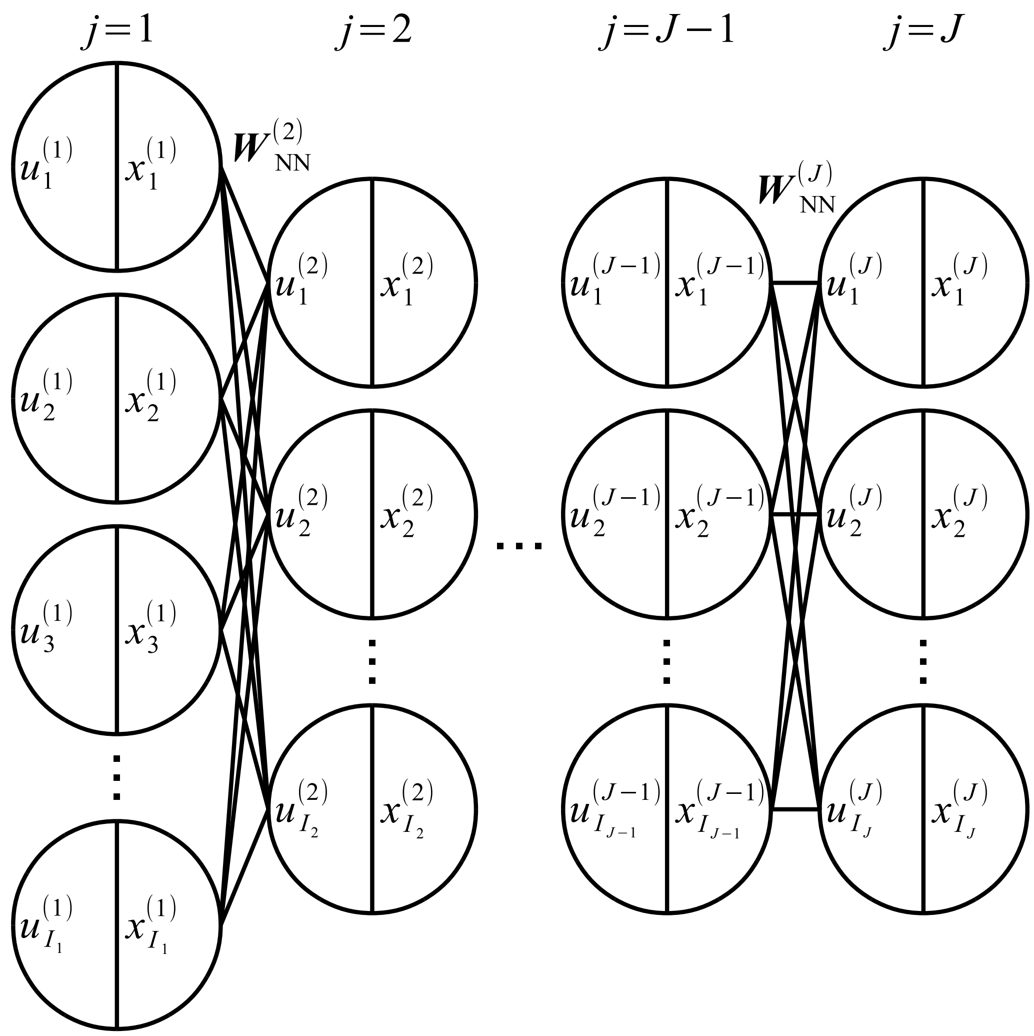


FIGURE 2.3: Schematic diagram of DNN

The $\boldsymbol{W}_{\mathrm{NN}}$ and $\boldsymbol{b}_{\mathrm{NN}}$ is trained by back-propagation, setting objective function as Eq. (2.78).

$$\mathcal{J} = -\frac{1}{2}\sum_{\tau=1}^{T} \|\boldsymbol{x}_{\mathrm{d},\tau} - f\,(\boldsymbol{u}_\tau; \boldsymbol{W}_{\mathrm{NN}}, \boldsymbol{b}_{\mathrm{NN}})\| \tag{2.78}$$

The input $\boldsymbol{u}_\tau$ is calculated from the generated observed signals O_τ. It is considered that DNN automatically extract acoustic features if it has a lot of layers and units and there is a large amount of training data. In this case, the observed signals are often directly used as the input variable $\boldsymbol{u}_\tau$. Mel-frequency cepstrum coefficient of the observed signals is also used as the input variable $\boldsymbol{u}_\tau$. Regarding the output, most of conventional methods design NN to output time-frequency mask directly.

Chapter 3

Automatic Parameter Switching (APS)

3.1 Introduction

As mentioned in Chapter 1, speech sounds need to be clearly captured for various kinds of applications, e.g., audio conferencing systems, vehicle-mounted microphones, headsets, communication robots, and so on. The speech sounds observed by microphones are affected by the acoustic environments in which such applications are used, which are often very noisy. Microphone-array speech enhancement is frequently applied in the practical applications, and is able to use the spatial cues of sound sources. Beamforming combined with post-filtering is a framework that is known to be practically effective; however, the PSDs of the target source and that of noise need to be estimated.

Methods for estimating the PSDs of sound sources separately by looking into the temporal [79] and spatial [22, 76] cues of each sound source have been proposed. With these methods, a set of parameters that can only be determined empirically are needed to calculate the PSDs and coefficients of the post-filter. Since previous studies discovered that speech enhancement performance using estimated PSDs is highly dependent on selected value of the post-filter parameter-set, an additional method for automatically selecting the best parameter-set value needed to be investigated [82, 83].

In this chapter, a method for automatically *switching* the post-filter parameter-set is proposed, providing the highest ASR accuracy. The method introduces the noise-power vector, which quantifies the features of noise contaminating each speech sentence by measuring the spectral power of different frequency bands. The noise-power vector is then used to group speech sentences to assign the best post-filter parameter-set. ASR systems will achieve the lowest word error rate (WER), following speech enhancement with the best post-filter parameter-set. The WER is modelled by a function of the centroids of groups and the post-filter parameter-sets. The lowest WER value is searched for by adjusting the position of the centroids and parameter-sets using the hill-climbing algorithm.

In Section 3.2, noise-power vector is introduced as a quantitative measurement of noise features. The concept of grouping noise-power vectors calculated for training data is introduced in Section 3.3, and optimizing of the group is explained in Section 3.4. We present the experimental results obtained using speech sentences in various noisy environments along with discussion in Section 3.5 and conclude this chapter with some remarks in Section 3.6.

3.2 Noise feature measurement

Because the features of a noisy environment vary depending on the scene where the ASR is used, one can hypothesize that the WER of ASR can be reduced by switching the values set for the parameters of speech enhancement depending on the noisy environment. The proposed parameter-switching method selects a parameter-set from J_{APS} pre-adjusted sets for each frequency that minimizes the WER. The J_{APS} pre-adjusted parameter-sets are described as Eq. (3.1), and the elements of the parameter-sets is described in Eqs. (2.55), (2.56) and (2.57).

$$\varXi_j = \left\{\xi_{1,j}(\omega), \xi_{2,j}(\omega), \xi_{3,j}(\omega)\right\} \qquad (j = 1, \ldots, J_{\mathrm{APS}}) \tag{3.1}$$

Given that a training dataset consisting of I_{utt} utterances with various types of noise being superimposed and their correct word labels are provided, assume the post-filter parameter-sets $\varXi_j$ are manually pre-adjusted using parts of the dataset. The ASR is applied to the denoised speech using every $\varXi_j$ $(j = 1, \ldots, J_{\mathrm{APS}})$, and the parameter-set that minimizes the WER is then selected as the optimal parameter-set.

To quantify the features of a noisy environment, a noise-power vector $\boldsymbol{\nu}_i$, composed of stationary noise power in different frequency bands is introduced as Eq. (3.3).

$$\boldsymbol{\nu}_i = \begin{bmatrix} \nu_{\mathrm{low},i} \\ \nu_{\mathrm{med},i} \\ \nu_{\mathrm{high},i} \end{bmatrix} \tag{3.2}$$

$$= \begin{bmatrix} \frac{\sum_{\tau_{\min} \le \tau < \tau_{\max}} \sum_{\omega_{\mathrm{low,min}} \le \omega < \omega_{\mathrm{low,max}}} \hat{\phi}_{\tilde{V}_1}(\omega,\tau)}{(\tau_{\max}-\tau_{\min})(\omega_{\mathrm{low,max}}-\omega_{\mathrm{low,min}})} \\ \frac{\sum_{\tau_{\min} \le \tau < \tau_{\max}} \sum_{\omega_{\mathrm{med,min}} \le \omega < \omega_{\mathrm{med,max}}} \hat{\phi}_{\tilde{V}_1}(\omega,\tau)}{(\tau_{\max}-\tau_{\min})(\omega_{\mathrm{med,max}}-\omega_{\mathrm{med,min}})} \\ \frac{\sum_{\tau_{\min} \le \tau < \tau_{\max}} \sum_{\omega_{\mathrm{high,min}} \le \omega < \omega_{\mathrm{high,max}}} \hat{\phi}_{\tilde{V}_1}(\omega,\tau)}{(\tau_{\max}-\tau_{\min})(\omega_{\mathrm{high,max}}-\omega_{\mathrm{high,min}})} \end{bmatrix} \quad (i = 1, \ldots, I_{\mathrm{utt}}) \tag{3.3}$$

The $[\tau_{\min}, \tau_{\max})$ is interval of frames to be averaged across. The $[\omega_{\mathrm{low,max}}, \omega_{\mathrm{low,max}})$, $[\omega_{\mathrm{med,max}}, \omega_{\mathrm{med,max}})$, and $[\omega_{\mathrm{high,max}}, \omega_{\mathrm{high,max}})$ are low, medium, and high frequency bands, respectively. The $\hat{\phi}_{\tilde{V}_1}(\omega, \tau)$ is obtained by calculating minimum statistics of $\phi_{Y_1}(\omega, \tau)$. Measuring the noise power in different frequency bands allows the use of the spectral distribution of the noise in addition to its loudness.

3.3 Parameter selection by grouping noise-power vectors

Fig. 3.1 shows a scatter plot of the noise-power vector calculated from I_{utt} noisy speech signals, i.e., $\boldsymbol{\nu}_i$ $(i = 1, \ldots, I_{\mathrm{utt}})$. Noise-power vectors are grouped into R_{grp} $(I_{\mathrm{utt}} \gg R_{\mathrm{grp}} \ge 2)$ groups; then, different parameter-set values are applied to the noisy speech signals that belong to each group in such a way that the overall WER averaged across the whole dataset is minimized. In other words, the i-th data is grouped into one of the R_{grp} groups r that is assigned the best parameter-set value that maximizes ASR accuracy according to its noise-power vector.

The grouping is made possible by using centroids defined in the space of the noise-power vector given by Eq. (3.4), where $c_{\mathrm{Low},r}$, $c_{\mathrm{Med},r}$, and $c_{\mathrm{High},r}$ are the coordinates

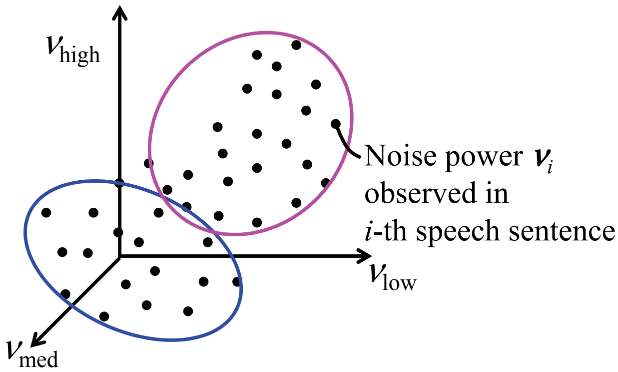


FIGURE 3.1: Group of noise-power vectors

of the r-th centroid.

$$\boldsymbol{c}_r = \begin{bmatrix} c_{\text{Low},r} \\ c_{\text{Med},r} \\ c_{\text{High},r} \end{bmatrix} \qquad \left(r = 1, \ldots, R_{\text{grp}}\right) \tag{3.4}$$

Initially, the centroids are set at the same positions as that of randomly selected R_{grp} noise-power vectors among all noise-power vectors measured from the I_{utt} speech sentences. Then, all noise-power vectors are grouped to their closest centroid by Eq. (3.5), where C_r denotes the set of indices of the utterances grouped into the r-th group.

$$C_r \ni \forall i \qquad \text{subject to} \qquad r = \arg\min_{r'} ||\mathbf{v}_i - \boldsymbol{c}_{r'}||^2 \tag{3.5}$$

3.4 Optimal grouping for maximizing speech recognition accuracy

The grouping now needs to be optimized in order to minimize the WER. On the basis of the hypothesis that the parameter-sets affect ASR accuracy, the WER of speech sentences in the C_r group will be represented by a function of $\boldsymbol{c}_r$ and $\varXi_j$, i.e., $f_{\text{WER}}\left(\boldsymbol{c}_r, \varXi_j\right)$. This implies that both $\boldsymbol{c}_r$ and $\varXi_j$ have to be adjusted to minimize the WER. The proposed method achieves this by tuning both $\boldsymbol{c}_r$ and $\varXi_j$ alternately as follows. Fig. 3.2 shows a flow chart of the optimization.

Once C_r is determined by the initial grouping given by Eq. (3.5), the best parameter-set value is selected for each group by Eq. (3.6), where j_r is the index of the chosen post-filter parameter-set for the r-th group.

$$j_r = \arg\min_j \left\{ f_{\mathrm{WER}} \left(\boldsymbol{c}_r, \Xi_j \right) \right\} \tag{3.6}$$

Then, the grouping is further optimized by adjusting the positions of the centroids provided the parameter-sets j_r are fixed. To this end, a cost function that quantifies the overall WER is introduced as Eq. (3.7).

$$\mathcal{J}_{\mathrm{WER}} = \frac{\sum_{r=1}^{R_{\mathrm{grp}}} f_{\mathrm{WER}} \left(\boldsymbol{c}_r, \Xi_{j_r} \right) N_{\mathrm{wrd},r}}{\sum_{r=1}^{R_{\mathrm{grp}}} N_{\mathrm{wrd},r}} \tag{3.7}$$

The $N_{\mathrm{wrd},r}$ denotes the number of words in the speech sentences that belong to the r-th group. Since $f_{\mathrm{WER}} \left(\boldsymbol{c}_r, \Xi_j \right)$ cannot be mathematically formulated, it is difficult to analytically derive $\boldsymbol{c}_r$ that minimizes $\mathcal{J}_{\mathrm{WER}}$. Instead, the hill-climbing algorithm [84] is applied to search the optimal $\boldsymbol{c}_r$. This method finds an optimal value by updating parameters while evaluating the change in the cost function. In this study, $\mathcal{J}_{\mathrm{WER}}$ in (3.7) was evaluated while $\boldsymbol{c}_r$ was perturbed by $\boldsymbol{\epsilon}_r$, given by

$$\boldsymbol{c}_r \leftarrow \boldsymbol{c}_r \pm \boldsymbol{\epsilon}_r, \tag{3.8}$$

$$\boldsymbol{\epsilon}_r = \begin{bmatrix} \epsilon_{\mathrm{Low},r} \\ \epsilon_{\mathrm{Med},r} \\ \epsilon_{\mathrm{High},r} \end{bmatrix} \tag{3.9}$$

After updating $\boldsymbol{c}_r$, the parameter-set values assigned to groups will be reviewed and updated by Eq. (3.6). This two-step process will carry on until no further reduction in $\mathcal{J}_{\mathrm{WER}}$ is observed by adjusting $\boldsymbol{c}_r$. Because $\mathcal{J}_{\mathrm{WER}}$ may not be minimized when the initial value of $\boldsymbol{c}_r$ is not arranged properly, the optimal search needs to be run multiple times with randomly selected initial centroids; then, Ξ_{j_r} and $\boldsymbol{c}_r$ that provide the lowest $\mathcal{J}_{\mathrm{WER}}$ after reaching its minimum value need to be selected.

