Data-driven Speech Enhancement

from Non-negative Matrix Factorization to Deep Representation Learning

Xiang, Yang

DOI (link to publication from Publisher):
10.54337/aau519583502

Publication date:
2022

Document Version
Publisher's PDF, also known as Version of record

DATA-DRIVEN SPEECH ENHANCEMENT: FROM NON-NEGATIVE MATRIX FACTORIZATION TO DEEP REPRESENTATION LEARNING

BY
YANG XIANG

DISSERTATION SUBMITTED 2022

AALBORG UNIVERSITY
DENMARK

Data-driven Speech Enhancement: from Non-negative Matrix Factorization to Deep Representation Learning

Ph.D. Dissertation

Yang Xiang

Department of Architecture, Design, and Media Technology
Aalborg University
Rendsburggade 14, 9000 Aalborg, Denmark

Dissertation submitted Oct, 2022

Dissertation submitted: December 2022

PhD supervisor: Prof. Mads Græsbøll Christensen
Aalborg University

PhD co-supervisors: Dr. Morten Højfeldt Rasmussen
Capturi A/S

Dr. Jesper Lisby Højvang
Capturi A/S

PhD committee: Associate Professor Erkut Cumhur (chair)
Aalborg University, Denmark

Professor Wenwu Wang
University of Surrey, England

Professor Nilesh Madhu
Ghent University – imec, Belgium

PhD Series: Technical Faculty of IT and Design, Aalborg University

Department: Department of Architecture, Design and Media Technology

ISSN (online): 2446-1628
ISBN (online): 978-87-7573-783-3

Published by:
Aalborg University Press
Kroghstræde 3
DK – 9220 Aalborg Ø
Phone: +45 99407140
aauf@forlag.aau.dk
forlag.aau.dk

Printed in Denmark by Stibo Complete, 2023

Curriculum Vitae

Yang Xiang

Yang Xiang was born in Kunming, Yunnan, China, in 1994. He received his B.S. and M.Sc. degrees in information and communication engineering from the Beijing University of Technology in 2016 and 2019, respectively. He is currently an industrial Ph.D. student at the Audio Analysis Lab, Department of Architecture, Design and Media Technology, Aalborg University, Aalborg, and Capturi A/S, Aarhus, Denmark. His research interests include speech enhancement, source separation, deep representation learning, and machine learning.

Abstract

In natural listening environments, speech signals are easily distorted by various acoustic interference, which reduces the speech quality and intelligibility of human listening; meanwhile, it makes difficult for many speech-related applications, such as automatic speech recognition (ASR). Thus, many speech enhancement (SE) algorithms have been developed in the past decades. However, most current SE algorithms are difficult to capture underlying speech information (e.g., phoneme) in the SE process. This causes it to be challenging to know what specific information is lost or interfered with in the SE process, which limits the application of enhanced speech. For instance, some SE algorithms aimed to improve human listening usually damage the ASR system.

The objective of this dissertation is to develop SE algorithms that have the potential to capture various underlying speech representations (information) and improve the quality and intelligibility of noisy speech. This study starts by introducing the hidden Markov model (HMM) into the Non-negative Matrix Factorization (NMF) model (NMF-HMM) because HMM is a convenient way to find underlying speech information for better SE performance. The key idea is applying HMM to capture the underlying speech temporal dynamics information in the NMF model. Additionally, a computationally efficient method is also proposed to ensure that this NMF-HMM model can achieve fast online SE.

Although NMF-HMM captures the underlying speech information, it is difficult to explain what detailed information is obtained. In addition, NMF-HMM cannot represent the underlying information in a vector form, which makes information analysis difficult. To address these problems, we introduce deep representation learning (DRL) for SE. DRL can also improve the SE performance of DNN-based algorithms since DRL can obtain a discriminative speech representation, which can reduce the requirements for the learning machine to perform a task successfully. Specifically, we propose a Bayesian permutation training variational autoencoder (PVAE) to analyze underlying speech information for SE, which can represent and disentangle underlying noisy speech information in a vector form. The experimental results indicate that disentangled signal representations can also help current DNN-based SE algorithms achieve better SE performance. Additionally, based on this PVAE framework, we propose applying β-VAE and generative adversarial networks to improve PVAE's information disentanglement and signal restoration ability, respectively.

Resumé

I naturlige lyttemiljøer, vil talesignaler forvrænges let af forskellige akustiske interferenser, hvilket kan reducere talekvaliteten og forståeligheden; samtidigt gør det vanskeligt for mange talerelaterede app'er, såsom automatisk talegenkendelse (automatic speech recognition, ASR). Derfor er mange taleforbedringsalgoritmer (speech enhancement, SE) blevet udviklet i de sidste årtier. De fleste nuværende SE-algoritmer dog er vanskelige at fange underliggende taleinformationer (fx. fonem) i SE-processen. Dette får det til at være udfordrende at vide, hvilke specifikke informationer er tabt eller forstyrret i SE-processen, herved begrænser anvendelsen af taleforbedring. F.eks. nogle SE-algoritmer, der kan forbedre menneskelig lytning, kan ofte beskadige ASR-systemet.

Formålet med denne afhandling er at udvikle SE-algoritmer, gøre den muligt til at fange forskellige underliggende talerepræsentationer (informationer), og forbedre kvaliteten og forståeligheden af støjende tale. Denne undersøgelse starter med at introducere den skjulte Markov-model (hidden Markov model, HMM) i den ikke-negative matrixfaktorisering (Non-negative Matrix Factorization, NMF) model (NMF-HMM), fordi HMM er en bekvem måde til at finde underliggende taleinformationer for at bedre SE-ydeevne. Hovedideen er at anvende HMM til at fange den underliggende tidmæssige dynamiske taleinformationer i NMF-modellen. Derudover foreslås også en høj effektiv beregningsmetode for at sikre, at denne NMF-HMM-model kan opnå online SE hurtigt.

NMF-HMM kan fange de underliggende taleinformationer, men er det vanskeligt at forklare, hvilke detaljerede informationer er fået. Derudover kan NMF-HMM ikke repræsentere de underliggende informationer i en vektorform, hvilket er vanskeligt til at udføre informationsanalyse. For at løse disse problemer, introducerer vi deep representation learning (DRL) for SE. DRL kan også forbedre SE-ydeevnen, som er baseret på DNN algoritmer, da DRL kan forskelbehandle talerepræsentation, derved reducere kravene til læringsmaskinen til at udføre en opgave med succes. Vi specifikt foreslår at bruge Bayesian permutation training variational autoencoder (PVAE) til at analysere underliggende taleinformationer for SE, som kan repræsentere og adskille underliggende støjende taleinformationer i en vektorform. De eksperimentelle resultater angiver, at adskillede signalrepræsentationer også kan hjælpe nuværende SE-algoritmer (baseret på DNN) med at opnå bedre SE-ydeevne. Derudover baseret på denne PVAE-ramme, foreslår vi at anvende β-VAE og generative kontradiktoriske netværk

for at forbedre PVAE's henholdsvis informationsadskillelse og signalgendannelsesevne.

List of Papers

The main body of this thesis consists of the following papers:

[A] **Y. Xiang**, L. Shi, J.L Højvang, M.H Rasmussen, and M.G Christensen, "An NMF-HMM speech enhancement method based on kullback-leibler divergence" in *Proc. Interspeech,* 2020, pp. 2667–2671.

[B] **Y. Xiang**, L. Shi, J.L Højvang, M.H Rasmussen, and M.G Christensen, "A Speech Enhancement Algorithm based on Non-negative Hidden Markov Model and Kullback-Leibler Divergence." in *EURASIP Journal on Audio, Speech, and Music Processing,* 2022(1), pp.1-15.

[C] **Y. Xiang**, L. Shi, J.L Højvang, M.H Rasmussen, and M.G Christensen, "A Novel NMF-HMM Speech Enhancement Algorithm Based on Poisson Mixture Model" in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.,* 2021, pp. 721-725.

[D] **Y. Xiang**, J.L Højvang, M.H Rasmussen, and M.G Christensen, "A Bayesian Permutation training deep representation learning method for speech enhancement with variational autoencoder." in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.,* 2022, pp. 381-385.

[E] **Y. Xiang**, J.L Højvang, M.H Rasmussen, and M.G Christensen, "A deep representation learning speech enhancement method using β-VAE." in *Proc. Eur. Signal Process. Conf.* 2022, pp. 359-363.

[F] **Y. Xiang**, J.L Højvang, M.H Rasmussen, and M.G Christensen, "A Two-stage Deep Representation Learning-based Speech Enhancement Method Using Variational Autoencoder and Adversarial Learning." in *IEEE/ACM Trans. Audio,Speech, and Lang. Process,* 2022, (Submitted).

Contents

Preface

This thesis is submitted to the Technical Faculty of IT and Design at Aalborg University in partial fulfillment of the requirements for the Degree of Doctor of Philosophy. This thesis includes a summary and a collection of published papers. The summary part first introduces the research background of speech enhancement. Meanwhile, the summary also indicates this thesis's motivation, a key idea, and objective. After that, this part reviews some non-negative matrix factorization and deep representation learning models, which is the technical fundamentals of this research. Finally, the contributions and conclusions of this project are given at the end of the summary. In the second part, a collection of papers that have been published or in peer-reviewed are expected to be presented.

The work was carried out from September 2019 to September 2022 in the Audio Analysis Lab at the Department of Architecture, Design, and Media Technology (CREATE) at Aalborg University and Capturi A/S. This project was partly supported by Innovation Fund Denmark (Grant No.9065-00046). First and foremost, I would like to express my sincerest gratitude to my supervisor Prof. Mads Græsbøll Christensen, without whom this thesis would not have been possible. He taught me how to think about scientific problems and perform correct scientific research. He also trusted me and gave me the freedom to investigate what I was motivated for in the field of speech signal processing. I also would like to thank my co-supervisors, Dr. Morten Højfeldt Rasmussen and Dr. Jesper Lisby Højvang at Capturi A/S. I really appreciated their endless patience. They spent their precious time discussing research details with me. They also gave much valuable advice when helping me revise papers. Another person I would like to thank is my labmate, Dr. Liming Shi. I gained many fundamentals of Bayesian theory from him in the first year of my Ph.D. study, which generated many benefits for my following research. I also thank all my labmates at the Audio Analysis Lab and friends for making life pleasant during the past three years. Finally, I owe my greatest gratitude to my family for supporting me and standing with me through my ups and downs.

Yang Xiang
Aalborg University, December 5, 2022

Part I

Summary

1 Background

Speech signal plays an essential role in human communication. However, in our daily listening environment, speech signals are easily corrupted by various acoustic interference, which makes human listeners, especially people with hearing loss, feel difficult to understand the conversation. Meanwhile, the corrupted speech signals can also cause many related downstream speech applications, such as automatic speech recognition (ASR) [1], speaker identification [2], and speech translation, to become very difficult. As a result, speech enhancement (SE) techniques, which aims to remove background noise and improve speech quality and intelligibility in a noisy environment, have been developed during the past decades [3]. At present, due to the needs of practical applications, there are more and more requirements for SE [4]. For example, SE is required to help the online meeting system decrease the word error rate (WER) for accurate live captioning when transmitting high-quality speech signals in different complex listening environments [5, 6]. Therefore, SE has become a hot research topic.