3.5 Experiments

We compared ASR accuracy obtained using APS with that using the conventional PSD-BS-BR, in which a single post-filter parameter-set value was applied.

3.5.1 Setup

The microphone array used in the experiment consisted of three cardioid microphones. Each microphone was oriented 120° from the others. Beamformers were designed by using the MVDR method to point their directivity towards the three directions to which the microphones were facing.

Training and evaluation data were manually prepared as follows. We measured the impulse responses from the positions of the target and noise sources to the microphone array as shown in Fig. 5.3. Impulse responses from eight different directions were measured to simulate incoherent background noise. The measurements were carried out in a reverberant chamber of which two different amounts of sound-absorbing material was put up on walls and the ceiling and in a meeting room to confirm that the proposed method is effective in various reverberant environments. Clean speech signals were convolved with the recorded impulse responses of the target to make the target sound signals. In the same way, one of four different types of background noise recorded in cars, offices, shopping centers, or exhibition halls was convolved with the impulse responses of the noise from the eight directions, which were summed together. Finally, the target sound and background noise were added up with different SNRs, which were varied from -10 to 10 dB.

The Complete Continuous Speech Recognition-I (CSR-I) corpus [85] was used for the clean speech signals. The corpus was divided into two subsets, that is, training and evaluation datasets, which were composed of 323 and 328 English utterances spoken by four individuals, respectively. An acoustic model and language model were trained using the Kaldi [86] baseline tool for the CHiME Challenge [87]. The acoustic model was constructed with a hidden Markov model (HMM) with GMM. For the proposed method, 28 sets of parameters were manually prepared.

It is expected that the number of the groups R_{grp} would also affect the ASR accuracy apart from the position of centroids and selected parameter-set values.

Table 3.1: Experimental conditions for evaluation of APS

Sampling rate (kHz)	16
Quantization bit rate (bit)	16
# of microphones, M	3
# of beamformers, L	3
Target distance, d_S (m)	0.5, 1.0, 1.5, 2.0
Noise distance, d_N (m)	2.0, 4.0
Frame length (ms)	32
Frame shift (ms)	16
# of sentences for training, I_{utt}	323
# of sentences for evaluation	328
# of groups, R_{grp}	2, 3, 4, 5, 6
# of parameter-sets, J_{APS}	28
Perturbation amplitude, $\boldsymbol{\epsilon}$	$[1, 1, 1]^{\mathrm{T}}$

Table 3.2: Centroids obtained by training when $R_{\mathrm{grp}} = 4$

	ν_{Low} (dBov)	ν_{Med} (dBov)	ν_{High} (dBov)
$\boldsymbol{c}_1$	−66	−61	−72
$\boldsymbol{c}_2$	−56	−64	−98
$\boldsymbol{c}_3$	−55	−42	−42
$\boldsymbol{c}_4$	−27	−30	−61

Since the relationship between R_{grp} and ASR accuracy is not easily modeled, we experimentally investigated the relationship by varying R_{grp} from 2 to 6.

The other experimental conditions are listed in Table 3.1.

3.5.2 Results

In the experiments, the centroids were derived by using the training process according to the flow chart in Fig. 3.2. The centroids derived when $R_{\mathrm{grp}} = 4$ are listed in Table 3.2, where the noise power is described in dBov, which denotes the level relative to the maximum value that can be stored in an integer format on a computer [88]. The centroids specified the groups of noise power observed in the speech sentences. Fig. 3.4 shows the noise-power vectors, which were divided into four groups using the centroids, in different colors/markers.

A parameter-set was assigned for each of the groups. The component values of the parameter-sets for different R_{grp} are listed in Table 3.3. Note that we averaged

TABLE 3.3: Frequency-averaged post-filter parameter-sets

		ξ_1	$\xi_2 \times 10^{-3}$	ξ_3
$R_{\mathrm{grp}} = 1$	Ξ_{j_1}	0.8	9.3	0.20
$R_{\mathrm{grp}} = 2$	Ξ_{j_1}	0.8	9.3	0.20
	Ξ_{j_2}	2.0	9.3	0.20
$R_{\mathrm{grp}} = 3$	Ξ_{j_1}	0.8	9.3	0.20
	Ξ_{j_2}	2.0	11.6	0.20
	Ξ_{j_3}	0.8	9.9	0.20
$R_{\mathrm{grp}} = 4$	Ξ_{j_1}	0.8	9.3	0.20
	Ξ_{j_2}	2.0	9.9	0.20
	Ξ_{j_3}	2.0	9.9	0.16
	Ξ_{j_4}	0.8	9.9	0.20
$R_{\mathrm{grp}} = 5$	Ξ_{j_1}	0.8	9.3	0.20
	Ξ_{j_2}	2.0	11.6	0.20
	Ξ_{j_3}	0.8	9.3	0.20
	Ξ_{j_4}	1.0	0.0	1.00
	Ξ_{j_5}	1.0	0.0	1.00
$R_{\mathrm{grp}} = 6$	Ξ_{j_1}	0.8	9.3	0.20
	Ξ_{j_2}	2.0	11.6	0.20
	Ξ_{j_3}	0.8	9.3	0.20
	Ξ_{j_4}	1.0	0.0	1.00
	Ξ_{j_5}	2.0	9.9	0.20
	Ξ_{j_6}	1.0	0.0	1.00

the values with respect to the frequency bins for simplicity. The experimental results obtained when $R_{\mathrm{grp}} = 1$ are equivalent to those with the conventional method. The same parameter-set used for $R_{\mathrm{grp}} = 1$ was also used to evaluate the conventional method when $R_{\mathrm{grp}} \geq 2$. The ASR results obtained using the assigned parameter-sets are described by WER in Table 3.4, in which WER values reduced by using APS are highlighted in boldface type.

The results clearly show that the WER was reduced in most groups with the proposed method than with the conventional method ($R_{\mathrm{grp}} = 1$).

Regardless of the value of R_{grp}, there was a majority group, or a group that included a relatively large number of speech sentences. Although the WER was not reduced in the majority groups, it was reduced in other groups, or minority groups. With the conventional method, in which a single parameter-set was assigned to all the speech sentences, the parameter-set suitable for the majority group was selected

TABLE 3.4: ASR results with WER $f_{\mathrm{WER}}\left(\boldsymbol{c}_r, \Xi_{j_r}\right)$

		# of utterances	w/o APS	with APS
$R_{\mathrm{grp}} = 1$	C_1	328	41.7%	41.7%
$R_{\mathrm{grp}} = 2$	C_1	219	26.4%	26.4%
	C_2	109	73.5%	**41.7**%
$R_{\mathrm{grp}} = 3$	C_1	186	25.9%	25.9%
	C_2	135	65.9%	**39.4**%
	C_3	7	11.6%	12.3%
$R_{\mathrm{grp}} = 4$	C_1	211	25.9%	25.9%
	C_2	79	72.0%	**38.8**%
	C_3	34	76.4%	**48.8**%
	C_4	4	11.9%	11.9%
$R_{\mathrm{grp}} = 5$	C_1	153	24.2%	24.2%
	C_2	98	74.2%	**39.1**%
	C_3	61	35.9%	35.9%
	C_4	15	35.9%	**31.4**%
	C_5	1	100.0%	100.0%
$R_{\mathrm{grp}} = 6$	C_1	195	26.1%	26.1%
	C_2	87	74.8%	**38.3**%
	C_3	40	49.4%	49.4%
	C_4	3	34.3%	**19.4**%
	C_5	3	21.4%	**10.7**%
	C_6	0	-	-

as the optimal one. However, such a parameter-set was not suitable for the minority groups. With APS, another parameter-set, which was especially suitable for each of the minority groups was selected to reduce the WER.

Referring to the experiment with $R_{\mathrm{grp}} = 6$, one of the groups included no speech sentences. The centroid was located too far from the whole dataset to form its group in the noise-power vector space during the training process. This shows the proposed method allows redundant groups to disappear in order to minimize the WER. As a result, although an ad hoc number of groups must be given at the beginning of the training process, the number of groups tends to be reduced to an appropriate number, not necessarily the optimal number, automatically. Fig. 3.5 shows the resultant numbers of groups when R_{grp} was increased. The numbers of groups saturated at five even if R_{grp} was increased beyond five. In these experiments, five groups were

TABLE 3.5: WER for whole dataset

	R_{grp}	Training dataset	Evaluation dataset
Conventional method	1	28.9%	41.7%
Proposed method	2	22.9%	31.4%
	3	23.5%	31.0%
	4	22.5%	31.1%
	5	22.4%	31.2%
	6	22.5%	32.1%

enough for the data to reduce the WER by the parameter switching.

Finally, we aggregated the WER for the whole dataset, as shown in Table 3.5. Note that the results listed in Tables 3.4 and 3.5 were obtained from the same experiments, although Table 3.5 includes results for the training dataset as well as the evaluation dataset. The WER reduction for the minority groups resulted in eight to ten-point reductions for the whole evaluation dataset. The experiment using the dataset described above showed that the WER with $R_{grp} \geq 3$ was roughly the same as that with $R_{grp} = 2$. Because there is no method of deriving the optimal R_{grp} deterministically, we suggest using the proposed method with varying R_{grp} and selecting the R_{grp} resulting in the lowest WER.

Overall, the experimental results verified that the proposed method was effective for improving ASR accuracy in noisy environments by switching the noise reduction parameter-sets according to the noise level measured in different frequency bands. It was also found that the proposed method was able to adjust the number of groups automatically.

3.6 Conclusion

We proposed and evaluated APS to improve the accuracy of ASR systems. Noise features are quantified by the noise-power vector measured from noisy speech signals, which is used to group the utterances and assign the best parameter-set values to achieve the highest recognition accuracy. Experiments using datasets in various noisy environments revealed that the WER could be reduced to 31.0% with APS compared to 41.7% with the conventional PSD-BS-BR.

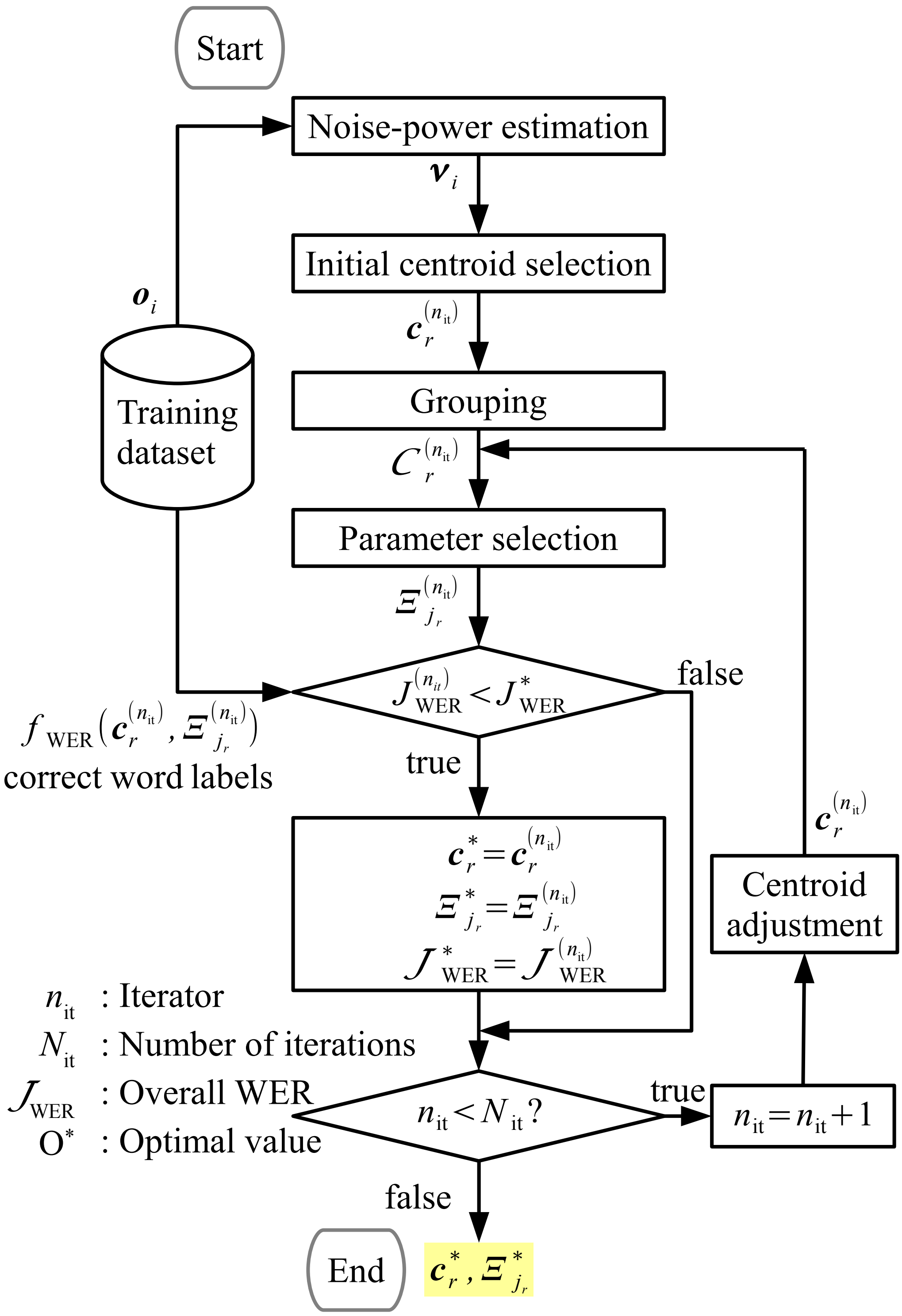


FIGURE 3.2: Flow chart of APS

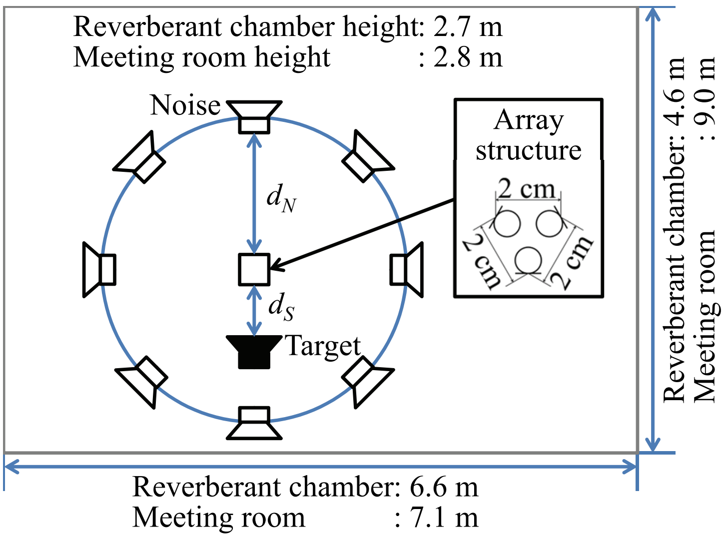
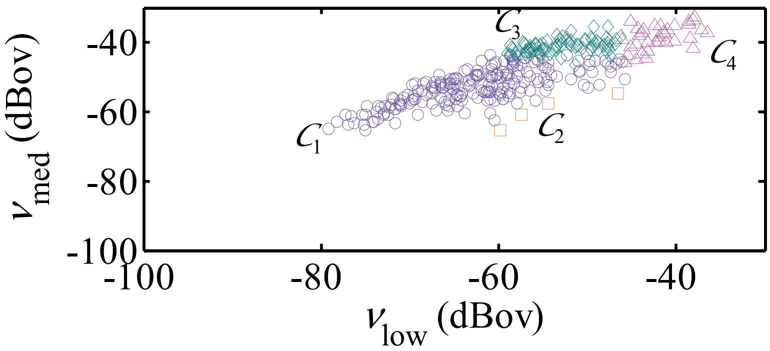
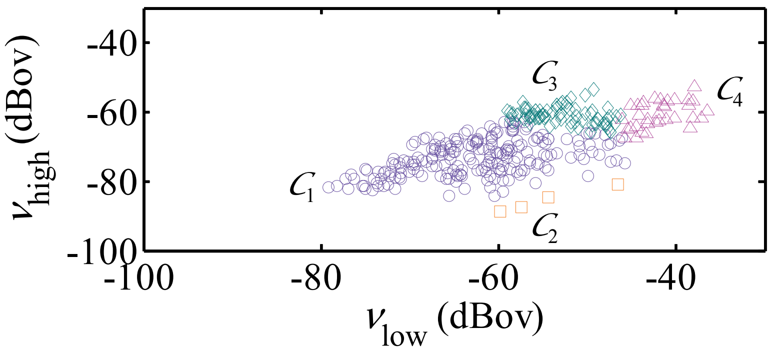


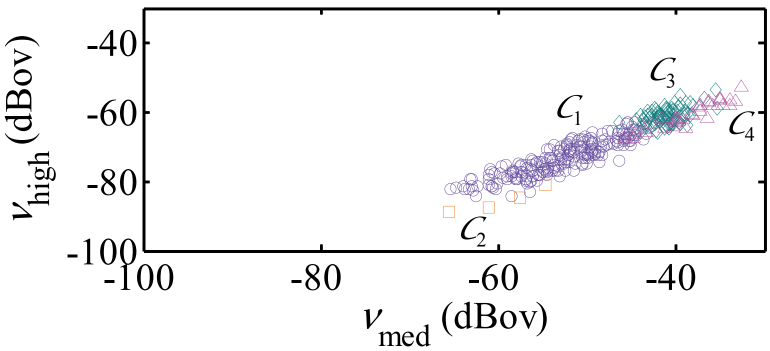
FIGURE 3.3: Noise and impulse response measurement setup to create evaluation data simulating microphone array observation



(A) ν_{low} and ν_{med}



(B) ν_{low} and ν_{high}



(C) ν_{med} and ν_{high}

FIGURE 3.4: Grouping results when $R_{\text{grp}} = 4$

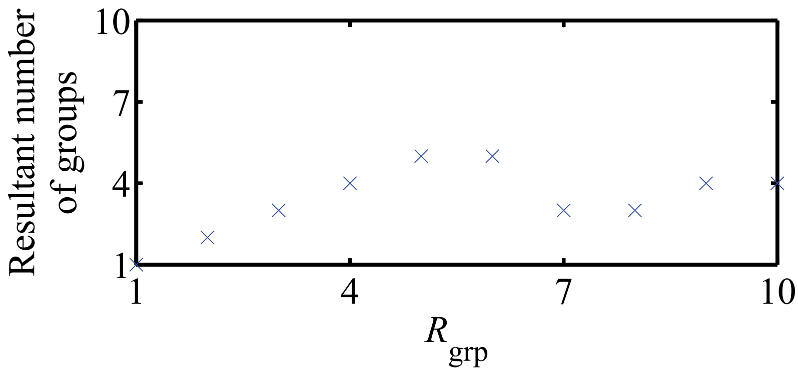


FIGURE 3.5: Relationship between given R_{grp} and resultant number of groups

Chapter 4

Integration of PSD-BS-BR and GMM (PSD-GMM)

4.1 Introduction

In Chapter 3, machine learning is introduced into microphone array speech enhancement in order to use spectral cues about noise. However, the principle of the speech enhancement described in Chapter 3 is same as conventional PSD-BS-BR. In this Chapter, a method for introducing machine learning to use spectral cues about speech is considered.

PSD-BS-BR is a method for estimating the PSDs with low computational complexity, in which it is assumed that the sound sources are sparse in the temporal-spatial domain and that the observed signals include spatial cues to segregate the target sound from the noise. It has been confirmed that using the spatial cues obtained from the microphone array enables accurate estimation of the noise PSD. However, the PSD is not accurately estimated under circumstances beyond this assumption. These errors can cause musical noise or signal distortion.

Machine-learning based speech enhancement methods have also been studied, mainly for the purpose of developing robust ASR in noisy environments. Many methods incorporate machine learning models of target speech PSDs, referred to as speech models, which are pre-trained using certain clean speech corpora, as prior knowledge of a speech. They can accurately preserve the features of the speech spectra, even after noise is reduced, by using the speech models to compose observation models, which represent the observed signals. Some machine-learning based speech enhancement methods designed for single microphone signals are suitable for real-time applications [44, 46, 47, 73, 89]. However, the speech enhancement performance rapidly deteriorates as the SNR of the received signal decreases because using a single microphone limits the accuracy of noise PSD estimation. To overcome this limitation, methods have been developed to attempt to integrate machine-learning based speech enhancement with microphone array [90–92]. However, one merely applies a tandemly connected beamforming and a machine-learning based approach [90]. Others represent noise by machine learning models on the basis of data and require iterative optimization based on batch processing to adapt the model parameters to the environment [91, 92]; thus, they cannot adapt to environmental variation in real time.

Therefore, a method for using signals observed by a microphone array for composing observation models to adapt these models to environmental variation in real time is proposed in this Chapter. In the proposed method, referred to as PSD-GMM, PSD-BS-BR and clean speech model using GMM are integrated to design Wiener filters. It had been already shown that the noise PSD can be robustly estimated under various environments in real time by a combination of PSD-BS-BR on the assumption that the background noise is temporally stationary. As it is hard to estimate the PSD of the target speech from the observed signals, we estimated it using the clean speech models. It is expected that the observation models match the observed signal and that the target speech is clearly extracted in various noisy environments by integrating the noise PSD by using PSD-BS-BR and pre-trained clean speech models.

PSD-GMM is evaluated from three points of view to verify its effectiveness. The first is that PSD-GMM reduces musical noise and signal distortion as well as improves SNR. The second is that the proposed method outperforms PSD-BS-BR in various environments. The third is that a system incorporating PSD-GMM operates in real time. The experimental evaluations are both subjective and objective.

In Section 4.2, a method to compose models of clean speech and observation is proposed. The observation model is composed from the clean speech model and noise PSD estimated by PSD-BS-BR. Then, a method to calculate Wiener post-filter from clean speech model and observation model is explained in Section 4.3. After presenting the experimental results that verify the effectiveness of the proposed method in Section 4.4, this chapter concludes in Section 4.5.

4.2 Target speech and observation model

PSD-BS-BR is able to estimate the speech and noise PSD with certain accuracy and be used to design a Wiener filter deterministically using temporal and spatial cues, given the sparseness of the sound sources. However, the errors in the target PSD estimation increase when noise levels are high and source sparseness is low. The musical noise or signal distortion in the resultant output signal tend to increase as the errors increase.

To overcome the aforementioned problem, our proposed method integrates PSD-BS-BR and a machine-learning based approach by using machine learning models as prior knowledge of the speech signals, limiting the target source to speech. Although the temporal and spatial cues are used in PSD-BS-BR, features of speech in the frequency domain can be used as the spectral cues to estimate the PSDs, by integrating a method for modeling them into PSD-BS-BR. With the machine learning models, we can accurately preserve the features of speech spectra and consequently improve the speech enhancement performance even in very noisy environments with many noise sources.

An overview of the proposed method is given in Fig. 4.1. The essential feature of the machine learning model-based speech enhancement method is distinguishing the characteristic patterns of speech PSD in the observed signals. We compose models of the observed signal, referred to as observation model, to calculate the likelihood for the observation model given the observed signals and recognize patterns in the observed signals. Because the clean speech model is an element of the observation model, we can also determine the speech PSD patterns from the corresponding observation patterns. It is important for composed observation models to precisely represent the observed signals.

The statistical clean speech model is an ergodic HMM with two internal states, i.e., silence ($j = 1$) and speech ($j = 2$), where j denotes the state index, as shown in Fig. 4.2. Each state is modeled using a GMM with I_{GMM} Gaussian components as Eq. (4.1) in the LPSD domain described in Eqs. (2.7) and (2.8).

$$p\left(\boldsymbol{\phi}_S^{\ln}(\tau)\middle| j\right) = \sum_{i=1}^{I_{\mathrm{GMM}}} \lambda_{S,j,i} \mathcal{N}\left(\boldsymbol{\phi}_S^{\ln}(\tau)\middle| \boldsymbol{\mu}_{S,j,i}(\tau), \boldsymbol{\Sigma}_{S,j,i}\right) \tag{4.1}$$

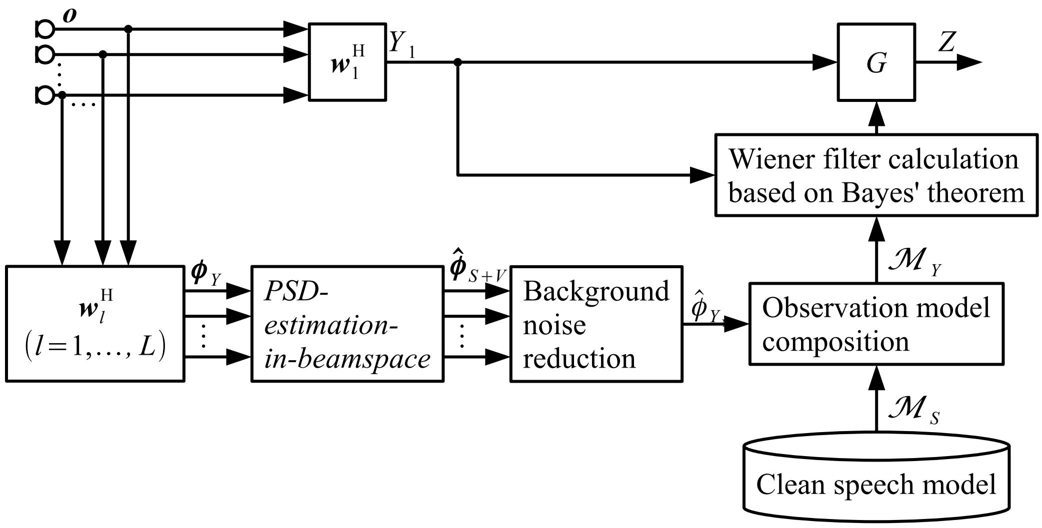


FIGURE 4.1: Overview of PSD-GMM

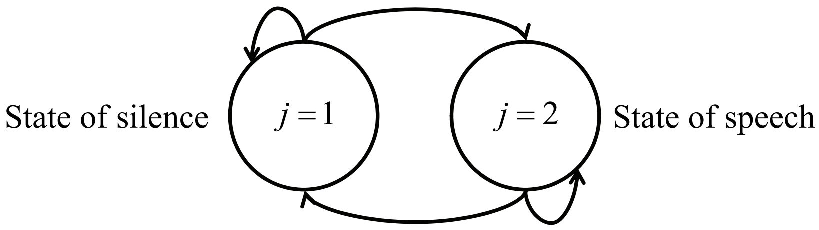


FIGURE 4.2: Statistical clean speech model

The index of the Gaussian component is denoted as i.

The model parameters are indicated together as $\mathcal{M}_S$, as in Eqs. (4.2)–(4.4).

$$\mathcal{M}_S = \left\{\lambda_{S,j,i} \quad \boldsymbol{\mu}_{S,j,i} \quad \boldsymbol{\Sigma}_{S,j,i}\right\} \tag{4.2}$$

$$\boldsymbol{\mu}_{S,j,i} = \begin{bmatrix} \mu_{S,1,j,i} \\ \mu_{S,2,j,i} \\ \vdots \\ \mu_{S,\Omega_{SB},j,i} \end{bmatrix} \tag{4.3}$$

$$\boldsymbol{\Sigma}_{S,j,i} = \mathrm{diag}\left(\sigma^2_{S,1,j,i}, \sigma^2_{S,2,j,i}, \ldots, \sigma^2_{S,\Omega_{SB},j,i}\right) \tag{4.4}$$

These clean speech model parameters are trained in advance by using a training dataset.