Many efforts have been made to improve SE performance in the past decades. In an environment where the noise is additive, the most direct way for SE is the spectral subtraction algorithm (SSA) [7]. SSA subtracts an estimated short-term noise spectrum from the noisy signal spectrum to obtain the target clean speech signal spectrum. To get a better signal estimation, some statistic-based SE algorithms are developed, like some Wiener filtering-based [8, 9] strategies. Moreover, the minimum mean-square error (MMSE) spectral amplitude estimator [10] is another classic SE method. Based on this work [10], a log-MMSE spectral amplitude estimator [11] is proposed to suppress more residual noise. To increase the estimation accuracy of the clean speech spectrum, some noise estimation algorithms, such as minima controlled recursive averaging (MCRA) noise estimator [12] and improved MCRA (IMCRA) noise estimator [13], are combined with amplitude estimators, such as the optimally-modified log-spectral amplitude (OM-LSA) method [14] and Log-MMSE [15], for SE. The signal subspace method [16–18] is also a classical SE algorithm that analyzes the speech and noise subspace of the observed signal for the SE application.

However, most of these methods are difficult to obtain satisfactory SE performance in a non-stationary noisy environment and usually introduce musical

noise since they less consider applying prior speech and noise information for SE. To address this problem, data-driven SE algorithms are developed. The key idea of data-driven methods is that the signal models can be pre-trained using speech or noise data to obtain the prior information of signals before SE. In the online SE stage, the pre-trained signal model can be directly used to perform SE given noisy data. Classical data-driven SE algorithms include codebook-based methods [19–21], non-negative matrix factorization (NMF) methods [21–26], and the auto-regressive hidden Markov model (ARHMM) [27, 28].

Recently, with the advance in deep learning techniques [29, 30], deep neural networks (DNNs) have significantly promoted SE progress [31] and shown their great potential for SE [31–39]. Compared to classic SE algorithms, DNNs' benefits for SE are that they apply fewer assumptions [31, 32, 40] for signal analysis. So, some inaccurate assumptions can be avoided in DNN-based SE algorithms. In general, input features (representations), training targets, and learning machines are the three critical components for the DNN-based SE [31] methods. For the input features (representations), a more discriminative feature can place less demand on the learning machine to perform a task successfully [31]. On the contrary, a powerful learning machine places less demand on features [31]. Many features have been investigated for SE [31], such as pitch-based features [41], amplitude modulation spectrogram (AMS) [42], Mel-frequency cepstral coefficient (MFCC), and Gammatone frequency cepstral coefficient (GFCC) [43]. Although these features can represent different signal information [31] for SE, their disentanglement property is limited. They cannot disentangle different signal information, increasing the requirements for learning machines to generate high-quality speech. Thus, more powerful tools, such as the DRL model [44], should be considered to obtain better disentangled signal representations for SE.

For the training target, [32, 40] leverage a feedforward multilayer perceptron (MLP) to predict the log-power spectrum (LPS) of the clean speech using noisy LPS as network input. After that, the final enhanced speech signal is estimated using waveform reconstruction. The related experimental analysis indicates that this proposed DNN framework [32, 40] is able to obtain a higher short-time objective intelligibility (STOI) [45] score and perceptual evaluation of speech quality (PESQ) [46] than classic methods. In [33, 47], MLP is used to estimate the ideal ratio mask (IRM) and ideal binary mask (IBM) for SE. Unlike the direct spectrum mapping methods [32, 40], mask-based methods leverage DNNs to find the pre-defined time-frequency relationship among the clean speech, noise, and noisy speech for SE. These mask-based methods can also achieve satisfactory SE performance. Moreover, DNN can also be used to predict various speech present probabilities [48] for the SE purpose.

In general, most DNN-based SE algorithms [32, 33, 40, 47, 48] only analyze the magnitude spectrum of signals and ignore the phase information because phase is not so essential for enhanced speech [49]. However, recent research [39, 50] indicates that accurate phase estimation is necessary to generate high-quality enhanced speech. Therefore, many DNN-based SE algorithms

that consider the phase information target are proposed. These methods include some phase-sensitive filter algorithms [51] and complex ideal ratio mask (cIRM) estimation algorithms that jointly estimate real and imaginary components of clean speech [52, 53]. Another way to address the phase estimation problem is to leverage DNNs to perform SE in the time domain, which directly estimates the waveform of clean speech. The end-to-end SE strategies [35, 54] can also achieve excellent SE performance.

Learning machines is another key component of the DNN-based SE algorithm. Many different learning machines have been investigated for the SE application. Convolutional neural networks (CNNs) [55] can discover robust and localized low-dimensional patterns [55], so CNN has the potential to achieve better SE performance than MLP. In [56, 57], CNN is applied to perform mask and spectrum estimation for SE. In [58], a fully convolutional encoder-decoder network (FCED) is used to perform spectrum mapping SE. Compared to MLP, CNN's other benefit for SE is that its number of parameters is smaller, so CNN is easier to apply to some embedded devices [58]. The speech signal is related to the temporal process, so deep recurrent neural networks (RNNs) [59] can generate benefits for the SE. In [60], RNN is used to conduct denoising tasks and performs better than MLP. A more advanced RNN structure, long short-term memory networks (LSTMs) [61], is utilized to predict multi-target for SE [34]. Moreover, LSTM can also be used to improve the ability of speaker generalization [62]. To make the best use of the advantage of CNN and RNN, convolutional recurrent networks (CRNs) [63, 64] are proposed to conduct the SE, which can benefit from CNN's feature extraction and RNN's temporal modeling ability. Experimental results in [63, 64] indicate that CRNs can obtain higher STOI and PESQ scores than single CNNs and RNNs models. The speech signals are the complex values in the time-frequency analysis using a short-time Fourier transform (STFT), so deep complex networks (DCNs) [65, 66] are also investigated to perform SE, which can directly process the complex-valued spectrum and avoid the phase estimation problem. Furthermore, combining CRNs and DCNs (deep complex convolution recurrent network (DCCRN)) [67] can achieve more excellent SE performance. Moreover, some generative models, such as variational autoencoder (VAE) [68] and generative adversarial networks (GANs) [69, 70], are also investigated for SE application and have shown their potential in SE. For instance, [71] and [72] use GANs to estimate speech spectrum and waveform, respectively. Both of them obtain satisfactory SE performance. In addition, many semi-supervised and unsupervised SE algorithms [36, 73–75] are also investigated in recent works to improve DNN's generalization ability.

To sum up, at present, many SE algorithms have been proposed and achieved satisfactory SE performance. Some of these algorithms [67, 76] can also meet practical applications [77]. However, with the progress of the times, there are more and more requirements for the SE technique. For instance, for some online meeting applications, SE needs to improve human hearing and reduce WER simultaneously. As a result, some new strategies are supposed to

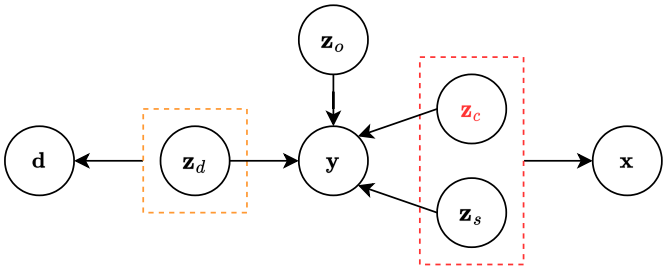
be considered to improve the performance of the current SE methods.

The rest of this section is organized as follows. First, we will explain the motivation of this study in subsection 1.1. Then, the fundamental ideas of using representation learning for SE are expected to be illustrated in subsection 1.2. After that, we will show the organization of this thesis in subsection 1.3.

1.1 Motivation

At present, although many SE algorithms have been proposed [3, 31], most of these methods only analyze the limited information for SE. They do not try to capture more underlying information (e.g., phoneme or text information) to improve SE performance. For example, most of the present DNN-based SE methods [31–33, 35–40] focus on optimizing the training targets and learning machines. These DNN-based methods use different learning machines [31] to predict different pre-defined targets (e.g., various masks [33], speech spectrum [40], and speech present probability [48]). Although direct prediction of pre-defined targets can prevent inaccurate signal assumptions [40], the lack of underlying information analysis in SE can cause it challenging to understand signal distortion. We cannot know the relationship between signal distortion and information loss, which limits the SE application in some downstream tasks. For instance, some SE methods aiming to improve human listening can damage the ASR system [5, 6]. The enhanced signals have a higher WER than noisy speech in the ASR system. In general, this unsatisfactory ASR performance is caused by the distortions of the enhanced signal. Without the analysis of underlying information (e.g., phoneme or text information), it is difficult to know how the ASR-needed information is lost in signal distortions. Similarly, the same problem also exists in traditional SE algorithms [3, 16–21]. Basically, the classic SE algorithms [3, 16–21] utilize STFT to analyze the T-F representations of the speech signal or directly analyze the time-domain waveform. However, analyzing T-F and waveform representations is ineffective in capturing and disentangling different underlying speech information because some information, such as phoneme and content information, is less related to the T-F and waveform representations. For example, different speakers are able to say the same sentence. T-F and waveform representations are difficult to disentangle and represent the speaker and content information and perform the related analysis of disentangled information.

To mitigate the above problems, this thesis aims to develop a SE framework that has the potential to capture underlying speech information (representation) and can improve speech quality and intelligibility when performing SE. Moreover, a good signal representation is also essential to improve DNN's performance [44] since a discriminative speech representation can place less demand on the learning machine to perform a task successfully [31].



Signals

y: noisy signal

x: speech

d: noise

latent variables

$\mathbf{z}_c$: content-related information

$\mathbf{z}_s$: style-related information

$\mathbf{z}_d$: noise information

$\mathbf{z}_o$: other information

Fig. 1: Graphic illustration of information of the noisy speech

.

1.2 Key Idea and Objectives

Fig.1 shows a simple graphic illustration of what information of a noisy speech could be included and analyzed. In Fig.1, we can find that a noisy signal could contain content-related information, style-related information, noise information, and other information. The different information can be represented by the different vector forms ($\mathbf{z}_c$, $\mathbf{z}_s$, $\mathbf{z}_d$, and $\mathbf{z}_o$). More specifically, content-related information involves the understanding of a word or sentence. Each different speaker can use the same content information to express the same. Content information is essential for the ASR application. Style-related information usually represents the speakers, which decides the style of a speech signal. Noise information determines what types of noise are included in this noisy signal. Other information means some other possible underlying information (e.g., reverberation or multi-speaker information) that may be included in the noisy speech, which is adjustable based on the practical application. Fig.1 also indicates that all the different information can decide the noisy signal $\mathbf{y}$. Meanwhile, each independent information can also decide different signals. For example, noise information can determine the noise signal $\mathbf{d}$. The combination of content and style information can decide the speech signal $\mathbf{x}$. Note, Fig.1 just shows a basic framework for information analysis of the noisy signal. In practical application, more information analysis could be considered in Fig.1. For instance, emotional information could also be included in the speech signal part.