The structure of the observation model is the same as that of the clean speech model. The parameters of the observation models in the LPSD domain $\mathcal{M}_Y(\tau)$ are expressed by Eqs. (4.5)–(4.7).

$$\mathcal{M}_Y(\tau) = \left\{ \lambda_{Y,j,i} \quad \boldsymbol{\mu}_{Y,j,i}(\tau) \quad \boldsymbol{\Sigma}_{Y,j,i} \right\} \tag{4.5}$$

$$\boldsymbol{\mu}_{Y,j,i}(\tau) = \begin{bmatrix} \mu_{Y,1,j,i}(\tau) \\ \mu_{Y,2,j,i}(\tau) \\ \vdots \\ \mu_{Y,\Omega_{\mathrm{SB}},j,i}(\tau) \end{bmatrix} \tag{4.6}$$

$$\boldsymbol{\Sigma}_{Y,j,i} = \mathrm{diag}\left(\sigma^2_{Y,1,j,i}, \sigma^2_{Y,2,j,i}, \ldots, \sigma^2_{Y,\Omega_{\mathrm{SB}},j,i}\right) \tag{4.7}$$

The mean vector of the observation models is time variant because it is derived from the noise PSD, which is obtained with PSD-BS-BR, as well as the mean vector of the clean speech model, which is trained in advance. In the PSD domain, Eq. (2.43) is expressed as Eq. (4.8).

$$\begin{aligned}\phi^{\ln}_{Y_{\mathrm{I}},\omega_{\mathrm{SB}}}(\tau) &= \ln\left(\boldsymbol{w}_{\mathrm{AFB},\omega_{\mathrm{SB}}} \begin{bmatrix} \phi_S(\omega,\tau)\,|_{\omega=0} \\ \phi_S(\omega,\tau)\,|_{\omega=1} \\ \vdots \\ \phi_S(\omega,\tau)\,|_{\omega=\Omega_{\mathrm{DFT}}-1} \end{bmatrix} + \boldsymbol{w}_{\mathrm{AFB},\omega_{\mathrm{SB}}} \begin{bmatrix} \phi_{Y_{\mathrm{N}}}(\omega,\tau)\,|_{\omega=0} \\ \phi_{Y_{\mathrm{N}}}(\omega,\tau)\,|_{\omega=1} \\ \vdots \\ \phi_{Y_{\mathrm{N}}}(\omega,\tau)\,|_{\omega=\Omega_{\mathrm{DFT}}-1} \end{bmatrix} \right) \\ &= \phi^{\ln}_{S,\omega_{\mathrm{SB}}}(\tau) + \ln\left\{1 + \exp\left(\phi^{\ln}_{Y_{\mathrm{N}},\omega_{\mathrm{SB}}}(\tau) - \phi^{\ln}_{S,\omega_{\mathrm{SB}}}(\tau)\right)\right\} \\ &= \phi^{\ln}_{S,\omega_{\mathrm{SB}}}(\tau) + f\left(\phi^{\ln}_{S,\omega_{\mathrm{SB}}}(\tau), \phi^{\ln}_{Y_{\mathrm{N}},\omega_{\mathrm{SB}}}(\tau)\right) \end{aligned} \tag{4.8}$$

The mismatch function $f\left(\phi^{\ln}_S(\omega,\tau), \phi^{\ln}_{Y_{\mathrm{N}}}(\omega,\tau)\right)$ is approximated by using zeroth order VTS expansion [46, 47] at the i-th mean of the j-th state clean speech model and the logarithmic estimation of noise PSD and is expressed by Eq. (4.9).

$$f\left(\phi^{\ln}_{S,\omega_{\mathrm{SB}}}(\tau), \phi^{\ln}_{Y_{\mathrm{N}},\omega_{\mathrm{SB}}}(\tau)\right) \approx \ln\left\{1 + \exp\left(\hat{\phi}^{\ln}_{Y_{\mathrm{N}},\omega_{\mathrm{SB}}}(\tau) - \mu_{S,\omega_{\mathrm{SB}},j,i}\right)\right\} \tag{4.9}$$

In the approximation, we assume that the variance in the mismatch function $f\left(\phi^{\ln}_S(\omega,\tau), \phi^{\ln}_{Y_{\mathrm{N}}}(\omega,\tau)\right)$ is negligible. The mean and variance of $\phi^{\ln}_{Y_{\mathrm{I}},\omega_{\mathrm{SB}}}(\tau)$ with respect to the j-th state and i-th Gaussian component are derived using Eqs. (4.10) and (4.11).

$$\mu_{Y,\omega_{\mathrm{SB}},j,i}(\tau) \approx \mu_{S,\omega_{\mathrm{SB}},j,i} + \ln\left\{1 + \exp\left(\hat{\phi}^{\ln}_{\mathrm{Y_N},\omega_{\mathrm{SB}}}(\tau) - \mu_{S,\omega_{\mathrm{SB}},j,i}\right)\right\} \tag{4.10}$$

$$\sigma^2_{Y,\omega_{\mathrm{SB}},j,i}(\tau) \approx \sigma^2_{S,\omega_{\mathrm{SB}},j,i} \tag{4.11}$$

Therefore, the parameters of the observation models are sequentially composed, as shown in Eqs. (4.12)–(4.14).

$$\lambda_{Y,j,i} = \lambda_{S,j,i} \tag{4.12}$$

$$\mu_{Y,\omega_{\mathrm{SB}},j,i}(\tau) = \ln\left\{\exp\left(\mu_{S,\omega_{\mathrm{SB}},j,i}\right) + \boldsymbol{w}_{\mathrm{AFB},\omega_{\mathrm{SB}}}\begin{bmatrix} \hat{\phi}_{\mathrm{Y_N}}(\omega,\tau)\,|_{\omega=0} \\ \hat{\phi}_{\mathrm{Y_N}}(\omega,\tau)\,|_{\omega=1} \\ \vdots \\ \hat{\phi}_{\mathrm{Y_N}}(\omega,\tau)\,|_{\omega=\Omega_{\mathrm{DFT}}-1} \end{bmatrix}\right\} \tag{4.13}$$

$$\boldsymbol{\Sigma}_{Y,j,i} = \boldsymbol{\Sigma}_{S,j,i} \tag{4.14}$$

The number of the Gaussian components in the observation models is same as that of the clean speech models, thus the mixture weight of the observation models is also same as that of the clean speech models. The parameters of the clean speech models are time-invariant, so only $\mu_{Y,\omega_{\mathrm{SB}},j,i}(\tau)$ is updated every time frame by using the noise PSD $\hat{\phi}_{\mathrm{Y_N}}(\omega,\tau)$ in Eq. (2.55).

4.3 Wiener post-filter calculation based on Bayes' theorem

There have been a number of studies on designing Wiener filters on the basis of generative models [46, 73, 89]. The proposed method applies one of these methods that calculate the Wiener filter by simply following Bayes' theorem [46, 89].

After model composition described in Section 4.2, the Wiener post-filter is designed by using the model parameters of the speech $\mathcal{M}_S$ and the beamformer's output $\mathcal{M}_Y(\tau)$. Each model has J_{stt} states, and each state consists of I_{GMM} Gaussian components. The Wiener filter is calculated using Eq. (4.15) if the Ω_{SB}-dimensional LPSD vector of the current beamforming's output $\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)$ is deterministically known to belong to the j-th state and i-th Gaussian component.

$$\mathcal{G}_{\mathrm{GMM},j,i}(\omega,\tau) = \frac{\exp\left(\boldsymbol{w}_{\mathrm{SFB},\omega}\boldsymbol{\mu}_{S,j,i}\right)}{\exp\left(\boldsymbol{w}_{\mathrm{SFB},\omega}\boldsymbol{\mu}_{S,j,i}\right) + \hat{\phi}_{\mathrm{Y_N}}(\omega,\tau)} \tag{4.15}$$

In contrast to Eq. (2.44), the estimated target speech PSD $\hat{\phi}_S(\omega,\tau)$ is substituted with the exponential mean contained in the clean speech model in Eq. (4.15). The target speech component is derived not from the estimation obtained using noisy speech but from estimation obtained using a large quantity of clean speech data. As a result, the target speech component contains less errors and the resultant Wiener filter is less likely to cause distortion.

However, $\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)$ would belong to every state and every component with a certain probability. Therefore, the Wiener filter is expressed as Eqs. (4.16) and (4.17) by weighted summing $\mathcal{G}_{\mathrm{GMM},j,i}(\omega,\tau)$ with respect to each state and each Gaussian component depending on the posterior probability.

$$G_{\mathrm{GMM}}(\omega,\tau) = \sum_{j=1}^{J_{\mathrm{stt}}}\sum_{i=1}^{I_{\mathrm{GMM}}} P\left(j,i\left|\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\right.\right)\cdot\mathcal{G}_{\mathrm{GMM},j,i}(\omega,\tau) \tag{4.16}$$

The $P\left(j,i\left|\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\right.\right)$ denotes the posterior probability with respect to the j-th state and i-th Gaussian component, and it is deformed using Eq. (4.17).

$$\begin{aligned} P\left(j,i\left|\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\right.\right) &= \frac{P\left(j,i,\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\right)}{p\left(\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\right)} \\ &= \frac{P\left(j,i,\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\right)}{P\left(j,\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\right)}\cdot\frac{P\left(j,\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\right)}{p\left(\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\right)} \\ &= P\left(i\left|j,\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\right.\right)P\left(j\left|\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\right.\right) \end{aligned} \tag{4.17}$$

From Bayes' theorem, $P\left(k\left|j,\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\right.\right)$ is expressed by Eq. (4.18).

$$P\left(i\left|j,\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\right.\right) = \frac{p\left(\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\,|j,i\right)P(i\,|j)}{\sum_{i=1}^{I_{\mathrm{GMM}}} p\left(\boldsymbol{\phi}^{\ln}_{\mathrm{Y_1}}(\tau)\,|j,i\right)P(i\,|j)} \tag{4.18}$$

To calculate $P\left(i \middle| j, \boldsymbol{\phi}_{Y_1}^{\ln}(\tau)\right)$, the likelihood of the corresponding Gaussian component $p\left(\boldsymbol{\phi}_{Y_1}^{\ln}(\tau) \middle| j, i\right)$ is calculated as Eq. (4.19).

$$p\left(\boldsymbol{\phi}_{Y_1}^{\ln}(\tau) \middle| j, i\right) = \mathcal{N}\left(\boldsymbol{\phi}_{Y_1}^{\ln}(\tau) \middle| \boldsymbol{\mu}_{Y,j,i}(\tau), \boldsymbol{\Sigma}_{Y,j,i}\right) \tag{4.19}$$

The $P(i|j)$ is regarded as Eq. (4.20).

$$P(i|j) = \lambda_{Y,j,i} \tag{4.20}$$

The $P\left(j | \boldsymbol{\phi}_{Y_1}^{\ln}(\tau)\right)$ is derived using Eq. (4.21), unlike the general method that computes it sequentially with the HMM's state transition probability.

$$P\left(j \middle| \boldsymbol{\phi}_{Y_1}^{\ln}(\tau)\right) = \begin{cases} 1 - G(\omega, \tau) & (j = 1) \\ G(\omega, \tau) & (j = 2) \end{cases} \tag{4.21}$$

Therefore, it is not necessary to train the HMM's state transition probability in advance while its output probability is trained with a conventional method [46].

Finally, the Wiener filter is obtained by substituting Eqs. (4.15), (4.18) and (4.21) into Eq. (4.16).

4.4 Experiment

PSD-GMM was evaluated experimentally in terms of reduced noise power and enhanced speech quality to confirm its effectiveness. We tested MVDR beamformer [9, 78], single-channel VTS method with switching Kalman filter (VTS) [46], and PSD-BS-BR as conventional methods to be compared.

4.4.1 Setup

The microphone array consisted of three cardioid microphones. Each microphone was turned 120° from the others.

Evaluation data were obtained by simulating the microphone array observation as follows. We measured the impulse responses of the target and interference by using the microphone array. The measurements were carried out in two reverberant

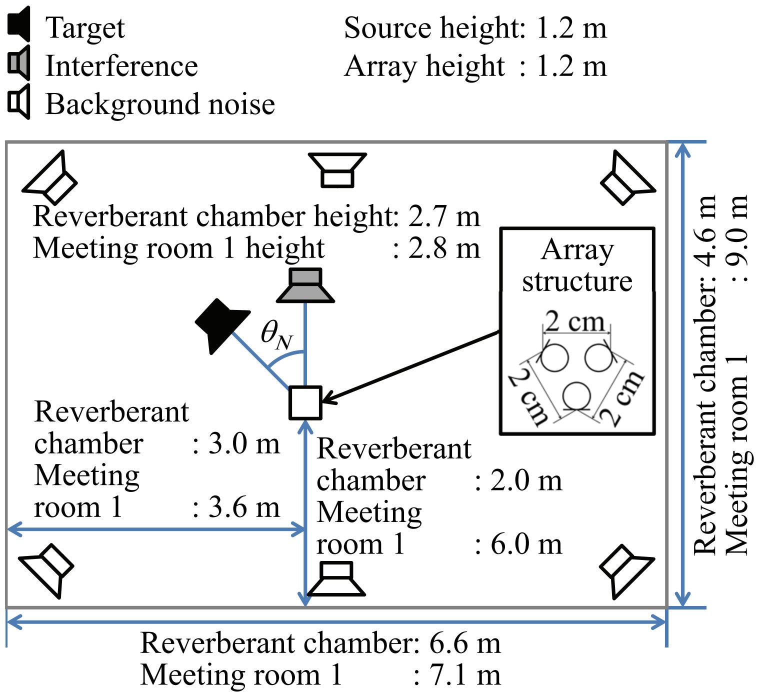
TABLE 4.1: Types and angles of interference noise

Test	# of interference sources, $K-1$	Type of interference noise	Angle, θ_N
1	1	music	180°
2	1	music	135°
3	1	music	90°
4	1	music	45°
5	3	music	45°, 90°, 135°
6	1	speech	180°
7	1	speech	135°
8	1	speech	90°
9	1	speech	45°
10	3	speech	45°, 90°, 135°

chambers and two meeting rooms to confirm that the proposed method is effective in various reverberant environments. The measurement setup is shown in Fig. 5.3. The θ_N denotes the angle between the target and interference. Target clean speech signals and interference signals were convolved with the recorded impulse responses to make the target sound and interference noise. Table 4.1 summarizes the interference source types and angles with the target. One of three different types of background noise recorded in offices, shopping centers, and exhibition halls was played from loudspeakers against a wall. The impulse responses and background noise were measured at different times with the microphone array, and the target sound, interference noise, and background noise were added with different SNRs through simulation. The background noise level was varied from −5 to 10 dB relative to the target, while the interference noise level was the same as the target.

We trained the clean speech models by using a training set of the WSJ0 corpus [85]. This training set was composed of 7138 utterances spoken by 83 individuals. The speech model had $J_{\mathrm{stt}} = 2$ states and each state had $I_{\mathrm{GMM}} = 64$ Gaussian components. The features of the speech model were $\Omega_{\mathrm{SB}} = 40$-dimensional LPSDs.

Clean data for evaluation were taken from the evaluation set of the WSJ0 corpus. Sixteen utterances by four males and four females were used as the target and interference speech under each noise condition for objective evaluation.



(A) Reverberant chambers and meeting room 1

(B) Meeting room 2

FIGURE 4.3: Noise and impulse response measurement setup to create evaluation data simulating microphone array observation

We used signal-to-interference ratio (SIR) as an evaluation index for noise reduction performance and signal-to-distortion ratio (SDR) as an evaluation index for signal distortion, for objective evaluation. The SIR and SDR were obtained by using BSS_EVAL Toolbox [93].

For subjective quality evaluation, we conducted mean opinion score (MOS) tests. Speech signals under two background noise level conditions, -5 and 0 dB relative to speech level, were evaluated. We used four utterances by two males and two females from the evaluation set of the WSJ0 corpus in the tests. The length of one utterance was about six seconds. The evaluation was conducted by fourteen non-expert evaluators. They were asked to comprehensively rate the followings, according to the degradation category scale shown in Table 4.2 compared to reference materials to follow the degradation category rating (DCR) test [94].

- target speech distortion
- noise residual level
- residual noise distortion

Two of the fourteen evaluators participated in DCR test for the first time and the others had experiences of participating in DCR tests. They had made a little practice before participating in this DCR test. The outputs with the ideal Wiener filter were used as the reference materials because the ideal Wiener filter gives the performance upper-bound of the speech enhancement using beamforming and Wiener post-filter. The ideal Wiener filter was calculated using true values of the PSD of the target and noise, which were known through the simulation for obtaining the evaluation data. We removed outliers, and the data for each background noise level condition contained 109 to 112 samples. The outliers were evaluated according to the definition in box plot [95], which is one of general method for handling data. Defining interquartile range (IQR) as the difference between the third and first quartiles, data that were either $1.5 \times \mathrm{IQR}$ or more above the third quartile or $1.5 \times \mathrm{IQR}$ or more below the first quartile were removed as outliers.

Table 4.3 summarizes the experimental conditions. The outputs with the proposed method were obtained on an Intel Xeon CPU E5-2650 2.60GHz machine with Windows operating system.

TABLE 4.2: Degradation category scale

5	Degradation is inaudible.
4	Degradation is audible but not annoying.
3	Degradation is slightly annoying.
2	Degradation is annoying.
1	Degradation is very annoying.

TABLE 4.3: Experimental conditions

	Objective evaluation	Subjective evaluation
Sampling rate	16 kHz	
# of microphones, M	3	
Interference noise condition	Tests 1–10	Tests 6, 10
Source distance	0.25, 1, 2 m	1 m
Background noise level	−5, 0, 5, 10 dB	−5, 0 dB
Reverberation time of reverberant chambers (1 kHz)	230, 350 ms	
Frame length	32 ms	
Frame shift	16 ms	
# of HMM states, J	2	
# of Gaussian mixtures, I_{GMM}	64	
# of filter bank channels, B	40	
Training data	WSJ0	

4.4.2 Objective evaluation results

Table 4.4 and 4.5 summarize the average results of the SIR and SDR evaluation, respectively, corresponding to the conditions of reverberation time and the microphone array location. Fig. 4.4 shows the waveforms and spectrograms of the target signal, observed signal and output signal, and it is shown that the target was successfully enhanced with the proposed method.

The values of background noise level in Table 4.4 and 4.5 indicate the background noise level in the first channel and are relative values to the target speech level. These values do not include the interference noise power; thus, the total noise power was higher than these values. The interference noise level was uniform between Tests 1–4, 6–9. That in Tests 5 and 10 was higher as there were three interference sources.

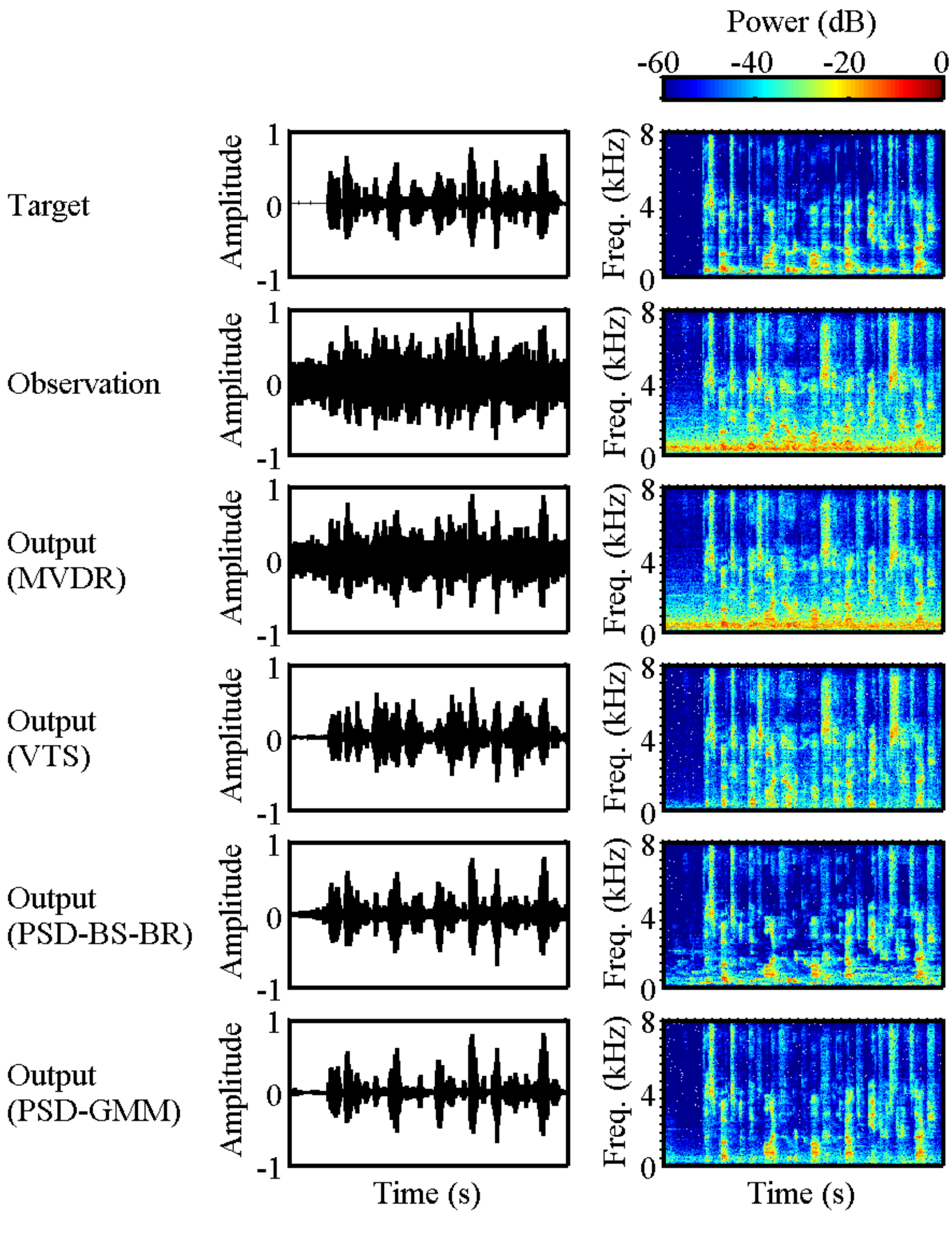


FIGURE 4.4: Waveforms and spectrograms of target source, captured signal, and output signals

In all tests, PSD-GMM outperformed MVDR and PSD-BS-BR. PSD-GMM improved the SIR and SDR by 8.2 dB and 6.2 dB on average compared to MVDR and by 3.4 dB and 2.7 dB on average compared to PSD-BS-BR, respectively. The MVDR method did not perform well because there was background noise arriving from the same directions as the target source. The improvements with PSD-GMM compared with those with PSD-BS-BR were due to estimating the speech PSD by using GMM since the estimation method for noise PSD is common between the two methods. Accurate estimation of speech PSD reduced speech signal distortion and accordingly improved output SIR. The improvements were prominent when the interference noise type was music because a pre-trained speech model segregate non-speech signals from target speech signal better than interfering speech signals. Although VTS performed well when background noise power was relatively high, its performance degraded when interference noise was relatively high. In principle, when the interference noise is speech, it is difficult to reduce the interfering speech signals with VTS. This is because interference speech cannot be distinguished from the target speech by using only VTS. PSD-GMM improved the SIR and SDR by 3.2 dB and 1.7 dB on average, respectively, compared to VTS.

It was also verified that PSD-GMM runs in real time since the real-time factor was 0.32 during the experiment.

4.4.3 Subjective evaluation results

Table 4.6 shows the MOS scores. The differences in the sound quality between the conventional methods and PSD-GMM were confirmed by t-test with 95% confidence intervals. The quality of output from the proposed method was significantly better than those from the other methods when the background noise level was −5 dB. When the background noise level was 0 dB, the quality of output from PSD-GMM was significantly better than those from VTS and PSD-BS-BR and no worse than that of MVDR. The degradation was less annoying when the background noise level was low. Table 4.7 lists the p-values of the t-tests.

Through the aforementioned evaluations, PSD-GMM was confirmed to enhance the target speech in various conditions with little signal distortion.

4.5 Conclusion

A method for integrating PSD-BS-BR and GMM was proposed. The observation models were composed of clean speech models and noise PSDs estimated using PSD-BS-BR. Wiener post-filter was designed based on Bayes' theorem using the observation models and beamformers' output. The experimental results in various noise environments showed that the SIR and SDR improved compared with using the conventional methods. Through subjective evaluations for speech quality, it was shown that PSD-GMM is capable of accurately preserving speech characteristics as well as reducing noise sufficiently.

TABLE 4.4: Results of SIR evaluation (dB)

		Background noise level relative to target speech level (dB)			
		−5	0	5	10
Test 1	MVDR	4.7	1.4	−2.7	−7.2
	VTS	10.8	8.9	6.5	**3.5**
	PSD-BS-BR	9.2	6.3	2.4	−2.4
	PSD-GMM	**12.5**	**10.4**	**7.6**	**3.5**
Test 2	MVDR	5.7	1.9	−2.5	−7.1
	VTS	10.4	8.7	6.3	3.4
	PSD-BS-BR	11.7	7.6	3.0	−2.1
	PSD-GMM	**14.0**	**11.1**	**7.8**	**3.6**
Test 3	MVDR	5.3	1.7	−2.6	−7.1
	VTS	7.5	6.6	5.0	2.6
	PSD-BS-BR	11.4	7.6	3.0	−2.1
	PSD-GMM	**13.5**	**11.0**	**7.9**	**3.7**
Test 4	MVDR	2.4	0.2	−3.2	−7.4
	VTS	5.5	5.1	3.9	2.0
	PSD-BS-BR	5.0	3.6	1.0	−3.0
	PSD-GMM	**7.6**	**6.8**	**5.2**	**2.1**
Test 5	MVDR	1.4	−0.5	−3.5	−7.5
	VTS	2.6	2.5	1.7	0.2
	PSD-BS-BR	5.5	4.0	1.2	−3.0
	PSD-GMM	**7.8**	**7.0**	**5.1**	**2.0**
Test 6	MVDR	3.9	1.1	−2.8	−7.2
	VTS	8.1	6.9	5.0	2.5
	PSD-BS-BR	8.8	6.4	2.6	−2.3
	PSD-GMM	**11.3**	**9.8**	**7.3**	**3.4**
Test 7	MVDR	5.7	1.9	−2.5	−7.1
	VTS	7.1	5.9	4.1	1.8
	PSD-BS-BR	11.9	7.8	3.1	−2.1
	PSD-GMM	**13.9**	**11.0**	**7.7**	**3.5**
Test 8	MVDR	4.6	1.4	−2.7	−7.2
	VTS	2.6	2.2	1.3	−0.4
	PSD-BS-BR	11.5	7.7	3.1	−2.1
	PSD-GMM	**13.0**	**10.5**	**7.5**	**3.4**
Test 9	MVDR	0.7	−0.9	−3.7	−7.5
	VTS	−0.1	−0.3	−0.8	−1.9
	PSD-BS-BR	2.1	1.4	−0.3	−3.5
	PSD-GMM	**2.6**	**2.1**	**1.2**	**−0.6**
Test 10	MVDR	0.1	−1.3	−3.9	−7.6
	VTS	−2.3	−2.5	−2.9	−3.7
	PSD-BS-BR	3.2	2.3	0.2	−3.4
	PSD-GMM	**3.7**	**3.0**	**1.8**	**−0.3**

TABLE 4.5: Results of SDR evaluation (dB)

		Background noise level relative to target speech level (dB)			
		−5	0	5	10
Test 1	MVDR	3.7	0.4	−3.8	−8.2
	VTS	8.5	6.4	4.0	**1.2**
	PSD-BS-BR	6.9	4.0	0.2	−4.6
	PSD-GMM	**8.8**	**6.6**	**4.0**	0.7
Test 2	MVDR	4.5	0.8	−3.6	−8.2
	VTS	8.2	6.2	4.0	**0.0**
	PSD-BS-BR	8.2	4.8	0.5	−4.3
	PSD-GMM	**9.5**	**7.0**	**4.2**	**0.0**
Test 3	MVDR	4.3	0.7	−3.7	−8.2
	VTS	6.0	4.8	3.0	0.6
	PSD-BS-BR	8.1	4.7	0.5	−4.4
	PSD-GMM	**9.3**	**6.9**	**4.2**	**0.8**
Test 4	MVDR	1.9	−0.6	−4.2	−8.4
	VTS	4.3	3.5	2.3	**0.1**
	PSD-BS-BR	3.9	2.1	−0.8	−4.9
	PSD-GMM	**5.6**	**4.4**	**2.6**	−0.2
Test 5	MVDR	0.4	−1.5	−4.6	−8.5
	VTS	1.8	1.4	0.5	−1.2
	PSD-BS-BR	3.5	1.9	−0.9	−5.0
	PSD-GMM	**4.9**	**3.9**	**2.2**	**−0.5**
Test 6	MVDR	2.9	0.0	−3.9	−8.3
	VTS	6.8	5.1	3.1	0.5
	PSD-BS-BR	6.4	3.9	0.2	−4.5
	PSD-GMM	**7.8**	**6.1**	**3.8**	**0.6**
Test 7	MVDR	3.9	0.5	−3.7	−8.2
	VTS	5.9	4.4	2.5	0.0
	PSD-BS-BR	7.9	4.6	0.5	−4.4
	PSD-GMM	**8.9**	**6.6**	**4.0**	**0.6**
Test 8	MVDR	3.6	0.4	−3.8	−8.3
	VTS	2.1	1.4	0.1	−1.8
	PSD-BS-BR	7.8	4.6	0.5	−4.4
	PSD-GMM	**8.7**	**6.5**	**4.0**	**0.6**
Test 9	MVDR	0.3	−1.5	−4.6	−8.5
	VTS	−0.5	−0.9	−1.6	−3.1
	PSD-BS-BR	1.5	0.5	−1.7	−5.2
	PSD-GMM	**1.9**	**1.2**	**0.0**	**−2.0**
Test 10	MVDR	−1.1	−2.5	−5.0	−8.7
	VTS	−2.6	−3.0	−3.5	−4.6
	PSD-BS-BR	1.0	0.0	−2.0	−5.6
	PSD-GMM	**1.4**	**0.8**	**−0.4**	**−2.5**

TABLE 4.6: MOS scores

	Background noise level relative to target speech level (dB)	
	−5	0
MVDR	3.0	2.6
VTS	3.1	2.5
PSD-BS-BR	3.0	2.4
PSD-GMM	**3.5**	**2.8**

TABLE 4.7: P-values (%) of t-tests

	Background noise level relative to target speech level (dB)	
	−5	0
PSD-GMM and MVDR	0.007	23.988
PSD-GMM and VTS	0.467	1.810
PSD-GMM and PSD-BS-BR	0.061	4.155

Chapter 5

PSD estimation using NN (PSD-NN)

5.1 Introduction

In Chapter 3 and 4, machine learning is introduced into microphone array speech enhancement and each of the proposed methods enables to use spectral cues about noise or speech for speech enhancement. In this Chapter, both of spectral cues about speech and those about noise are extracted by uniform machine learning models, using NN.

In PSD-BS-BR, the mapping function between the beamformers' output and the source signals was modelled by a linear function; thus, the estimated PSDs were derived by using a least squares method. However, in fact, the linear function did not accurately represent the relationship of these two signals in the power spectral domain due to an approximation introduced in the modeling. Therefore, a hypothesis is introduced that the estimation accuracy of the PSDs may be improved by replacing the relationship by using a non-linear function. In a recent study [96], DNN was used to map the beamformers' output to the Wiener filter directly. Although this study has demonstrated that using NN delivers a more accurate estimation of the Wiener filter, it requires a large-scale DNN, which is not suitable for some of the practical devices, such as smartphones, wireless headsets, and microphones in vehicles. One could anticipate that the small-scale NN would perform well when the inter-dependency between the input and output variables of the NN is maximized. Should the dependency between the two variables be maximized, the mapping function could become simpler so that it would enable the small-scale NN to estimate the PSDs accurately.