To capture and analyze various information in a speech signal, we can con-

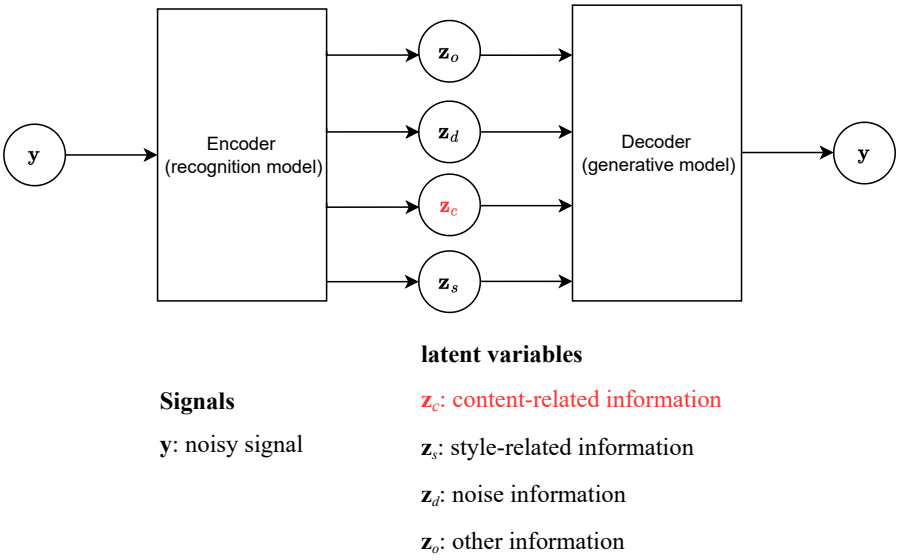
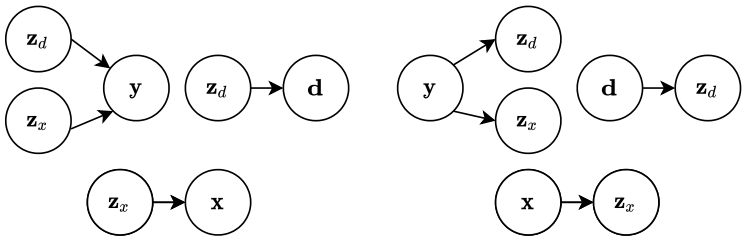


Fig. 2: Graphic illustration of representation analysis for deep representation learning model.

sider leveraging the hidden Markov model (HMM) [27] or the deep representation learning (DRL) model [44]. In general, HMM can capture the speech temporal dynamic information, and each HMM's state can intend to find specific speech information (e.g., a phoneme). HMM has been widely applied in the ASR area [1]. HMM is easily combined with some traditional SE algorithms like NMF. However, the drawbacks of using HMM to find speech information are that HMM is difficult to represent various underlying information efficiently (e.g., using a vector to represent the information), which causes it challenging to perform detailed information analysis. In addition, HMM is also difficult to disentangle different underlying information, which may lead to inaccurate information analysis. To mitigate these issues, we consider using DRL for information analysis. DRL can represent underlying information in a vector form and disentangle different speech information [78–80], effectively analyzing signals. At present, the DRL model has been widely used in speech conversion and synthesis domains [81, 82]. Fig.2 shows how to use DRL to perform signal analysis. In general, the DRL model contains an encoder and decoder [82]. The encoder also named the recognition model, is responsible for finding and disentangling various underlying information. Finally, the encoder can disentangle multiple pieces of information into different vector forms (latent variables). We can perform related information analysis using these vectors. The decoder also named the generative model, is responsible for signal recovery using previously analyzed vectors (latent variables). The decoder can use all disentangled latent variables to generate a signal, as shown in Fig.2. Alternatively, the decoder can just use some of them to recover the signal, depending



(a) Generative model (b) Recognition model

Fig. 3: Graphic illustration of simplified signal analysis model [83].

on the practical application. For example, we only need the content and style information to reconstruct a clean speech signal for SE, as shown in Fig.1.

This thesis first aims to leverage HMM to capture the underlying speech information because HMM is easily combined with some existing signal models like NMF [21–26] for SE. However, due to the limitation in HMM, we will focus on using the DRL model for signal analysis. This thesis attempts to leverage DRL for SE, which is a novel aspect of signal analysis in SE. Therefore, as a preliminary study of investigating the representation learning for SE, we simplify the signal model in Fig.1 as the model in Fig.3 [83]. In this study, we are based on the simplified signal model (Fig.3) to conduct the related signal analysis. Comparing the simplified model (Fig.3) to the original model (Fig.1), we can find that the simplified model only considers speech information $\mathbf{z}_x$ and noise information $\mathbf{z}_d$. $\mathbf{z}_x$ is the combination of $\mathbf{z}_c$ and $\mathbf{z}_s$. The simplified model does not disentangle more detailed information and treats $\mathbf{z}_c$ and $\mathbf{z}_s$ as a whole information $\mathbf{z}_x$. The reason for using a simplified model is that it is more convenient to validate the correctness of the proposed framework. Note that although we verify our framework in a simplified signal model, the proposed framework could be a more general strategy that can be easily extended to analyze the more complex signal information. Moreover, another benefit of using DRL for SE is that DRL can also help DNN-based SE algorithms improve SE performance since DRL can learn a discriminative speech representation [44], and a discriminative speech representation can reduce the requirements for the learning machine to conduct the SE task successfully [31].

1.3 Organization of Thesis

We structure the summary of the thesis as follows.

In this thesis, we first consider using the combination of NMF and HMM to capture the underlying speech information for SE, so we will explain some fundamental mathematical theories of NMF and HMM in Section 2. In addition, we focus on leveraging DRL models to perform SE, so we will also introduce some advanced DRL models, such as variational autoencoder (VAE) [68], β-

VAE [84, 85], and generative adversarial networks (GANs) [69, 70], in section 3. After that, we will indicate the contributions of this study and draw conclusions in section 4. Finally, since this is a preliminary work of investigating the application of representation learning in SE, we will indicate some possibly meaningful research directions in section 5.

2 Non-negative Matrix Factorization Model

This section first explains the basic NMF algorithm with Kullback-Leibler (KL) divergence. After that, this section shows the NMF's application in SE. This thesis considers applying HMM in NMF, so this section also shows the basic framework of how we combine HMM and NMF.

2.1 Basic Non-negative Matrix Factorization with Kullback-Leibler Divergence

In general, NMF is a group of algorithms in multivariate analysis and linear algebra, which can factorize a matrix $\mathbf{V}$ into two matrices $\mathbf{W}$ and $\mathbf{H}$. All of the three matrices $\mathbf{V}$, $\mathbf{W}$, and $\mathbf{H}$ have no negative elements. The NMF can make the matrix data easier to inspect and analyze. At present, NMF has been widely applied in computer vision [86] and audio signal processing [24, 87–89].

Consider a clean speech signal $x(t)$, t is the time index, and its short-time Fourier transform (STFT) coefficients are $X(f,n)$. Here, $n \in [1,N]$ and $f \in [1,F]$ represent the time frame and frequency bin indices, respectively. Gathering the F frequency bins and N time frames, the clean speech magnitude spectrum matrice can be defined as $\mathbf{X}_N$, where $\mathbf{X}_N = [\mathbf{x}_1, \cdots, \mathbf{x}_n, \cdots, \mathbf{x}_N]$ and $\mathbf{x}_n = [|X(1,n)|, \cdots, |X(f,n)|, \cdots, |X(F,n)|]^T$ [90]. There are no negative elements in $\mathbf{X}_N$, so many measures [90], such as KL divergence [23], Itakura-Saito (IS) divergence [91], β divergence [92], and Euclidian distance [93], can be used as the loss function to perform NMF for $\mathbf{X}_N$. In this thesis, we choose KL divergence as the loss function to conduct NMF because the best SE performance can be achieved using the KL divergence as the NMF loss function [94] with the magnitude spectrum [90, 94] compared with other NMF loss functions. In addition, the KL divergence-based NMF can derive to the multiplicative update (MU) rule for the NMF parameters update, which is an efficient way for the parameters calculation.

Based on the definition, the KL divergence measure between the two matrices $\mathbf{D}$ and $\hat{\mathbf{D}}$ can be represented as [90]

$$D_{KL}(\mathbf{D}||\hat{\mathbf{D}}) = \sum_{i,j}(d_{i,j}\log(d_{i,j}/\hat{d}_{i,j}) - d_{i,j} + \hat{d}_{i,j}), \tag{1}$$

where $d_{i,j}$ and $\hat{d}_{i,j}$ are the elements from the i^{th} row and j^{th} column of in matrices $\mathbf{D}$ and $\hat{\mathbf{D}}$, respectively. $D_{KL}(||)$ denotes the KL divergence. Based on

the KL measure, the $\mathbf{X}_N$ can be factorized as a basis matrix $\overline{\mathbf{W}}$ and a activation matrix $\overline{\mathbf{H}}$. The NMF loss function with KL divergence can be represented as [90]

$$(\overline{\mathbf{W}}, \overline{\mathbf{H}}) = \underset{\overline{\mathbf{W}}, \overline{\mathbf{H}}}{\arg\min}\, D_{KL}(\mathbf{X}_N || \overline{\mathbf{W}} \times \overline{\mathbf{H}}). \tag{2}$$

In [23, 90], it has been shown that $\overline{\mathbf{W}}$ and $\overline{\mathbf{H}}$ can be estimated iteratively using the MU rules:

$$\overline{\mathbf{W}} \leftarrow \overline{\mathbf{W}} \odot \frac{\dfrac{\mathbf{X}_N}{\overline{\mathbf{W}} \times \overline{\mathbf{H}}} \overline{\mathbf{H}}^T}{\mathbf{1}\overline{\mathbf{H}}^T}, \tag{3}$$

$$\overline{\mathbf{H}} \leftarrow \overline{\mathbf{H}} \odot \frac{\overline{\mathbf{W}}^T \dfrac{\mathbf{X}_N}{\overline{\mathbf{W}} \times \overline{\mathbf{H}}}}{\overline{\mathbf{W}}^T \mathbf{1}}, \tag{4}$$

where $\odot$ and all divisions represent element-wise multiplication and division operations, respectively, and $\mathbf{1}$ is a matrix of ones that has the same dimension as matrix $\mathbf{X}_N$ [23, 90].

In [90, 95], it shows that MU rule in (3) and (4) using gradient descent derivation can be also derived from a statistical analysis aspect. Specifically, the KL divergence-based NMF can also be obtained from the hierarchical statistical model [90, 95]

$$\mathbf{X}_N = \sum_{k=1}^{K} \mathbf{C}(k), \tag{5}$$

$$c_{f,n}(k) \sim \mathcal{PO}(c_{f,n}(k); \overline{W}_{f,k}\overline{H}_{k,n}), \tag{6}$$

where $\mathcal{PO}(y;\lambda) = \dfrac{\lambda^y e^{-\lambda}}{\Gamma(y+1)}$ denotes the Poisson distribution. Additionally, $\Gamma(y+1) = y!$ is the gamma function for positive integer y, the K is the number of basis vectors, $\mathbf{C}(k)$ denotes the latent matrix, and $c_{f,n}(k)$ is the element of matrix $\mathbf{C}(k)$ in the f^{th} row and n^{th} column [90, 95]. $\overline{W}_{k,n}$ and $\overline{H}_{k,n}$ are the elements of the basis $\overline{\mathbf{W}}$ and activation matrices $\overline{\mathbf{H}}$, respectively.