Motivated by these hypotheses, an alternative approach to estimating the Wiener filter from the output of multiple beamformers is introduced in this chapter. The proposed method, referred as to PSD-NN, consists of two steps similar to PSD-BS-BR. The first step estimates the PSD of each sound source (source PSDs) mapped from the PSD of multiple beamformers' output by using NN. Then, the second step simply calculates the Wiener filter from the estimated PSDs by following the definition, which has already been optimized in the sense of MMSE. Because the NN is able to realize a non-linear approximation of the mapping relationship between the source PSDs and the PSD of beamformers' outputs, better performance will be expected for the overall sound source enhancement algorithm.

The detail of PSD-NN is explained in Section 5.2. Experimental results obtained using speech sentences in various noisy environments are shown in Section 5.3 and this chapter is concluded in Section 5.4.

5.2 PSD-NN

As seen in (2.51), PSD-BS-BR introduced a linear function model to approximate the relationship between the source PSDs and the PSD of the multiple beamformers' output. However, various modeling errors occurring in practical applications degrade the estimation accuracy of the method. To improve the overall performance of speech enhancement, PSD-NN replaces the linear function model by NNs optimized by using a preliminary training process, which will be able to represent the relationship more accurately by a non-linear function.

Consider a situation in which the relationship between $\boldsymbol{\phi}_Y(\omega, \tau)$ and each of the $\phi_{Y_S}(\omega, \tau)$, $\phi_{Y_N}(\omega, \tau) - \phi_{\tilde{V}_1}(\omega, \tau)$, and $\phi_{\tilde{V}_1}(\omega, \tau)$ is represented by a NN with J layers as shown in Fig. 5.1. The NN is used to solve a regression problem. The NNs are prepared for each of the $\phi_{Y_S}(\omega, \tau)$, $\phi_{Y_N}(\omega, \tau) - \phi_{\tilde{V}_1}(\omega, \tau)$, and $\phi_{\tilde{V}_1}(\omega, \tau)$, and each frequency band independently; thus, the number of independent NNs is $R_{\mathrm{NN}} = 3\Omega$, where Ω denotes the number of the frequency bins of the source PSDs. The j-th layer of the r-th NN has I_j nodes, receives input features $\boldsymbol{u}_{r,\omega}^{(j)}$, and produces output $\boldsymbol{x}_{r,\omega}^{(j)}$.

A block diagram of speech enhancement using NNs is shown in Fig. 5.2. Two different input features, both generated from the beamformers' output signals, are considered. In addition to the PSDs of the beamformers' output signals themselves,

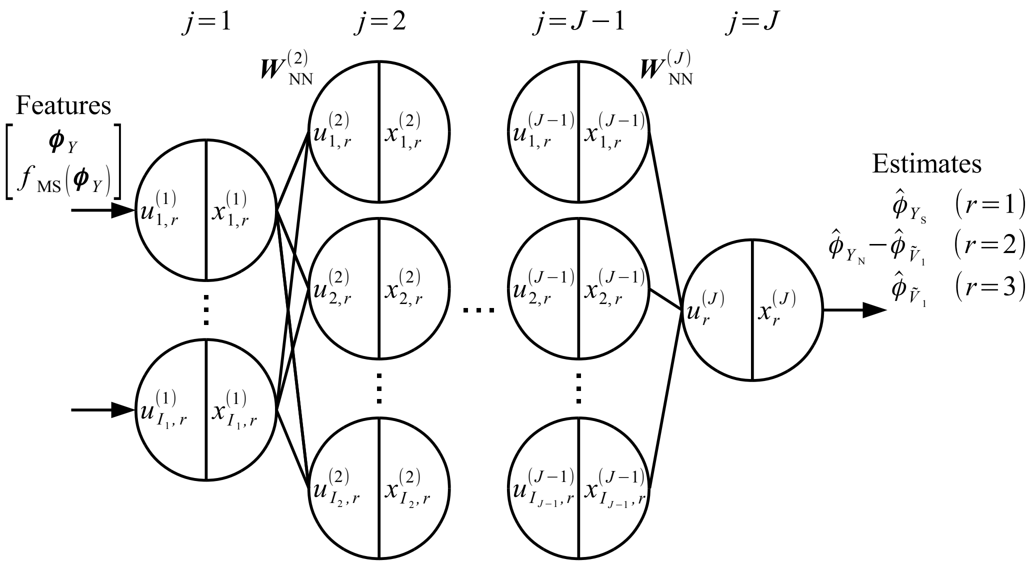


FIGURE 5.1: Diagram of NN

the stationary component calculated by Eq. (5.1) is also used as the input feature.

$$f_{\mathrm{MS}}\left(\boldsymbol{\phi}_Y(\omega,\tau)\right) = \begin{bmatrix} \min_{\tau\in T}\left\{\sum_{q=0}^{\tau-1}\beta(\omega)\left(1-\beta(\omega)\right)^q \phi_{Y_1}(\omega,\tau)\right\} \\ \min_{\tau\in T}\left\{\sum_{q=0}^{\tau-1}\beta(\omega)\left(1-\beta(\omega)\right)^q \phi_{Y_2}(\omega,\tau)\right\} \\ \vdots \\ \min_{\tau\in T}\left\{\sum_{q=0}^{\tau-1}\beta(\omega)\left(1-\beta(\omega)\right)^q \phi_{Y_L}(\omega,\tau)\right\} \end{bmatrix} \tag{5.1}$$

Thus, the output of the input layer, i.e., the input features of NNs, is described as Eq. (5.2).

$$\begin{aligned} \boldsymbol{x}_r^{(1)}(\omega,\tau) &= \begin{bmatrix} x_{1,r}^{(1)}(\omega,\tau) \\ x_{2,r}^{(1)}(\omega,\tau) \\ \vdots \\ x_{2L,r}^{(1)}(\omega,\tau) \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{\phi}_Y(\omega,\tau) \\ f_{\mathrm{MS}}\left(\boldsymbol{\phi}_Y(\omega,\tau)\right) \end{bmatrix} \end{aligned} \tag{5.2}$$

The input and output of nodes in the hidden and output layers are respectively

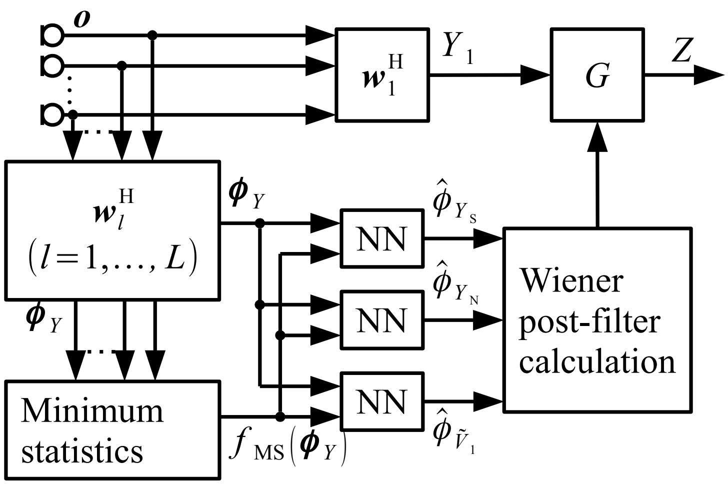


FIGURE 5.2: Procedure of sound source enhancement using NNs

calculated by (5.3) and (5.4) with an activation function $\varphi(\cdot)$ and two kinds of pre-trained parameters: combination weight from the $j-1$ to j-th layer defined in (5.5) and bias given to the j-th layer defined in (5.7).

$$\boldsymbol{u}_{r,\omega}^{(j)} := \begin{bmatrix} u_{1,r,\omega}^{(j)} \\ u_{2,r,\omega}^{(j)} \\ \vdots \\ u_{I_j,r,\omega}^{(j)} \end{bmatrix} = \boldsymbol{W}_{\mathrm{NN},r,\omega}^{(j)} \boldsymbol{x}_{r,\omega}^{(j-1)} + \boldsymbol{b}_{\mathrm{NN},r,\omega}^{(j)} \qquad (j = 2, \ldots, J) \tag{5.3}$$

$$\boldsymbol{x}_{r,\omega}^{(j)} := \begin{bmatrix} x_{1,r,\omega}^{(j)} \\ x_{2,r,\omega}^{(j)} \\ \vdots \\ x_{I_j,r,\omega}^{(j)} \end{bmatrix} = \begin{bmatrix} \varphi\left(u_{1,r,\omega}^{(j)}\right) \\ \varphi\left(u_{2,r,\omega}^{(j)}\right) \\ \vdots \\ \varphi\left(u_{I_j,r,\omega}^{(j)}\right) \end{bmatrix} \qquad (j = 2, \ldots, J) \tag{5.4}$$

$$\boldsymbol{W}_{\mathrm{NN},r,\omega}^{(j)} = \begin{bmatrix} \boldsymbol{w}_{\mathrm{NN},1,r,\omega}^{(j)} & \boldsymbol{w}_{\mathrm{NN},2,r,\omega}^{(j)} & \ldots & \boldsymbol{w}_{\mathrm{NN},I_{j-1},r,\omega}^{(j)} \end{bmatrix} \tag{5.5}$$

$$\boldsymbol{w}_{\mathrm{NN},i,r,\omega}^{(j)} = \begin{bmatrix} W_{\mathrm{NN},1,i,r,\omega}^{(j)} \\ W_{\mathrm{NN},2,i,r,\omega}^{(j)} \\ \vdots \\ W_{\mathrm{NN},I_j,i,r,\omega}^{(j)} \end{bmatrix} \tag{5.6}$$

$$\boldsymbol{b}_{\mathrm{NN},r,\omega}^{(j)} = \begin{bmatrix} b_{\mathrm{NN},1,r,\omega}^{(j)} \\ b_{\mathrm{NN},2,r,\omega}^{(j)} \\ \vdots \\ b_{\mathrm{NN},I_j,r,\omega}^{(j)} \end{bmatrix} \tag{5.7}$$

The combination weight and the bias are trained so as to minimize a cost function. The cost function is the MMSE between the NN's output and the desired source PSD using a training dataset. It is minimized by using back propagation. The units in the hidden layers are rectified linear units (ReLU), which have an activation function

TABLE 5.1: Compared methods

Method ID	Input features	Output estimates
M1	PSD estimation using (2.51)	
M2	$\boldsymbol{\phi}_Y$	G
M3	$\left[\boldsymbol{\phi}_Y \quad f_{\mathrm{MS}}\left(\boldsymbol{\phi}_Y\right)\right]^\top$	G
M4	$\boldsymbol{\phi}_Y$	$\left\{\phi_{Y_\mathrm{S}}, \phi_{Y_\mathrm{N}} - \phi_{\tilde{V}_1}, \phi_{\tilde{V}_1}\right\}$
M5	$\left[\boldsymbol{\phi}_Y \quad f_{\mathrm{MS}}\left(\boldsymbol{\phi}_Y\right)\right]^\top$	$\left\{\phi_{Y_\mathrm{S}}, \phi_{Y_\mathrm{N}} - \phi_{\tilde{V}_1}, \phi_{\tilde{V}_1}\right\}$

defined in (5.8).

$$\varphi\left(u_{i,r,\omega}^{(j)}\right) = \max\left(0, u_{i,r,\omega}^{(j)}\right) \qquad (j = 2, \ldots, J-1) \tag{5.8}$$

Finally the estimated source PSDs are provided by Eqs. (5.9)–(5.11).

$$\hat{\phi}_{Y_\mathrm{S}}(\omega, \tau) = x_{1,\omega}^{(J)} \tag{5.9}$$

$$\hat{\phi}_{Y_\mathrm{N}}(\omega, \tau) - \hat{\phi}_{\tilde{V}_1}(\omega, \tau) = x_{2,\omega}^{(J)} \tag{5.10}$$

$$\hat{\phi}_{\tilde{V}_1}(\omega, \tau) = x_{3,\omega}^{(J)} \tag{5.11}$$

5.3 Experiments

Experiments were conducted to evaluate the performance of the proposed method. Table 5.1 summarizes the methods evaluated in the experiments. To investigate the need for $f_{\mathrm{MS}}\left(\boldsymbol{\phi}_Y(\omega, \tau)\right)$ as a part of the input features of the NNs, the proposed method was tested by using two types of input features, i.e., without and with $f_{\mathrm{MS}}\left(\boldsymbol{\phi}_Y(\omega, \tau)\right)$, defined as M4 and M5, respectively. Since a NN may be able to estimate the Wiener filter directly without explicitly estimating the source PSD like in [96], two cases defined as methods M2 and M3 were also tested. These methods were evaluated by using three metrics: signal-to-noise ratio (SNR), spectral distortion (SD), and root mean square error (RMSE) of the estimated Wiener filter from its ideal value. A high SNR indicates that a higher amount of noise is reduced, while a low SD indicates high reproducibility of the target signal.

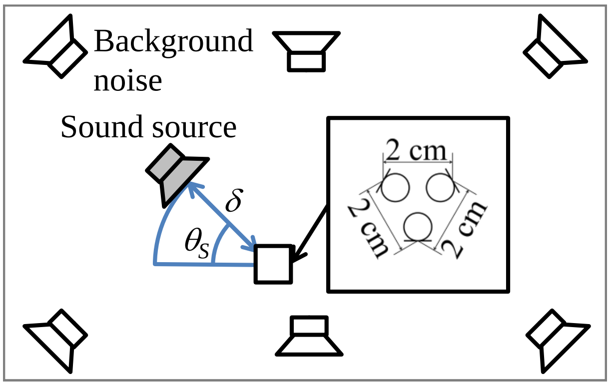


FIGURE 5.3: Noise and impulse response measurement setup

5.3.1 Setup

A corpus of room impulse responses and background noise recorded in various rooms was used to generate the microphone array observation signals. The configuration of the microphone array and loudspeakers used for collecting the corpus are summarized in Fig. 5.3. The microphone array was located at two different positions in each room. It consisted of three cardioid microphones whose directivity pointed at angles 120 degrees apart from each other. Both the impulse responses and background noise were recorded in three different rooms; two of them were used for training the NN, whereas the rest was used for the evaluation of the proposed method.

One of the speech files uttered by four male and four female speakers being convolved with an impulse responses was used as the observation of the target source or the coherent interference. The level of all coherent signals was set the same. Incoherent background noise was then added at different levels ranging from -10 to 10 dB. Table 5.2 summarizes other parameters used in the corpus.

Table 5.3 summarizes the parameters of the methods used in the experiments. The number of frequency bands was compressed to Ω by using a filter bank. Centre frequencies of the filter bank were arranged at equal distance in the equivalent rectangular bandwidth (ERB) scale [97]. The initial values of the combination weight and bias of the NNs were determined by using random initialization.

TABLE 5.2: Details of the corpus

	Training data	Evaluation data
Total length (min)	133	8
# of sound sources, K	2, 3	2, 3
# of utterances	32	16
Direction, θ_{S_1} (deg)	90	90
Direction, θ_{S_2} (deg)	0, 45, 135, 180	45
Direction, θ_{S_3} (deg)	0, 45, 135	0
W × D × H of room (m)	$6.6 \times 4.6 \times 2.7$	$7.1 \times 9.0 \times 2.8$
Distance, δ (cm)	25, 50, 75 100, 150, 200	100
Background noise	office shopping centre	office exhibition hall

TABLE 5.3: Parameters used in processing

# of microphones, M	3
Sampling rate (kHz)	16
Frame length (ms)	16
# of filter bank channels, Ω_{SB}	50
# of beamformers, L	5
# of layers, J	3
# of nodes, I_j ($j = 1, \ldots, J-1$)	10
Learning coefficient $\times 10^3$	10, 5, 2.5
# of iterations	20
Momentum coefficient	0.5 (first 5), 0.9 (after 6)
Decay weight $\times 10^9$	20

5.3.2 Results

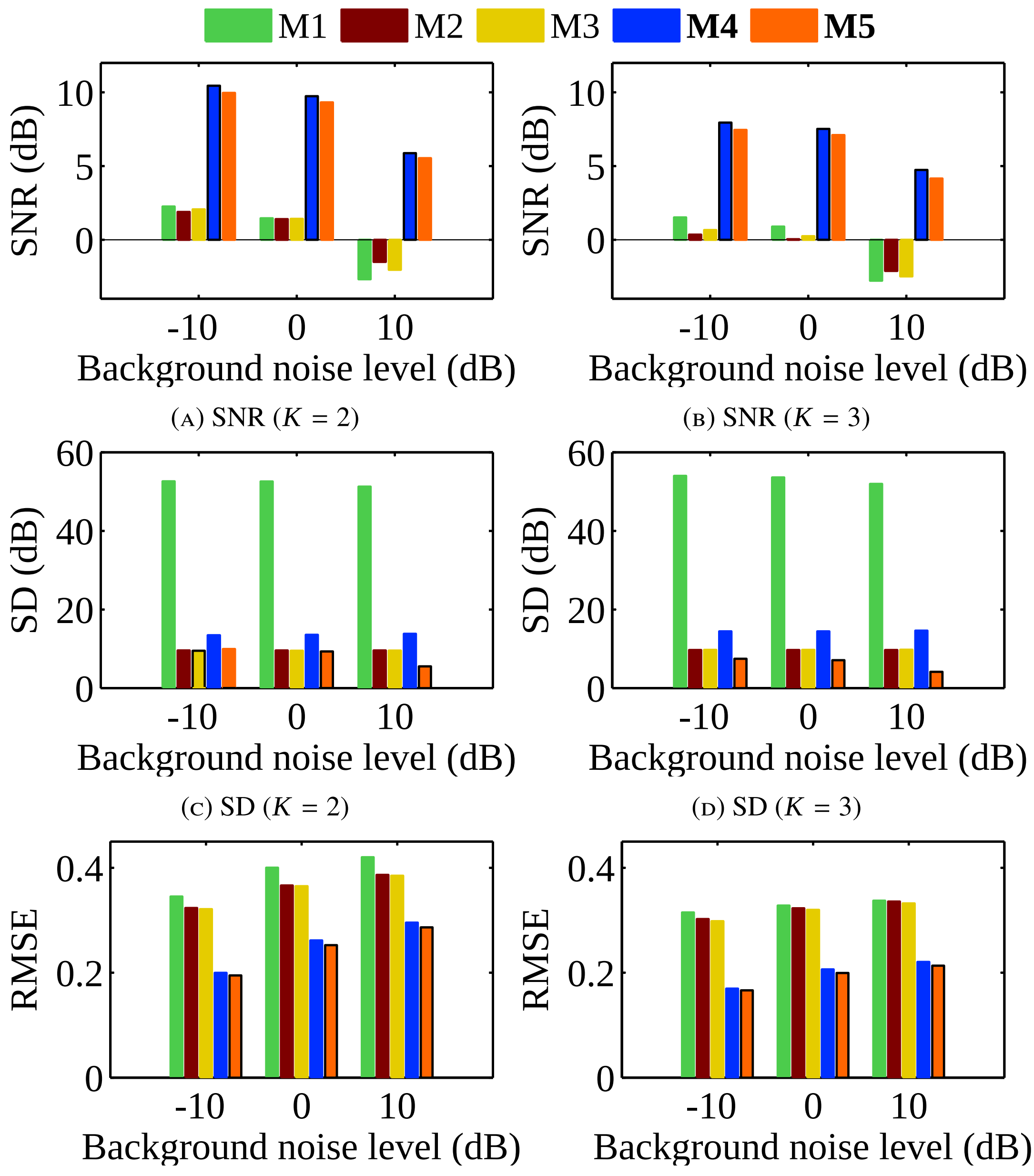
Fig. 5.4 summarizes the SNR, SD, and RMSE of the results given by the five different methods. Overall, M5 estimated the Wiener filter most accurately and achieved the lowest SD, although M4 slightly outperformed M5 in terms of the SNR. Because the Wiener filter obtained by M4 filtered out more signal power than the ideal Wiener filter, the noise was reduced well, but the target speech signals were distorted. M2 and M3 did not outperform even the conventional method M1 except for the RMSE, the improvement of which was also marginal. A possible cause of the failure of M2 and M3 may be the complexity of the relationship between the input and output features of the NNs. In general, it would be easier for a NN to estimate the relationship between its input and output variables if the relationship is simpler. The attempt to directly estimate the Wiener filter may have complicated the relationship and resulted in the failure. Because the Wiener filter has already been optimized in a MMSE sense, it would probably be better to use the NNs to estimate the source PSDs rather than the Wiener filter so that the relationship that the NNs need to replicate would become simpler.

Finally, an example of estimated source PSDs is shown in Fig. 5.5. The values displayed in Fig. 5.5 are the averages of PSDs for all time frames. Overall, the proposed method (M4 and M5) estimated the PSDs of both the target and interference noise more accurately compared with the conventional method (M1); however, M4 overestimated the PSD of the background noise. This would have occurred because NNs used in M4 had less information on the background noise since the NNs did not receive stationary component $\boldsymbol{\phi}_{Y,\mathrm{St}}(\omega, \tau)$ as their input feature.

5.4 Conclusion

An alternative method for estimating source PSDs for calculating the Wiener post-filter was proposed. The proposed PSD-NN applies NNs to represent the relationship between the source PSDs and PSD of beamformers' output signals by a non-linear function. Experiments using a corpus of practical measurements revealed that PSD-NN contributed to estimating the Wiener post-filter more accurately so that the source enhancement performance also outperformed that of the previous method.

In the model proposed in this chapter, spectral cues are utilized within divided frequency bands. Thus, entire spectral structures specific to each sound source are not sufficiently utilized, unlike PSD-GMM. Future work to solve this problem includes investigation of other network configuration such as CNN.



(A) SNR ($K = 2$)

(B) SNR ($K = 3$)

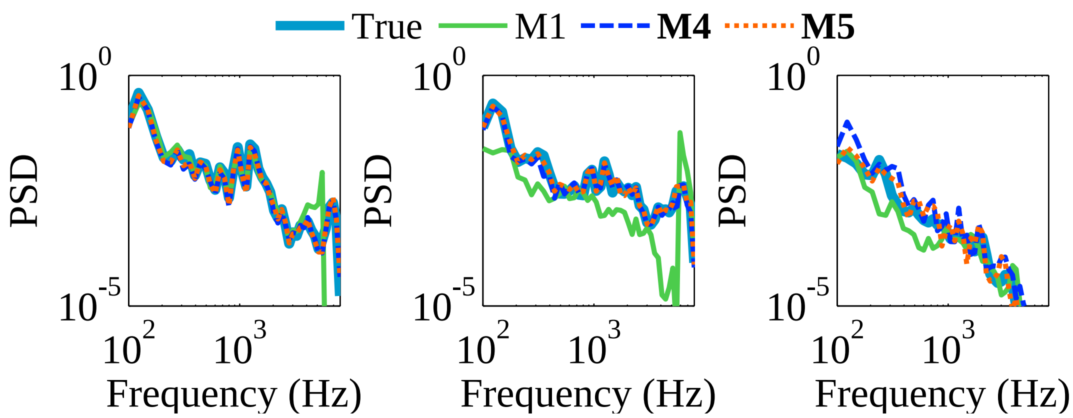
(C) SD ($K = 2$)

(D) SD ($K = 3$)

(E) RMSE of G ($K = 2$)

(F) RMSE of G ($K = 3$)

FIGURE 5.4: Results of SNR, SD, and RMSE of estimated Wiener filter



(A) ϕ_{Y_S} (B) $\phi_{Y_N} - \phi_{\tilde{V}_1}$ (C) $\phi_{\tilde{V}_1}$

FIGURE 5.5: PSD estimation results

Chapter 6

Conclusions

Needs for speech communication and ASR is growing in daily life. The goal of this study was to pick up speech signals clearly even under environments where various types of noise exist, for practical applications. To this end, features of sounds should be captured from different perspectives, i.e., temporal, spatial and spectral cues should be utilized for separating sounds. Additionally, speech enhancement system should be computationally cheap to be applied to many applications.

A conventional framework composed of beamforming and post-filtering was focused on this study. It is because the framework is capable of utilizing temporal and spatial cues effectively and has been verified in many practical applications but further work was needed to improve the performance by utilizing spectral cues. The contribution of this study is proposing methods to design post-filter which is capable of utilizing spectral cues.

Chapter 1 states the background and the purpose of the study. It is pointed out that there already exists methods capable of temporal, spatial and spectral cues but most of them cannot be implemented to local devices with non-powerful computer because of large-scale machine learning model with numerous parameters. Chapter 2 explains PSD-BS-BR, which is composed of beamforming and post-filtering, and single-channel machine-learning based approach.

In Chapter 3, a method is investigated to design post-filter according to noise features when training data of noisy observed signal is obtained. In PSD-BS-BR, post-filter is adjusted by empirically setting values of parameters, which is used for PSD estimation and post-filter calculation. APS is thus proposed, which adopts PSD-BS-BR and introduces switching of the values. To select optimal values of the post-filter parameters, noise-power vector is introduced to quantify the noise features and the training dataset is grouped in the space of the noise-power vector. As a result of evaluation experiments, it is shown that the values of parameters can be automatically set to improve speech enhancement performance.

In Chapter 4, another method is investigated for the case that training data of clean speech signals is obtained. The fact that GMM is often used to model speech signals is focused on, and PSD-GMM is proposed to integrate PSD-BS-BR and machine-learning based approach using GMM. PSD-GMM is capable of accurately estimating speech component utilizing spectral cues about speech signals, and estimating noise component utilizing temporal and spatial cues. As a result, speech distortion and musical noise are reduced, and the experimental results show the effectiveness.

In Chapter 5, PSD-NN is proposed to estimate source PSDs by using training data of noisy observed signals and the corresponding clean speech signals, to design post-filter. PSD of multiple beamformers' output is set to input of NN, emulating the composition of PSD-BS-BR. In PSD-NN, NN approximates the relationship between PSD of beamformers' output and source PSDs. The relationship, which is linearly approximated in PSD-BS-BR, is not very complicated, thus it can be represented by small-scale NN. Therefore, PSD-NN can be implemented to local devices and applied to not only ASR but also speech communication, unlike most of conventional methods using NN. The experimental results show the effectiveness of adopting small-scale NN to estimate source PSD.

In summary, on the one hand, utilizing only physical model like conventional PSD-BS-BR is too simple method to perform in various environments. On the other hand, recently attracting multi-channel deep learning approach is too computationally expensive to implement into various applications. To improve the performance in various environments with reasonably small amount of calculation and memory, machine learning model can be designed to be small sized and integrated to the composition of beamforming and post-filtering. In this way, physical cues and spectral cues can be utilized by the composition of beamforming and post-filtering, and machine learning, respectively. As a future work, the relationship among the scale of machine learning model and the enhancement performance should be investigated. Additionally, more various designs of machine learning model should be also investigated.

References

[1] P. Loizou, *Speech Enhancement: Theory and Practice*, 2nd. Boca Raton, FL, USA: CRC Press, Inc., 2013.

[2] J. Benesty, M. M. Sondhi, and Y. Huang, *Springer Handbook of Speech Processing*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007, ISBN: 3540491252.

[3] N. Ito, E. Vincent, T. Nakatani, N. Ono, S. Araki, and S. Sagayama, "Blind suppression of nonstationary diffuse acoustic noise based on spatial covariance matrix decomposition", *Journal of Signal Processing Systems*, vol. 79, no. 2, pp. 145–157, 2015.

[4] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 27, no. 2, pp. 113–120, Apr. 1979.

[5] N. Wiener, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series: With Engineering Applications*. Cambridge, MA, USA: MIT Press, 1949.

[6] A. Cichocki and R. Unbehauen, "Robust neural networks with on-line learning for blind identification and blind separation of sources", *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 43, no. 11, pp. 894–906, 1996.

[7] A. Hyvärinen, J. Karhunen, and E. Oja, Eds., *Independent Component Analysis*. New Jersey: Wiley, 2001.

[8] M. Brandstein and D. Ward, Eds., *Microphone Arrays: Signal Processing Techniques and Applications*. Berlin Heidelberg: Springer, 2001.

[9] D. H. Johnson and D. E. Dudgeon, *Array Signal Processing: Concepts and Techniques*. Englewood Cliffs, NJ: Simon & Schuster, 1993.

[10] J. Flanagan, D. Berkley, G. Elko, J. West, and M. Sondhi, "Autodirective microphone systems", *Acta Acustica united with Acustica*, vol. 73, no. 2, pp. 58–71, 1991.

[11] Y. Huang, J. Benesty, and J. Chen, *Acoustic MIMO signal processing*. Springer Science & Business Media, 2006.

[12] J. Capon, "High-resolution frequency-wavenumber spectrum analysis", *Proceedings of the IEEE*, vol. 57, no. 8, pp. 1408–1418, 1969.

[13] O. L. Frost, "An algorithm for linearly constrained adaptive array processing", *Proceedings of the IEEE*, vol. 60, no. 8, pp. 926–935, 1972.

[14] R. A. Monzingo and T. W. Miller, *Introduction to adaptive arrays*. Scitech publishing, 1980.

[15] L. Griffiths and C. Jim, "An alternative approach to linearly constrained adaptive beamforming", *IEEE Transactions on antennas and propagation*, vol. 30, no. 1, pp. 27–34, 1982.

[16] R. Zelinski, "A microphone array with adaptive post-filtering for noise reduction in reverberant rooms", in *in Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process. (ICASSP'88)*, vol. 5, Apr. 1988, pp. 2578–2581.

[17] C. Marro, Y. Mahieux, and K. Simmer, "Analysis of noise reduction and dereverberation techniques based on microphone arrays with postfiltering", *IEEE Trans. Speech Audio Process.*, vol. 6, no. 3, pp. 240–259, 1998.

[18] I. McCowan and H. Bourlard, "Microphone array post-filter based on noise field coherence", *IEEE Trans. Speech Audio Process.*, vol. 11, no. 6, pp. 709–716, 2003.

[19] S. Lefkimmiatis, D. Dimitriadis, and P. Maragos, "An optimum microphone array post-filter for speech applications.", in *Interspeech*, 2006.

[20] A. H. Kamkar-Parsi and M. Bouchard, "Improved noise power spectrum density estimation for binaural hearing aids operating in a diffuse noise field environment", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 4, pp. 521–533, 2009.

[21] K. Kumatani, B. Raj, R. Singh, and J. W. McDonough, "Microphone array post-filter based on spatially-correlated noise measurements for distant speech recognition", in *13th Annual Conference of the International Speech Communication Association, Portland, Oregon, USA*, 2012, pp. 298–301.

[22] Y. Hioka, K. Furuya, K. Kobayashi, K. Niwa, and Y. Haneda, "Underdetermined sound source separation using power spectrum density estimated by combination of directivity gain", *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 6, pp. 1240–1250, 2013.