In (6), we assume that $c_{f,n}(k)$ is a Poisson distribution, which means that $c_{f,n}(k)$ can be only applied for the discrete variables [90, 95]. However, the practical signals are the continuous variables, so we leverage the gamma function to replace the factorial calculation [90, 95] in the original Poisson distribution. Moreover, papers [90, 95] indicate that the iterative update of the parameters $\overline{\mathbf{H}}$ and $\overline{\mathbf{W}}$ using MU rule in (3) and (4) is identical to applying the EM algorithm to perform parameters update in(5) and (6).

2.2 NMF-based Speech Enhancement

This subsection will illustrate how to use the basic NMF to conduct SE in an environment with additive noise. In general, the additive noisy signal model can be written as [90]

$$y(t) = x(t) + m(t), \tag{7}$$

where $y(t)$ and $m(t)$ are the noisy signal and noise, respectively. The STFT coefficients of $y(t)$ can be represented as [90]

$$Y(f,n) = X(f,n) + M(f,n), \tag{8}$$

where $Y(f,n)$ and $M(f,n)$ represent the STFT spectrums of $y(t)$ and $m(t)$, respectively. The magnitude spectrum matrices [90] of noisy signal and noise are defined as $\mathbf{Y}_N$ and $\mathbf{M}_N$, which is similar to the definition of $\mathbf{X}_N$. Moreover, the $\mathbf{y}_n$ and $\mathbf{m}_n$ also have the similar definition to $\mathbf{x}_n$. To simplify the model analysis, we assume that $\mathbf{Y}_N = \mathbf{X}_N + \mathbf{M}_N$.

In general, there are two stages for the NMF-based SE [90]: offline training and online enhancement. In the first offline stage [90], we leverage the clean speech and noise databases to pre-train the clean speech basis matrix $\overline{\mathbf{W}}$ and noise basis matrix $\ddot{\mathbf{W}}$, respectively. In the following part, the overbar $(\bar{\cdot})$ and double dots $(\ddot{\cdot})$ are used to represent the clean speech and noise for the matrix expression, respectively. The training loss function and related parameters update are shown in (2), (3), and (4), respectively [90]. In the online stage [90], the noisy speech basis matrix $\mathbf{W}$ is built by connecting the pre-trained noise and speech basis matrices, which can be expressed as $\mathbf{W} = [\overline{\mathbf{W}}, \ddot{\mathbf{W}}]$. The activation matrix $\mathbf{H}$ of the noisy speech can be estimated iteratively by replacing $\mathbf{X}_N$, $\overline{\mathbf{W}}$, and $\overline{\mathbf{H}}$ in (4) with $\mathbf{Y}_N$, $\mathbf{W}$, and $\mathbf{H}$, respectively. Finally, we are able to acquire enhanced speech by various related post-processing algorithms [24, 87–89]. One of the general post-processing strategies for SE is to apply the Wiener-filter-like spectral gain $\mathbf{g}_n$ function [90] to estimate a clean speech magnitude spectrum:

$$\mathbf{g}_n = \frac{\overline{\mathbf{W}}\,\overline{\mathbf{h}}_n}{\overline{\mathbf{W}}\,\overline{\mathbf{h}}_n + \ddot{\mathbf{W}}\,\ddot{\mathbf{h}}_n}, \tag{9}$$

$$\begin{aligned}\mathbf{h}_n &= \left[\overline{\mathbf{h}}_n^T, \ddot{\mathbf{h}}_n^T\right]^T \\ &= \arg\min_{\mathbf{h}_n} D_{KL}(\mathbf{y}_n || \mathbf{W}\mathbf{h}_n).\end{aligned} \tag{10}$$

To solve (10), we can apply (4) to perform related calculations. Finally, the enhanced signal can be estimated by

$$\hat{\mathbf{x}}_n = \mathbf{y}_n \odot \mathbf{g}_n. \tag{11}$$

Although NMF is an effective strategy for speech signal analysis, its ability in capturing underlying speech information is limited. As a result, many other

algorithms (e.g., HMM and DNN [96]) are usually combined with NMF to achieve a better SE performance.

2.3 Combination of HMM and NMF

In this subsection, we will show how to apply HMM to capture the underlying temporal dynamic information of the signal in NMF. Based on the conditional independence property of the standard HMM [90, 97], we can obtain such a likelihood function [90]

$$p(\mathbf{X}_N;\mathbf{\Phi}) = \sum_{\overline{\mathbf{s}}_\mathbf{N}} \prod_{n=1}^{N} p(\mathbf{x}_n|\overline{s}_n)p(\overline{s}_n|\overline{s}_{s-1}), \tag{12}$$

where $\overline{\mathbf{s}}_\mathbf{N} = [\overline{s}_1, \cdots, \overline{s}_n, \cdots, \overline{s}_N]^T$ represents the states collection, and $\overline{s}_n \in \{1, 2, \cdots, \overline{J}\}$ is the state at the n^{th} frame [90]. The $\overline{J}$ is the total number of states [90]. Here, the probability function $p(\overline{s}_n|\overline{s}_{n-1})$ represents the state transition probability from state $\overline{s}_{n-1}$ to $\overline{s}_n$ using $p(\overline{s}_1|\overline{s}_0)$ as the initial state probability [90]. $p(\mathbf{x}_n|\overline{s}_n)$ is the state-conditioned likelihood function [90]. Moreover, $\mathbf{\Phi}$ is defined as the collection of estimation parameters [90]. In (12), we need to obtain $p(\overline{s}_n|\overline{s}_{n-1})$ and $p(\mathbf{x}_n|\overline{s}_n)$ to calculate the whole likelihood function $p(\mathbf{X}_N;\mathbf{\Phi})$ [90].

For the $p(\overline{s}_n|\overline{s}_{n-1})$, it can be modeled using a first-order Markov chain, which can be represented as [90]

$$p(\overline{s}_n|\overline{s}_{n-1}) = \prod_{i=1}^{\overline{J}} \prod_{j=1}^{\overline{J}} \overline{A}_{i,j}^{l(\overline{s}_n=j,\overline{s}_{n-1}=i)}, \tag{13}$$

$$p(\overline{s}_1|\overline{s}_0) = p(\overline{s}_1) = \prod_{j=1}^{\overline{J}} \overline{\pi}_j^{l(\overline{s}_1=j)}, \tag{14}$$

where $l(\cdot)$ is an indicator function [90], which can be written as

$$l(y) = \begin{cases} 1, & \text{if logic expression } y \text{ is true} \\ 0, & \text{if logic expression } y \text{ is false} \end{cases}. \tag{15}$$

In (13) and (14), $\overline{A}_{i,j}$ represents the transition probability from state i to state j. $\overline{\pi}_j$ is the initial probability for the first frame's state $\overline{x}_1$ being state j. Gathering all the initial and transition probabilities, they can be written as the vector and matrix forms, $\overline{\boldsymbol{\pi}} = [\overline{\pi}_1, \cdots, \overline{\pi}_j, \cdots, \overline{\pi}_{\overline{J}}]^T$ and $\overline{\mathbf{A}}$ with $\overline{A}_{i,j}$ being the element at the i^{th} row and j^{th} column. As a result, we can express the related modeling parameters of state transition probability as $\mathbf{\Phi}_{\text{hmm}} = \{\overline{\mathbf{A}}, \overline{\boldsymbol{\pi}}, \overline{J}\}$, where $\overline{J}$ is predefined, and $\overline{\mathbf{A}}$ and $\overline{\boldsymbol{\pi}}$ can be updated using the EM algorithm [90].

For the state-conditioned likelihood function, we can apply (5) and (6) to build it, which can be represented as [25]

$$\mathbf{x}_n = \sum_{k=1}^{\overline{K}} \overline{\mathbf{c}}_n(k), \tag{16}$$

$$p(\overline{\mathbf{c}}_n(k)|\overline{s}_n) = \prod_{f=1}^{F} \mathcal{PO}(\overline{c}_{f,n}(k); \overline{W}_{f,k}^{\overline{s}_n} \overline{H}_{k,n}^{\overline{s}_n}), \tag{17}$$

where $\overline{K}$ is the basis vectors number, $\overline{\mathbf{c}}_n(k)$ contains the hidden variables, and $\overline{W}_{k,n}^{\overline{s}_n}$ and $\overline{H}_{k,n}^{\overline{s}_n}$ correspond to the elements of the basis and activation matrices, respectively [90]. Here, we can write $\overline{\mathbf{c}}_n$ as: $\overline{\mathbf{c}}_n = [\overline{\mathbf{c}}_n(1)^T, \overline{\mathbf{c}}_n(2)^T, \cdots, \overline{\mathbf{c}}_n(\overline{K})^T]^T$. Finally, the $p(\mathbf{x}_n|\overline{s}_n)$ can be expressed as

$$\begin{aligned} p(\mathbf{x}_n|\overline{s}_n) &= \int p(\mathbf{x}_n|\overline{\mathbf{c}}_n) p(\overline{\mathbf{c}}_n|\overline{s}_n)\, d\overline{\mathbf{c}}_n \\ &= \prod_{f=1}^{F} \mathcal{PO}(|X(f,n)|; \sum_{k=1}^{\overline{K}} \overline{W}_{f,k}^{\overline{s}_n} \overline{H}_{k,n}^{\overline{s}_n}), \end{aligned} \tag{18}$$

where the superposition property of the Poisson random variable [90, 95] is leveraged. Collect the NMF parameters $\{\overline{W}_{f,k}^{\overline{s}_n}\}$ and $\{\overline{H}_{k,n}^{\overline{s}_n}\}$ and express them as the matrix forms, we can obtain $\{\overline{\mathbf{W}}^j\}$ and $\{\overline{\mathbf{H}}^j\}$. Here, we can find that different from the traditional NMF that leverages only one basis matrix to model signal, the signal can be modeled by $\overline{J}$ basis matrices using the combination of NMF and HMM. Moreover, each basis matrix has the potential to find a specific signal information (e.g., a phoneme or emotion). The related modeling parameters of $p(\mathbf{x}_n|\overline{s}_n)$ can be written as $\mathbf{\Phi}_{\text{like}} = \{\{\overline{\mathbf{W}}^j\}, \{\overline{\mathbf{H}}^j\}, \overline{K}, \overline{J}\}$, where $\overline{J}$ and $\overline{K}$ are pre-defined and $\{\overline{\mathbf{W}}^j\}$ and $\{\overline{\mathbf{H}}^j\}$ can be obtained by EM algorithm.

To sum up, there are four types of parameters in $\mathbf{\Phi}$: $\mathbf{\Phi} = \mathbf{\Phi}_{\text{hmm}} \cup \mathbf{\Phi}_{\text{like}}$. They are the pre-defined parameters: $\overline{J}$ and $\overline{K}$, NMF parameters: $\{\overline{\mathbf{W}}^j\}$ and $\{\overline{\mathbf{H}}^j\}$, transition matrix: $\overline{\mathbf{A}}$, and initial state probabilities: $\overline{\boldsymbol{\pi}}$. Apart from the pre-defined parameters, the other three types of parameters can be estimated using EM algorithms [90, 95].

3 Deep Representation Learning and Generative Models

Recently, many DRL and generative models, such as variational autoencoder (VAE) [68, 98], generative adversarial networks (GAN) [69, 70], Bayesian wavenet [99, 100], and diffusion probabilistic models [101–103], have been proposed for data analysis. These models have shown great potential in data learning and data generation. This section briefly introduces the VAE, β-VAE [84, 85], and GAN models.

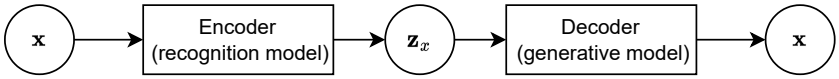


Fig. 4: Graphic illustration for VAE

3.1 Variational Autoencoder (VAE)

The VAE is a probabilistic generative model [68]. Based on the variational Bayes [68], VAE provides a probabilistic generative process between the observed data and the corresponding latent variables. VAE also defines a principled strategy to jointly learn latent variables, recognition, and generative models [68, 83]. Generative and recognition models can be jointly trained by maximizing the evidence lower bound (ELBO) or minimizing the KL divergence between their true joint distribution [83] and the corresponding estimation [68] using the stochastic gradient descent (SGD) or Adagrad [104] algorithm. Fig.4 shows a general VAE structure. Here, $\mathbf{x}$ represents the clean speech magnitude spectrum. For simplicity, we remove the time frame indices n. $\mathbf{z}_x$ is the latent variables of $\mathbf{x}$, where $\mathbf{z}_x \in \mathbb{R}^L$ and L is the dimensions of vectors $\mathbf{z}_x$.

The VAE's optimization process can be seen as minimizing the KL divergence between real joint probability distribution $p(\mathbf{x}, \mathbf{z}_x)$ and corresponding estimation $q(\mathbf{x}, \mathbf{z}_x)$, which can be expressed as follow [83]:

$$\begin{aligned} &D_{KL}\left(p(\mathbf{x}, \mathbf{z}_x)||q(\mathbf{x}, \mathbf{z}_x)\right) = \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\log p(\mathbf{x})\right] \\ &+ \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[D_{KL}\left(p(\mathbf{z}_x|\mathbf{x})\right)||q(\mathbf{x}, \mathbf{z}_x))]. \end{aligned} \tag{19}$$

In (19), the term $\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\log p(\mathbf{x})\right]$ can be seen as a constant, so minimizing their KL divergence is equal to minimizing [83]

$$\begin{aligned} &\mathcal{L}(\theta_x, \varphi_x; \mathbf{x}) \\ &\quad = \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[D_{KL}\left(p(\mathbf{z}_x|\mathbf{x})\right)||q(\mathbf{x}, \mathbf{z}_x))] \\ &\quad = \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[D_{KL}\left(p(\mathbf{z}_x|\mathbf{x})\right)||q(\mathbf{z}_x))\right] \\ &\qquad - \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{x})}\left[\log q(\mathbf{x}|\mathbf{z}_x)\right]\right], \end{aligned} \tag{20}$$

where θ_x and φ_x are the encoder and decoder parameters, respectively, which is applied to perform the related probability estimation. Here, $-\mathcal{L}(\theta_x, \varphi_x; \mathbf{x})$ can be also seen as the ELBO [83]. This ELBO can be written as

$$\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[\log q(\mathbf{x})] \geq -\mathcal{L}(\theta_x, \varphi_x; \mathbf{x}). \tag{21}$$

Minimizing $\mathcal{L}(\theta_x, \varphi_x; \mathbf{x})$ is equal to maximizing this ELBO. The whole VAE framework can be trained using (20) as a loss function.

To calculate (20), we need to estimate posterior distribution $p(\mathbf{z}_x|\mathbf{x})$, $p(\mathbf{z}_x)$, and prior distribution $q(\mathbf{x}|\mathbf{z}_x)$ using parameter θ_x and φ_x. In general, for speech signal analysis [68, 98], $p(\mathbf{z}_x)$ can be predefined as a centered isotropic multivariate Gaussian: $\mathbf{z}_x \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, which can be written as

$$q(\mathbf{z}_x) = \mathcal{N}(\mathbf{z}_x; \mathbf{0}, \mathbf{I}), \tag{22}$$

where $\mathbf{I}$ is the identity matrix. The $p(\mathbf{z}_x|\mathbf{x})$ and $q(\mathbf{x}|\mathbf{z}_x)$ can be estimated using encoder and decoder, respectively. For the simplicity of calculation, the posterior and prior distributions are usually assumed to have multivariate normal distributions with diagonal covariance [68], which has been widely applied in many VAE-based SE algorithms [98, 105–108]. Therefore, $p(\mathbf{z}_x|\mathbf{x})$ and $q(\mathbf{x}|\mathbf{z}_x)$ can be expressed as

$$p(\mathbf{z}_x|\mathbf{x}) = \mathcal{N}\left(\mathbf{z}_x; \mu_{\theta_x}(\mathbf{x}), \sigma^2_{\theta_x}(\mathbf{x})\mathbf{I}\right), \tag{23}$$

$$q(\mathbf{x}|\mathbf{z}_x) = \mathcal{N}\left(\mathbf{x}; \mu_{\varphi_x}(\mathbf{z}_x), \sigma^2_{\varphi_x}(\mathbf{z}_x)\mathbf{I}\right), \tag{24}$$

where $\mu_{\theta_x}(\mathbf{x})$ and $\sigma^2_{\theta_x}(\mathbf{x})$ are the mean and covariance of posterior distribution $p(\mathbf{z}_x|\mathbf{x})$, respectively, which can be estimated using encoder with $\mathbf{x}$ as input. Similarly, $\mu_{\varphi_x}(\mathbf{z}_x)$ and $\sigma^2_{\varphi_x}(\mathbf{z}_x)$ are the mean and covariance of prior distribution $q(\mathbf{x}|\mathbf{z}_x)$, respectively. They can be estimated using a decoder with latent variable $\mathbf{z}_x$ as input. To obtain $\mathbf{z}_x$ from posterior distribution $p(\mathbf{z}_x|\mathbf{x})$, we need to apply the reparameterization trick, which can be expressed as [68]

$$\mathbf{z}_x = \mu_{\theta_x}(\mathbf{x}) + \sigma^2_{\theta_x}(\mathbf{x}) \odot \epsilon, \tag{25}$$

where ϵ is a random noise variable, which satisfies $\epsilon \in \mathbb{R}^L$ and $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. This reparameterization trick also ensures that the parameter update in VAE is differentiable.

At present, VAE has been widely used in the SE task [98, 105–109]. However, most of these algorithms only apply VAE to learn the clean speech representation, and they do not attempt to disentangle clean speech representation with other noise representations, which leads to inaccurate speech estimation. Additionally, entangled representation also causes these methods to apply a linear NMF to model noise. As a result, the noise modeling ability of these methods is not satisfactory compared to these non-linear DNN-based methods [68]. Therefore, an effective disentanglement strategy is essential for these VAE-based SE methods.

3.2 β-VAE

β-VAE [84, 85] is a developed DRL model based on the VAE. β-VAE adds an adjustable hyperparameter β in VAE's KL divergence term. Thus, the optimization target in (20) can be rewritten as

$$\begin{aligned}
&\mathcal{L}(\theta_x, \varphi_x; \mathbf{x}) \\
&\quad = \beta \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[D_{KL}\left(p(\mathbf{z}_x|\mathbf{x})\right)||q(\mathbf{z}_x))\right] \\
&\qquad - \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{x})}\left[\log q(\mathbf{x}|\mathbf{z}_x)\right]\right].
\end{aligned} \tag{26}$$

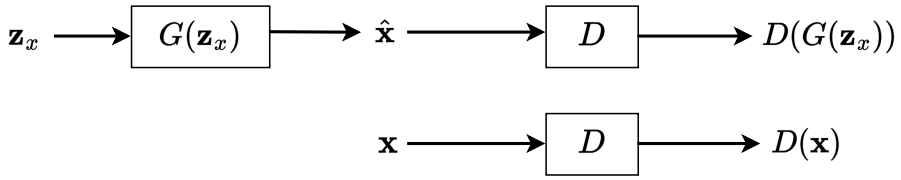


Fig. 5: Graphic illustration for GAN

(26) is the loss function of β-VAE. β-VAE aims to help the original VAE [68] to obtain a better signal representation. In β-VAE, the signal representation can be disentangled if the data has more than one underlying factor of variation [84]. Generally, $\beta > 1$ can lead to more disentangled latent representations [84]. A better and more disentangled representation can be obtained by setting a higher value of β. However, the β-VAE's drawback is that it has a trade-off problem. More specifically, a better disentanglement performance in latent space can usually result in a worse signal reconstruction performance [84].

3.3 Generative Adversarial Network (GAN)

GAN [69] is a very powerful probabilistic generative model that can generate an expected sample from a probability distribution using DNNs. The DNN's input can be a random vector. A GAN [69] contains a generator network and a discriminator network. The generator network $G(\cdot)$ can map a random variable $\mathbf{z}_x$ ($\mathbf{z}_x \sim q(\mathbf{z}_x)$) to a expected sample. The purpose of the generator network is to imitate real data to cheat the discriminator [36]. Typically, there are no rigid restrictions for the distribution $q(\mathbf{z}_x)$ [70]. The task of discriminator network $D(\cdot)$ is to determine whether $\mathbf{x}$ is an actual training sample or it is generated by the generator model through $\hat{\mathbf{x}} = G(\mathbf{z}_x)$. This means that the discriminator network needs to distinguish these generated data from the true data. As a result, there is an adversarial learning process [69] between $G(\cdot)$ and $D(\cdot)$ [36]. Fig.5 shows the GAN's framework.

The GAN can be derived by a general variational divergence estimation approach [70]. In [70], it has been shown that GAN can be trained by any f-divergences. In general, we can choose Jensen–Shannon (JS) divergence to optimize GAN because JS divergence has an upper and lower bound, which is easier to converge. If we use JS divergence, GANs can be optimized by the minimax of the loss function [69]:

$$\begin{aligned} &\min_G \max_D \mathcal{L}_{gan}(G, D) = \\ &\mathbb{E}_{\mathbf{x}\sim q_{data}(\mathbf{x})}[\log(D(\mathbf{x}))] + \mathbb{E}_{\mathbf{z}_x \sim q(\mathbf{z}_x)}[\log(1 - D(G(\mathbf{z}_x)))]. \end{aligned} \tag{27}$$

One of the benefits of GAN is that we do not need the probability distribution assumptions for the GAN's input and output samples. This differs from the VAE, which can avoid inaccurate prior assumption problems. At present, GANs

have been widely used in speech enhancement tasks [36, 71, 72, 110, 111], but most of these algorithms less consider how a good speech representation can be obtained as the input of the GAN for SE. Typically, they directly use the observed signal as the GAN's input to generate the clean speech signal [71, 72]. Although there are no restrictions for the GAN's input, an accurate speech representation can usually make GAN easier to train and lead to better generative performance [31, 112, 113].