[23] J. Watt, R. Borhani, and A. K. Katsaggelos, *Machine Learning Refined: Foundations, Algorithms, and Applications*. Cambridge University Press, 2016. DOI: 10.1017/CBO9781316402276.

[24] D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, Eds., *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*. Cambridge, MA, USA: MIT Press, 1986.

[25] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, *et al.*, "Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups", *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[26] G. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets", *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[27] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition", *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[28] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition", *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[29] T. Mikolov, M. Karafiát, L. Burget, J. Černockỳ, and S. Khudanpur, "Recurrent neural network based language model", in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[30] O. Yilmaz and S. Rickard, "Blind separation of speech mixtures via time-frequency masking", *IEEE Transactions on signal processing*, vol. 52, no. 7, pp. 1830–1847, 2004.

[31] S. Srinivasan, N. Roman, and D. Wang, "Binary and ratio time-frequency masks for robust speech recognition", *Speech Communication*, vol. 48, no. 11, pp. 1486–1501, 2006.

[32] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition", in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, IEEE, 2013, pp. 7092–7096.

[33] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 6, pp. 1109–1121, 1984.

[34] R. McAulay and M. Malpass, "Speech enhancement using a soft-decision noise suppression filter", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 2, pp. 137–145, 1980.

[35] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription", in *2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (IEEE Cat. No.03TH8684)*, Oct. 2003, pp. 177–180.

[36] C. Févotte, N. Bertin, and J.-L. Durrieu, "Nonnegative matrix factorization with the itakura-saito divergence: with application to music analysis", *Neural computation*, vol. 21, no. 3, pp. 793–830, 2009.

[37] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization", *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[38] A. T. Cemgil, "Bayesian inference for nonnegative matrix factorisation models", *Computational intelligence and neuroscience*, vol. 2009, 2009.

[39] M. N. Schmidt, O. Winther, and L. K. Hansen, "Bayesian non-negative matrix factorization", in *International Conference on Independent Component Analysis and Signal Separation*, Springer, 2009, pp. 540–547.

[40] P. Smaragdis, B. Raj, and M. Shashanka, “Supervised and semi-supervised separation of sounds from single-channel mixtures”, *Independent Component Analysis and Signal Separation*, pp. 414–421, 2007.

[41] T. Virtanen, “Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria”, *IEEE transactions on audio, speech, and language processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[42] S. A. Raczyński, N. Ono, and S. Sagayama, “Multipitch analysis with harmonic nonnegative matrix approximation”, in *in ISMIR 2007, 8th International Conference on Music Information Retrieval*, Citeseer, 2007.

[43] A. Ozerov, C. Févotte, and M. Charbit, “Factorial scaled hidden markov model for polyphonic audio representation and source separation”, in *Applications of Signal Processing to Audio and Acoustics, 2009. WASPAA'09. IEEE Workshop on*, IEEE, 2009, pp. 121–124.

[44] J. C. Segura, Á. de la Torre, M. C. Benítez, and A. M. Peinado, “Model-based compensation of the additive noise for continuous speech recognition. experiments using the aurora II database and tasks”, in *7th European Conference on Speech Communication and Technology, 2nd INTERSPEECH Event*, 2001, pp. 221–224.

[45] S. T. Roweis, “Factorial models and refiltering for speech separation and denoising”, in *Eighth European Conference on Speech Communication and Technology*, 2003.

[46] M. Fujimoto, K. Ishizuka, and T. Nakatani, “A study of mutual front-end processing method based on statistical model for noise robust speech recognition”, in *10th Annual Conference of the International Speech Communication Association, Brighton, United Kingdom*, 2009, pp. 1235–1238.

[47] P. J. Moreno, B. Raj, and R. M. Stern, “A vector Taylor series approach for environment-independent speech recognition”, in *in Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process. (ICASSP'96)*, vol. II, May 1996, pp. 733–736.

[48] Y. Wang and D. Wang, "Towards scaling up classification-based speech separation", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 7, pp. 1381–1390, Jul. 2013.

[49] 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 (TASLP)*, vol. 23, no. 1, pp. 7–19, 2015.

[50] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks", in *Advances in neural information processing systems*, 2007, pp. 153–160.

[51] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders", in *Proceedings of the 25th international conference on Machine learning*, ACM, 2008, pp. 1096–1103.

[52] K. Han, Y. He, D. Bagchi, E. Fosler-Lussier, and D. Wang, "Deep neural network based spectral feature mapping for robust speech recognition", in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[53] X. Feng, Y. Zhang, and J. Glass, "Speech feature denoising and dereverberation via deep autoencoders for noisy reverberant speech recognition", in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, IEEE, 2014, pp. 1759–1763.

[54] J. Du, L.-R. Dai, and Q. Huo, "Synthesized stereo mapping via deep neural networks for noisy speech recognition", in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, IEEE, 2014, pp. 1764–1768.

[55] Y. Ueda, L. Wang, A. Kai, and B. Ren, "Environment-dependent denoising autoencoder for distant-talking speech recognition", *EURASIP Journal on Advances in Signal Processing*, vol. 2015, no. 1, p. 92, 2015.

[56] L. Hui, M. Cai, C. Guo, L. He, W.-Q. Zhang, and J. Liu, "Convolutional maxout neural networks for speech separation", in *Signal Processing and Information Technology (ISSPIT), 2015 IEEE International Symposium on*, IEEE, 2015, pp. 24–27.

[57] F. Weninger, J. R. Hershey, J. Le Roux, and B. Schuller, "Discriminatively trained recurrent neural networks for single-channel speech separation", in *Signal and Information Processing (GlobalSIP), 2014 IEEE Global Conference on*, IEEE, 2014, pp. 577–581.

[58] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: discriminative embeddings for segmentation and separation", in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, IEEE, 2016, pp. 31–35.

[59] K. Kinoshita, M. Delcroix, A. Ogawa, T. Higuchi, and T. Nakatani, "Deep mixture density network for statistical model-based feature enhancement", in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, IEEE, 2017, pp. 251–255.

[60] A. Narayanan and D. Wang, "Joint noise adaptive training for robust automatic speech recognition", in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, IEEE, 2014, pp. 2504–2508.

[61] N. Roman, D. Wang, and G. J. Brown, "Speech segregation based on sound localization", *The Journal of the Acoustical Society of America*, vol. 114, no. 4, pp. 2236–2252, 2003.

[62] M. I. Mandel, R. J. Weiss, and D. P. Ellis, "Model-based expectation-maximization source separation and localization", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 2, pp. 382–394, 2010.

[63] 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, no. 3, pp. 516–527, 2011.

[64] Y. Jiang, D. 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 (TASLP)*, vol. 22, no. 12, pp. 2112–2121, 2014.

[65] A. Ozerov and C. Févotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 550–563, 2010.

[66] H. Sawada, H. Kameoka, S. Araki, and N. Ueda, "Multichannel extensions of non-negative matrix factorization with complex-valued data", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 971–982, 2013.

[67] D. Kitamura, H. Saruwatari, H. Kameoka, Y. Takahashi, K. Kondo, and S. Nakamura, "Multichannel signal separation combining directional clustering and nonnegative matrix factorization with spectrogram restoration", *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 23, no. 4, pp. 654–669, 2015.

[68] D. Liu, P. Smaragdis, and M. Kim, "Experiments on deep learning for speech denoising", in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[69] S. Araki, T. Hayashi, M. Delcroix, M. Fujimoto, K. Takeda, and T. Nakatani, "Exploring multi-channel features for denoising-autoencoder-based speech enhancement", in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, IEEE, 2015, pp. 116–120.

[70] T. N. Sainath, R. J. Weiss, K. W. Wilson, A. Narayanan, M. Bacchiani, *et al.*, "Speaker location and microphone spacing invariant acoustic modeling from raw multichannel waveforms", in *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*, IEEE, 2015, pp. 30–36.

[71] T. Hori, Z. Chen, H. Erdogan, J. R. Hershey, J. Le Roux, V. Mitra, and S. Watanabe, "Multi-microphone speech recognition integrating beamforming, robust feature extraction, and advanced dnn/rnn backend", *Computer Speech & Language*, 2017.

[72] T. Wolff and M. Buck, "A practical beamformer-postfilter system for adaptive speech enhancement in non-stationary noise environments", in *Hands-free Speech Communication and Microphone Arrays (HSCMA), 2011 Joint Workshop on*, IEEE, 2011, pp. 159–160.

[73] T. Arakawa, M. Tsujikawa, and R. Isotani, "Model-based wiener filter for noise robust speech recognition", in *in Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process. (ICASSP'06)*, vol. I, 2006.

[74] S. Saleem, A. T. Piercy, M. Typrin, S. Somashekar, and K. W. Piersol, *Multiple-source speech dialog input*, US Patent 9792901B1, Oct. 2017.

[75] T. N. Sainath, R. J. Weiss, K. W. Wilson, B. Li, A. Narayanan, E. Variani, M. Bacchiani, I. Shafran, A. Senior, K. Chin, *et al.*, "Multichannel signal processing with deep neural networks for automatic speech recognition", *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 5, pp. 965–979, 2017.

[76] Y. Hioka and K. Niwa, "PSD estimation in beamspace for source separation in a diffuse noise field", in *in Proc. Int. Workshop Acoust. Echo Noise Control*, Sep. 2014, pp. 85–88.

[77] J. G. Proakis and D. G. Manolakis, *Introduction to digital signal processing*. Prentice Hall Professional Technical Reference, 1988.

[78] M. Wolfel and J. McDonough, "Minimum variance distortionless response spectral estimation", *IEEE Signal Process. Mag.*, vol. 22, no. 5, pp. 117–126, Sep. 2005.

[79] K. Niwa, Y. Hioka, and K. Kobayashi, "Post-filter design for speech enhancement in various noisy environments", in *in Proc. Int. Workshop Acoust. Echo Noise Control*, Sep. 2014, pp. 35–39.

[80] R. Martin, "Noise power spectral density estimation based on optimal smoothing and minimum statics", *Speech and Audio Processing IEEE Transacitions on*, vol. 9, no. 5, 2001.

[81] S. Sakauchi, A. Nakagawa, Y. Haneda, and A. Kataoka, “Implementing and evaluating an audio teleconferencing terminal with noise and echo reduction”, in *Proc. 8th International Workshop on Acoustic Echo and Noise Control (IWAENC 2003)*, 2003, pp. 191–194.

[82] T. Arakawa, H. Al-Hassanieh, M. Tsujikawa, and R. Isotani, “Extended minimum classification error training in voice activity detection”, in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, Nov. 2009, pp. 232–236.

[83] K. Horii, T. Fukumori, M. Morise, T. Nishiura, and Y. Yamashita, “The determination of dynamic subtraction for spectral subtraction towards musical tone reduction”, *The Journal of the Acoustical Society of America*, vol. 131, no. 4, pp. 3446–3446, 2012.

[84] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2010.

[85] J. Garofolo, D. Graff, D. Paul, and D. Pallett, *CSR-I (WSJ0) complete*, https://catalog.ldc.upenn.edu/LDC93S6A.

[86] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, “The kaldi speech recognition toolkit”, in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, 2011.

[87] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, “The third ’CHiME’ speech separation and recognition challenge: dataset, task and baselines”, in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, Dec. 2015, pp. 504–511.

[88] ITU-T, *G.191 : software tools for speech and audio coding standardization*, https://www.itu.int/rec/T-REC-G.191.

[89] M. Fujimoto and S. Nakamura, “Particle filter based non-stationary noise tracking for robust speech recognition”, in *in Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process. (ICASSP’05)*, vol. I, Mar. 2005, pp. 257–260.

[90] W. Herbordt, T. Horiuchi, M. Fujimoto, T. Jitsuhiro, and S. Nakamura, "Hands-free speech recognition and communication on PDAs using microphone array technology", in *in Proc. IEEE Workshop Automatic Speech Recognition and Understanding*, Nov. 2005, pp. 302–307.

[91] X. Zhao and Z. Ou, "Closely coupled array processing and model-based compensation for microphone array speech recognition", *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 3, pp. 1114–1122, Mar. 2007.

[92] T. Nakatani, S. Araki, T. Yoshioka, M. Delcroix, and M. Fujimoto, "Dominance based integration of spatial and spectral features for speech enhancement", *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 12, pp. 2516–2531, 2013.

[93] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation", *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 14, no. 4, pp. 1462–1469, 2006.

[94] ITU-T Rec. P. 800 Annex D, *Degradation category rating (DCR) method*, 1996.

[95] M. Frigge, D. C. Hoaglin, and B. Iglewicz, "Some implementations of the boxplot", *The American Statistician*, vol. 43, no. 1, pp. 50–54, 1989.

[96] K. Niwa, Y. Koizumi, T. Kawase, K. Kobayashi, and Y. Hioka, "Pinpoint extraction of distant sound source based on dnn mapping from multiple beamforming outputs to prior snr", in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, May 2016, pp. 435–439.

[97] B. C. J. Moore and B. R. Glasberg, "Suggested formulae for calculating auditory-filter bandwidths and excitation patterns", *The Journal of the Acoustical Society of America*, vol. 74, no. 3, pp. 750–753, 1983.

List of Publications

Journal articles

1. T. Kawase, K. Niwa, Y. Hioka, and K. Kobayashi, "Automatic Parameter Switching of Noise Reduction for Speech Recognition", *Journal of Signal Processing,* vol. 21, no. 2, pp. 63-71, 2017.

2. T. Kawase, K. Niwa, M. Fujimoto, K. Kobayashi, S. Araki, and T. Nakatani, "Integration of spatial cue-based noise reduction and speech model-based source restoration for real time speech enhancement", *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences,* vol. 100, no. 5, pp. 1127–1136, 2017.

Conference papers

1. T. Kawase, K. Niwa, Y. Hioka, and K. Kobayashi, "Selection of optimal array noise reduction parameter set for accurate speech recognition in various noisy environments", in *Western Pacific Acoustics Conference*, 2015.

2. K. Niwa, Y. Koizumi, T. Kawase, K. Kobayashi, and Y. Hioka, "Pinpoint extraction of distant sound source based on DNN mapping from multiple beamforming outputs to prior SNR", in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on,* 2016, pp. 435-439.

3. T. Kawase, K. Niwa, M. Fujimoto, N. Kamado, K. Kobayashi, S. Araki, and T. Nakatani, "Real-time integration of statistical model-based speech enhancement with unsupervised noise PSD estimation using microphone array", in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on,* 2016, pp. 604-608.

4. K. Niwa, T. Kawase, K. Kobayashi, and Y. Hioka, "PSD estimation in beamspace using property of M-matrix", in *Acoustic Signal Enhancement (IWAENC), 2016 IEEE International Workshop on,* 2016.

5. T. Kawase, K. Niwa, K. Kobayashi, and Y. Hioka, "Application of neural network to source PSD estimation for wiener filter based array sound source enhancement", in *Acoustic Signal Enhancement (IWAENC), 2016 IEEE International Workshop on,* 2016.

6. K. Niwa, Y. Koizumi, T. Kawase, K. Kobayashi, and Y. Hioka, "Supervised source enhancement composed of nonnegative auto-encoders and complementarity subtraction", in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on,* 2017, pp. 266-270.