4 Contributions

This thesis mainly considers how to capture and analyze more underlying speech information when performing SE. To achieve this purpose, we have proposed to apply NMF, HMM, VAE, β-VAE, and GAN for SE. The main body of this thesis is constituted by papers A-F. Papers A and B introduce HMM to capture speech temporal dynamics information in NMF for SE. Moreover, based on the NMF-HMM framework in papers A and B, paper C utilizes the Poisson mixture model (PMM) to model signals, which intends to capture more underlying information. Paper D introduces the DRL model for SE, which leverages the VAEs to analyze underlying signal representations. Based on the proposed DRL framework in paper D, to obtain a better SE performance, paper E and paper F propose to apply β-VAE and GAN to optimize the information disentanglement and signal restoration ability of the previous DRL framework, respectively.

Paper A [25] This paper leverages HMM to capture the speech temporal dynamics information because HMM is easily combined with NMF for SE. More specifically, this paper presents a novel supervised NMF-HMM SE framework. This framework contains an offline training stage and an online enhancement stage. In the offline stage [25], the sum of Poisson is utilized as the observation model in each HMM's state, which ensures that computationally efficient MU is able to be applied in this framework. Based on NMF-HMM, we derive a novel MMSE estimator in the online stage. This novel MMSE estimator is able to perform online parallel computing, which reduces the time consumption of online SE. The experimental results show that the novel NMF-HMM framework can achieve a 7% STOI score advantage over the state-of-the-art NMF-based SE methods [17, 21], which illustrates that capturing the speech temporal dynamics information can obtain a better SE performance for the NMF-based SE algorithms.

Paper B [90] In paper A, the effectiveness of the NMF-HMM framework for SE has been preliminarily verified. However, there are various parameters in the proposed framework. The effect of these parameters on SE performance is not investigated in preliminary work. The proper parameter selection is critical for the NMF-HMM framework to achieve satisfactory SE performance. In

addition, paper A only conducts the limited noise experiment and does not perform experiments in more challenging noisy environments. The potential of the NMF-HMM algorithm is not totally investigated. This paper expands our preliminary research on these two aspects. Furthermore, in this paper, the proposed NMF-HMM framework is also compared with some more advanced DNN-based SE algorithms [114] and temporal NMF algorithms [115–117]. The NMF-HMM's performance is more comprehensively assessed in paper B.

Paper C [26] The NMF-HMM-based SE framework is proposed and analyzed in papers A and B. However, this framework only applies a single Poisson distribution as the HMM's likelihood function, which is difficult to effectively model the speech and noise signal since the signal behavior is complex. To mitigate this issue, this paper proposes [26] to apply the Poisson Mixture Model-based NMF-HMM (PMM-NMF-HMM) for SE, which is a more sophisticated statistical model. PMM-NMF-HMM has the potential to capture more complex signal behavior like the Gaussian mixture models (GMM) [118]. Compared to the previous NMF-HMM model, PMM-NMF-HMM can better describe the speech and noise signals since these signals may be governed by multiple underlying causes, and each of these causes may be responsible for one particular mixture component in the PMM distribution. If such causes are identified, then the PMM-NMF-HMM can be decomposed into a set of cause-dependent or context-dependent component distributions [26, 118]. Finally, better SE performance can be achieved by using PMM-NMF-HMM. Additionally, similar to the previous NMF-HMM-based SE method, PMM-NMF-HMM can leverage MU rule for the parameters. Based on this PMM-NMF-HMM framework, we also derive a PMM-NMF-HMM-based MMSE estimator, which can also perform parallel computations in the online SE stage. The experimental results show that the novel PMM-NMF-HMM algorithm is able to achieve a better STOI and PESQ performance than the previous NMF-HMM-based method.

Paper D [83] In papers A, B, and C, we have proposed an NMF-HMM-based SE framework. However, the main drawbacks of using HMM to capture underlying information are that we cannot understand what specific information is captured. Additionally, HMM cannot efficiently represent various underlying information, making it difficult for information analysis. To address these problems, we introduce the DRL for SE. DRL can represent and disentangle different signal information (as shown in Fig.1) in different vector forms [44, 82] for the related information analysis. More specifically, we propose to leverage VAE for SE. At present, VAE has been widely applied for SE. However, current VAE-based SE methods only apply VAE to model speech signal and use NMF to model noise since these VAE-based methods are difficult to disentangle the different information from the noisy signal. Based on Bayesian theory, this paper derives a novel ELBO for VAE. This ELBO help [83] VAEs perform training in semi-supervision and disentangle various latent variables from the noisy sig-

nal. Moreover, the proposed method leverages VAE to model noise, which is totally different from the previous VAE-based SE algorithm. The experimental results show that the proposed VAE framework can disentangle different latent variables from the noisy signal. Meanwhile, it also achieves a better scale-invariant signal-to-distortion ratio (SI-SDR) [119], PESQ, and STOI than the similar DNN-based SE method.

Paper E [120] In paper D, we have preliminarily verified that the proposed Bayesian permutation training VAE (PVAE) SE framework can effectively perform SE. In addition, the traditional DNN-based SE algorithm's performance can also be improved using DRL. This paper applies β-VAE further to improve the performance of representation disentanglement [120] in PVAE. Generally, a good signal representation is crucial for DNN to generate high-quality signals [44]. Specifically, the proposed β-VAE can improve PVAE's capacity of disentangling speech and noise latent variables from the noisy signal. Meanwhile, the proposed β-VAE addresses the trade-off issue between representation disentanglement and signal reconstructions [120], although this trade-off widely exists in β-VAE algorithms. Moreover, this β-VAE algorithm can also be used to reduce the model size of the PVAE framework. This means that the proposed algorithms can improve PVAE's SE performance when reducing the number of training parameters in PVAE. The experimental analysis indicates that the proposed β-VAE algorithm can acquire more disentangled signal (speech and noise) latent representations and achieve a better SE performance than the previous PVAE.

Paper F In paper E, we focus on optimizing the learning process of signal representation to achieve a better SE performance. The signal recovery is still based on the original VAE framework [83]. To obtain a higher quality enhanced speech, this paper proposes a two-stage DRL-based SE method using adversarial training. In the first stage, we apply the β-VAE algorithm to get the speech and noise posterior estimations from the noisy signal. Since the posteriors are intractable and we can only use conditional assumptions to estimate posteriors, it is difficult to ensure that these posterior estimations are always accurate. Inaccurate posterior estimations will degrade the final SE performance. In the second stage, we leverage adversarial training to reduce the effect of the inaccurate posterior toward signal reconstruction, which makes our previous algorithm more robust for inaccurate posterior estimations. The ability of signal restoration is strengthened in the second stage. Therefore, we can obtain a better SE performance than our previous algorithms. Moreover, the experimental results also indicate that our two-stage DRL-based SE algorithm outperforms recent competitive SE algorithms [71, 114].

5 Conclusions and Future Work

5.1 Conclusions

Understanding and capturing underlying signal information is crucial for SE because it can help us understand signal distortions. Meanwhile, it is also essential for some downstream speech applications, like ASR and speaker recognition. This thesis proposes to apply NMF-HMM and DRL model to capture underlying signal information from the observed signal when performing SE. NMF-HMM is a convenient way to capture underlying speech information using current NMF SE algorithms. However, we find that although NMF-HMM can capture the speech temporal dynamics information and improve traditional NMF's SE performance, the drawbacks of this framework are that we cannot understand what specific information is captured by NMF-HMM. Meanwhile, NMF-HMM cannot also represent the information in a vector form and disentangle the different information, which causes it challenging to perform detailed information analysis in SE. To address this problem, we introduce the DRL for SE. DRL can address these problems. More specifically, we propose a PVAE framework for SE. As a preliminary research of investing DRL's application in SE, we only apply the proposed PVAE to analyze speech and noise information of the noisy signal. The purpose is to verify the correctness of the proposed framework. More detailed information, such as text and speaker information, can be considered in future work. Our framework has the potential to analyze more complex information. The experimental results show that the proposed framework can successfully disentangle different latent variables and achieve better SE performance. In addition, we also indicate the DRL's importance for improving the performance of the DNN-based SE algorithms. Furthermore, we use β-VAE and GAN to further optimize the PVAE's information disentanglement and signal restoration ability, respectively. The experimental results also verify the effectiveness of two strategies for SE.

5.2 Future Work

One of the important contributions of this thesis is that we propose a DRL-based SE framework. However, in this work, we only preliminarily verify the correctness of this framework. The potential of this framework is still not achieved. As a result, we will end this summary by listing some possible research directions.

Speech Enhancement for ASR and Human Listening

At present, to improve office efficiency, there has been an increasing need for online meetings [121], where SE technique is wanted to help the online meeting system reduce the WER for accurate live captioning when transmitting enhanced speech signals [4–6]. However, SE techniques usually have a trade-off relationship between human listening and ASR. Some SE methods aimed

at improving human listening may degrade the performance of the ASR system due to signal distortion. Signal distortion may cause the information loss that is needed by the ASR system. In this thesis, we have proposed a DRL-based SE framework. As preliminary work, we only analyze the speech and noise information in the noisy signal. In future work, we can apply this framework to analyze and disentangle more underlying signal information, such as content-related and style-related information, as shown in Fig.1, which has the potential to prevent information loss in the signal distortion and can generate benefits for some downstream applications like ASR and speaker recognition.

Sophisticated probabilistic model and DNN framework for VAE-GAN SE algorithm

In paper F, we have proposed to leverage VAE and GAN for SE. However, this paper only applies the basic probability assumption and DNN model for SE. Therefore, two possible aspects can be considered in future work. At first, we can consider applying some more advanced distributions to describe signals. Currently, we only use the multivariate normal distributions. The following work can involve the complex Gaussian distributions in the current framework. Secondly, more advanced neural network structures can also be considered in the current VAE-GAN framework. For instance, we can leverage complex neural networks [37, 65, 122, 123] to perform prior and posterior estimations in VAE-GAN with complex Gaussian distributions. Moreover, some powerful generative models, such as HiFi-GAN [124, 125], Bayesian wavenet [99, 100], and diffusion probabilistic models [101–103], can be used as the decoder to generate high-quality speech.

Self-supervision-based Deep Representation Learning Algorithm

This thesis introduces the DRL for SE and proposes a VAE-GAN SE framework. However, this framework needs to apply the probability assumptions for the prior and posterior distributions. The probability assumptions cause we can only approximate speech data representations because we cannot know the real data distributions. In future work, if we want to describe data representation more accurately, one possible way is to remove these probability distribution assumptions and make the DRL algorithms learn data representations in a self-supervision way [78, 79, 82]. At present, many self-supervised speech representation learning [82] strategies have been proposed. These methods have huge potential to optimize the proposed DRL-based SE framework. As a result, we can capture more accurate speech information from noisy signals.

References

[1] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 22, no. 4, pp. 745–777, 2014.

[2] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[3] P. C. Loizou, *Speech enhancement: theory and practice*. CRC press, 2013.

[4] Y.-J. Lu, X. Chang, C. Li, W. Zhang, S. Cornell, Z. Ni, Y. Masuyama, B. Yan, R. Scheibler, Z.-Q. Wang *et al.*, "Espnet-se++: Speech enhancement for robust speech recognition, translation, and understanding," *arXiv preprint arXiv:2207.09514*, 2022.

[5] S. E. Eskimez, X. Wang, M. Tang, H. Yang, Z. Zhu, Z. Chen, H. Wang, and T. Yoshioka, "Human Listening and Live Captioning: Multi-Task Training for Speech Enhancement," in *Proc. Interspeech*, 2021, pp. 2686–2690.

[6] K. Iwamoto, T. Ochiai, M. Delcroix, R. Ikeshita, H. Sato, S. Araki, and S. Katagiri, "How bad are artifacts?: Analyzing the impact of speech enhancement errors on asr," *arXiv preprint arXiv:2201.06685*, 2022.

[7] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 27, no. 2, pp. 113–120, 1979.

[8] J. Lim and A. Oppenheim, "All-pole modeling of degraded speech," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 26, no. 3, pp. 197–210, 1978.

[9] P. Scalart *et al.*, "Speech enhancement based on a priori signal to noise estimation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 2. IEEE, 1996, pp. 629–632.

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

[11] ——, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 33, no. 2, pp. 443–445, 1985.

[12] I. Cohen and B. Berdugo, "Spectral enhancement by tracking speech presence probability in subbands," in *International Workshop on Hands-Free Speech Communication*, 2001.

[13] I. Cohen, "Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 5, pp. 466–475, 2003.

[14] I. Cohen and B. Berdugo, "Speech enhancement for non-stationary noise environments," *Signal Process.*, vol. 81, no. 11, pp. 2403–2418, 2001.

[15] T. Gerkmann and R. C. Hendriks, "Unbiased MMSE-based noise power estimation with low complexity and low tracking delay," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 20, no. 4, pp. 1383–1393, 2011.

[16] F. Jabloun and B. Champagne, "Incorporating the human hearing properties in the signal subspace approach for speech enhancement," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 6, pp. 700–708, 2003.

[17] J. R. Jensen, J. Benesty, and M. G. Christensen, "Noise reduction with optimal variable span linear filters," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 24, no. 4, pp. 631–644, 2015.

[18] K. B. Christensen, M. G. Christensen, J. B. Boldt, and F. Gran, "Experimental study of generalized subspace filters for the cocktail party situation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2016, pp. 420–424.

[19] S. Srinivasan, J. Samuelsson, and W. B. Kleijn, "Codebook driven short-term predictor parameter estimation for speech enhancement," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 14, no. 1, pp. 163–176, 2005.

[20] M. S. Kavalekalam, J. K. Nielsen, J. B. Boldt, and M. G. Christensen, "Model-based speech enhancement for intelligibility improvement in binaural hearing aids," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 27, no. 1, pp. 99–113, 2018.

[21] M. S. Kavalekalam, J. K. Nielsen, L. Shi, M. G. Christensen, and J. Boldt, "Online parametric NMF for speech enhancement," in *Proc. European Signal Processing Conf.*, 2018, pp. 2320–2324.

[22] 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.

[23] ——, "Algorithms for non-negative matrix factorization," in *Proc. Advances in Neural Inform. Process. Syst.*, 2001, pp. 556–562.

[24] N. Mohammadiha, P. Smaragdis, and A. Leijon, "Supervised and unsupervised speech enhancement using nonnegative matrix factorization," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 21, no. 10, pp. 2140–2151, 2013.

[25] Y. Xiang, L. Shi, J. L. Højvang, M. H. Rasmussen, and M. G. Christensen, "An NMF-HMM speech enhancement method based on kullback-leibler divergence," in *Proc. Interspeech*, 2020, pp. 2667–2671.

[26] ——, "A novel NMF-HMM speech enhancement algorithm based on poisson mixture model," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2021, pp. 721–725.

[27] D. Y. Zhao and W. B. Kleijn, "HMM-based gain modeling for enhancement of speech in noise," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 15, no. 3, pp. 882–892, 2007.

[28] F. Deng, C. Bao, and W. B. Kleijn, "Sparse Hidden Markov Models for speech enhancement in non-stationary noise environments," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 23, no. 11, pp. 1973–1987, 2015.

[29] Y. Bengio *et al.*, "Learning deep architectures for AI," *Foundations and trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

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

[31] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 26, no. 10, pp. 1702–1726, 2018.

[32] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 23, no. 1, pp. 7–19, 2014.

[33] Y. Wang, A. Narayanan, and D. Wang, "On training targets for supervised speech separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 22, no. 12, pp. 1849–1858, 2014.

[34] L. Sun, J. Du, L.-R. Dai, and C.-H. Lee, "Multiple-target deep learning for lstm-rnn based speech enhancement," in *2017 Hands-free Speech Communications and Microphone Arrays (HSCMA)*, 2017, pp. 136–140.

[35] Y. Luo and N. Mesgarani, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 27, no. 8, pp. 1256–1266, 2019.

[36] Y. Xiang and C. Bao, "A parallel-data-free speech enhancement method using multi-objective learning cycle-consistent generative adversarial network," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 28, pp. 1826–1838, 2020.

[37] Y. Hu, Y. Liu, S. Lv, M. Xing, S. Zhang, Y. Fu, J. Wu, B. Zhang, and L. Xie, "DCCRN: Deep Complex Convolution Recurrent Network for Phase-Aware Speech Enhancement," in *Proc. Interspeech*, 2020, pp. 2472–2476.

[38] A. Li, W. Liu, X. Luo, C. Zheng, and X. Li, "Icassp 2021 deep noise suppression challenge: Decoupling magnitude and phase optimization with a two-stage deep network," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2021, pp. 6628–6632.

[39] Z.-Q. Wang, G. Wichern, and J. Le Roux, "On the compensation between magnitude and phase in speech separation," *IEEE Signal Processing Letters*, vol. 28, pp. 2018–2022, 2021.

[40] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," vol. 21, no. 1, pp. 65–68, 2013.

[41] Z. Jin and D. Wang, "A supervised learning approach to monaural segregation of reverberant speech," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 17, no. 4, pp. 625–638, 2009.

[42] G. Kim, Y. Lu, Y. Hu, and P. C. Loizou, "An algorithm that improves speech intelligibility in noise for normal-hearing listeners," *The Journal of the Acoustical Society of America*, vol. 126, no. 3, pp. 1486–1494, 2009.

[43] Y. Shao and D. Wang, "Robust speaker identification using auditory features and computational auditory scene analysis," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2008, pp. 1589–1592.

[44] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[45] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, 2011.

[46] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 2, 2001, pp. 749–752.

[47] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2013, pp. 7092–7096.

[48] Y.-H. Tu, J. Du, and C.-H. Lee, "Speech enhancement based on teacher–student deep learning using improved speech presence probability for noise-robust speech recognition," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 27, no. 12, pp. 2080–2091, 2019.

[49] D. Wang and J. Lim, "The unimportance of phase in speech enhancement," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 30, no. 4, pp. 679–681, 1982.

[50] K. Paliwal, K. Wójcicki, and B. Shannon, "The importance of phase in speech enhancement," *speech communication*, vol. 53, no. 4, pp. 465–494, 2011.

[51] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 708–712.

[52] D. S. Williamson, Y. Wang, and D. Wang, "Complex ratio masking for monaural speech separation," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 24, no. 3, pp. 483–492, 2015.

[53] Z.-Q. Wang, K. Tan, and D. Wang, "Deep learning based phase reconstruction for speaker separation: A trigonometric perspective," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 71–75.

[54] Y. Luo and N. Mesgarani, "Tasnet: time-domain audio separation network for real-time, single-channel speech separation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2018, pp. 696–700.

[55] B. Du, W. Xiong, J. Wu, L. Zhang, L. Zhang, and D. Tao, "Stacked convolutional denoising auto-encoders for feature representation," *IEEE transactions on cybernetics*, vol. 47, no. 4, pp. 1017–1027, 2016.

[56] S.-W. Fu, Y. Tsao, and X. Lu, "Snr-aware convolutional neural network modeling for speech enhancement." in *Proc. Interspeech*, 2016, pp. 3768–3772.

[57] E. M. Grais and M. D. Plumbley, "Single channel audio source separation using convolutional denoising autoencoders," in *2017 IEEE global conference on signal and information processing (GlobalSIP)*. IEEE, 2017, pp. 1265–1269.

[58] S. R. Park and J. Lee, "A fully convolutional neural network for speech enhancement," *arXiv preprint arXiv:1609.07132*, 2016.

[59] D. Servan-Schreiber, A. Cleeremans, and J. McClelland, "Learning sequential structure in simple recurrent networks," *Advances in neural information processing systems*, vol. 1, 1988.

[60] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint optimization of masks and deep recurrent neural networks for monaural source separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 23, no. 12, pp. 2136–2147, 2015.

[61] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[62] J. Chen and D. Wang, "Long short-term memory for speaker generalization in supervised speech separation," *The Journal of the Acoustical Society of America*, vol. 141, no. 6, pp. 4705–4714, 2017.

[63] H. Zhao, S. Zarar, I. Tashev, and C.-H. Lee, "Convolutional-recurrent neural networks for speech enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2018, pp. 2401–2405.

[64] K. Tan and D. Wang, "A convolutional recurrent neural network for real-time speech enhancement." in *Proc. Interspeech*, vol. 2018, 2018, pp. 3229–3233.

[65] H.-S. Choi, J.-H. Kim, J. Huh, A. Kim, J.-W. Ha, and K. Lee, "Phase-aware speech enhancement with deep complex u-net," in *International Conference on Learning Representations*, 2018.

[66] C. Trabelsi, O. Bilaniuk, Y. Zhang, D. Serdyuk, S. Subramanian, J. F. Santos, S. Mehri, N. Rostamzadeh, Y. Bengio, and C. J. Pal, "Deep complex networks," *arXiv preprint arXiv:1705.09792*, 2017.

[67] Y. Hu, Y. Liu, S. Lv, M. Xing, S. Zhang, Y. Fu, J. Wu, B. Zhang, and L. Xie, "Dccrn: Deep complex convolution recurrent network for phase-aware speech enhancement," *arXiv preprint arXiv:2008.00264*, 2020.

[68] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[69] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Advances in Neural Inform. Process. Syst.*, 2014, pp. 2672–2680.

[70] S. Nowozin, B. Cseke, and R. Tomioka, "f-gan: Training generative neural samplers using variational divergence minimization," *Proc. Advances in Neural Inform. Process. Syst.*, vol. 29, 2016.

[71] D. Michelsanti and Z.-H. Tan, "Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification," *Proc. Interspeech*, pp. 2008–2012, 2017.

[72] S. Pascual, A. Bonafonte, and J. Serrà, "Segan: Speech enhancement generative adversarial network," *Proc. Interspeech*, pp. 3642–3646, 2017.

[73] S.-W. Fu, C. Yu, K.-H. Hung, M. Ravanelli, and Y. Tsao, "Metricgan-u: Unsupervised speech enhancement/dereverberation based only on noisy/reverberated speech," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2022, pp. 7412–7416.

[74] G. Yu, A. Li, Y. Wang, Y. Guo, H. Wang, and C. Zheng, "Joint magnitude estimation and phase recovery using cycle-in-cycle gan for non-parallel speech enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2022, pp. 6967–6971.

[75] G. Yu, Y. Wang, H. Wang, Q. Zhang, and C. Zheng, "A two-stage complex network using cycle-consistent generative adversarial networks for speech enhancement," *Speech Communication*, vol. 134, pp. 42–54, 2021.

[76] Y. Ju, W. Rao, X. Yan, Y. Fu, S. Lv, L. Cheng, Y. Wang, L. Xie, and S. Shang, "Tea-pse: Tencent-ethereal-audio-lab personalized speech enhancement system for icassp 2022 dns challenge," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2022, pp. 9291–9295.

[77] C. K. Reddy, H. Dubey, K. Koishida, A. Nair, V. Gopal, R. Cutler, S. Braun, H. Gamper, R. Aichner, and S. Srinivasan, "Interspeech 2021 deep noise suppression challenge," in *Proc. Interspeech*, 2021.

[78] K. Chan, Y. Yu, C. You, H. Qi, J. Wright, and Y. R. Ma, "a white-box deep network from the principle of maximizing rate reduction. arxiv. 2021," *arXiv preprint arXiv:2105.10446*, 2021.

[79] X. Dai, S. Tong, M. Li, Z. Wu, K. H. R. Chan, P. Zhai, Y. Yu, M. Psenka, X. Yuan, H. Y. Shum *et al.*, "Closed-loop data transcription to an ldr via minimaxing rate reduction," *arXiv preprint arXiv:2111.06636*, 2021.

[80] J. Wright and Y. Ma, *High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications*. Cambridge University Press, 2022.

[81] J. Lian, C. Zhang, G. K. Anumanchipalli, and D. Yu, "Towards improved zero-shot voice conversion with conditional dsvae," *arXiv preprint arXiv:2205.05227*, 2022.

[82] A. Mohamed, H.-y. Lee, L. Borgholt, J. D. Havtorn, J. Edin, C. Igel, K. Kirchhoff, S.-W. Li, K. Livescu, L. Maaløe *et al.*, "Self-supervised speech representation learning: A review," *arXiv preprint arXiv:2205.10643*, 2022.

[83] Y. Xiang, J. L. Højvang, M. H. Rasmussen, and M. G. Christensen, "A bayesian permutation training deep representation learning method for speech enhancement with variational autoencoder," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2022, pp. 381–385.

[84] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," *International Conference on Learning Representations*, 2017.

[85] C. P. Burgess, I. Higgins, A. Pal, L. Matthey, N. Watters, G. Desjardins, and A. Lerchner, "Understanding disentangling in β-vae," *arXiv preprint arXiv:1804.03599*, 2018.

[86] S. Sra and I. Dhillon, "Generalized nonnegative matrix approximations with bregman divergences," *Advances in neural information processing systems*, vol. 18, 2005.

[87] E. M. Grais and H. Erdogan, "Single channel speech music separation using nonnegative matrix factorization and spectral masks," in *Int. Conf. Digital Signal Process.*, 2011, pp. 1–6.

[88] K. W. Wilson, B. Raj, and P. Smaragdis, "Regularized non-negative matrix factorization with temporal dependencies for speech denoising," in *Proc. Interspeech*, 2008.

[89] G. J. Mysore, P. Smaragdis, and B. Raj, "Non-negative Hidden Markov modeling of audio with application to source separation," in *International Conference on Latent Variable Analysis and Signal Separation*. Springer, 2010, pp. 140–148.

[90] Y. Xiang, L. Shi, J. L. Højvang, M. H. Rasmussen, and M. G. Christensen, "A speech enhancement algorithm based on a non-negative hidden markov model and kullback-leibler divergence," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2022, no. 1, pp. 1–15, 2022.

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

[92] C. Févotte and J. Idier, "Algorithms for nonnegative matrix factorization with the β-divergence," *Neural Comput.*, vol. 23, no. 9, pp. 2421–2456, 2011.

[93] P. Paatero, "Least squares formulation of robust non-negative factor analysis," *Chemometrics and intelligent laboratory systems*, vol. 37, no. 1, pp. 23–35, 1997.

[94] D. FitzGerald, M. Cranitch, and E. Coyle, "On the use of the beta divergence for musical source separation," 2009.

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

[96] S. Nie, S. Liang, W. Liu, X. Zhang, and J. Tao, "Deep learning based speech separation via nmf-style reconstructions," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 26, no. 11, pp. 2043–2055, 2018.

[97] L. E. Baum, "An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes," *Inequalities*, vol. 3, no. 1, pp. 1–8, 1972.

[98] Y. Bando, M. Mimura, K. Itoyama, K. Yoshii, and T. Kawahara, "Statistical speech enhancement based on probabilistic integration of variational autoencoder and non-negative matrix factorization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 716–720.

[99] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[100] J. Chorowski, R. J. Weiss, S. Bengio, and A. Van Den Oord, "Unsupervised speech representation learning using wavenet autoencoders," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 27, no. 12, pp. 2041–2053, 2019.

[101] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 684–10 695.

[102] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *International Conference on Machine Learning*. PMLR, 2015, pp. 2256–2265.

[103] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[104] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization." *Journal of machine learning research*, vol. 12, no. 7, 2011.

[105] S. Leglaive, L. Girin, and R. Horaud, "A variance modeling framework based on variational autoencoders for speech enhancement," in *Proc. IEEE Workshop Machine Learning. Signal Process.*, 2018, pp. 1–6.

[106] ——, "Semi-supervised multichannel speech enhancement with variational autoencoders and non-negative matrix factorization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 101–105.

[107] G. Carbajal, J. Richter, and T. Gerkmann, "Guided variational autoencoder for speech enhancement with a supervised classifier," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 681–685.

[108] H. Fang, G. Carbajal, S. Wermter, and T. Gerkmann, "Variational autoencoder for speech enhancement with a noise-aware encoder," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 676–680.

[109] G. Carbajal, J. Richter, and T. Gerkmann, "Disentanglement learning for variational autoencoders applied to audio-visual speech enhancement," in *Proc. IEEE Workshop Appl. of Signal Process. to Aud. and Acoust.* IEEE, 2021, pp. 126–130.

[110] S.-W. Fu, C. Yu, K.-H. Hung, M. Ravanelli, and Y. Tsao, "Metricgan-u: Unsupervised speech enhancement/dereverberation based only on noisy/reverberated speech," *arXiv preprint arXiv:2110.05866*, 2021.

[111] W.-Y. Ting, S.-S. Wang, H.-L. Chang, B. Su, and Y. Tsao, "Speech enhancement based on cyclegan with noise-informed training," *arXiv preprint arXiv:2110.09924*, 2021.

[112] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," *arXiv preprint arXiv:2106.06103*, 2021.

[113] Y. Zhang, J. Cong, H. Xue, L. Xie, P. Zhu, and M. Bi, "Visinger: Variational inference with adversarial learning for end-to-end singing voice synthesis," *arXiv preprint arXiv:2110.08813*, 2021.

[114] S. Braun and I. Tashev, "Data augmentation and loss normalization for deep noise suppression," in *International Conference on Speech and Computer*. Springer, 2020, pp. 79–86.

[115] C. Févotte, J. Le Roux, and J. R. Hershey, "Non-negative dynamical system with application to speech and audio," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2013, pp. 3158–3162.

[116] P. Smaragdis, "Convolutive speech bases and their application to supervised speech separation," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 15, no. 1, pp. 1–12, 2006.

[117] P. D. O'grady and B. A. Pearlmutter, "Discovering speech phones using convolutive non-negative matrix factorisation with a sparseness constraint," *Neurocomputing*, vol. 72, no. 1-3, pp. 88–101, 2008.

[118] D. Yu and L. Deng, *Automatic Speech Recognition*. Springer, 2016.

[119] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr–half-baked or well done?" in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 626–630.

[120] Y. Xiang, J. L. Højvang, M. H. Rasmussen, and M. G. Christensen, "A deep representation learning speech enhancement method using β-vae," *Accepted by Eurosipco (arXiv preprint: arXiv:2205.05581)*, 2022.

[121] N. Pandey, A. Pal *et al.*, "Impact of digital surge during covid-19 pandemic: A viewpoint on research and practice," *International journal of information management*, vol. 55, p. 102171, 2020.

[122] C. Trabelsi, O. Bilaniuk, Y. Zhang, D. Serdyuk, S. Subramanian, J. Santos, S. Mehri, N. Rostamzadeh, Y. Bengio, and C. Pal, "Deep complex networks," *arXiv preprint arXiv:1705.09792*, 2017.

[123] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[124] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *Proc. Advances in Neural Inform. Process. Syst.*, vol. 33, pp. 17 022–17 033, 2020.

[125] G. Yang, S. Yang, K. Liu, P. Fang, W. Chen, and L. Xie, "Multi-band melgan: Faster waveform generation for high-quality text-to-speech," in *2021 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2021, pp. 492–498.

Part II

Papers

Paper A

An NMF-HMM Speech Enhancement Method based on Kullback-Leibler Divergence

Yang Xiang, Liming Shi, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, Mads Græsbøll Christensen

The paper has been published in the
Interspeech, 2020

The layout has been revised.

Abstract

In this paper, we present a novel supervised Non-negative Matrix Factorization (NMF) speech enhancement method, which is based on Hidden Markov Model (HMM) and Kullback-Leibler (KL) divergence (NMF-HMM). Our algorithm applies the HMM to capture the timing information, so the temporal dynamics of speech signal can be considered by comparing with the traditional NMF-based speech enhancement method. More specifically, the sum of Poisson, leading to the KL divergence measure, is used as the observation model for each state of HMM. This ensures that the parameter update rule of the proposed algorithm is identical to the multiplicative update rule, which is quick and efficient. In the training stage, this update rule is applied to train the NMF-HMM model. In the online enhancement stage, a novel minimum mean-square error (MMSE) estimator that combines the NMF-HMM is proposed to conduct speech enhancement. The performance of the proposed algorithm is evaluated by perceptual evaluation of speech quality (PESQ) and short-timeobjective intelligibility (STOI). The experimental results indicate that the STOI score of proposed strategy is able to outperform 7% than current state-of-the-art NMF-based speech enhancement methods.

1 Introduction

The aim of single-channel speech enhancement (SE) is to remove background noise from the noisy environment to improve quality and intelligibility of noisy speech. Nowadays, SE has achieved a wide range of applications in hearing aids, mobile communication, robust speech recognition (ASR) [1], teleconferencing and speech coding etc. Therefore, during the past decades, many different approaches have been proposed [2].

In an environment with additive noise, the spectral subtraction algorithm [3] is the simplest strategy to achieve SE, which subtracts the noise spectrum from the observed signal. Furthermore, some unsupervised algorithms like Wiener filtering [4], signal subspace algorithm [5], minimum mean-square error (MMSE) spectral amplitude estimator [6] and log-MMSE spectral amplitude estimator [7] are also the effective strategies to conduct the SE. However, these methods cannot always achieve satisfactory performance in the non-stationary noisy environment because they are usually based on some inaccurate assumptions and do not apply the prior information of clean speech and noise.

As a result, some supervised SE methods have been developed. These approaches usually consider to train a model and the model parameters are acquired by using the speech and noise signals. These methods include codebook-based algorithms [8], Hidden Markov Model (HMM)-based strategies [9] and Deep Neural Network (DNN)-based approaches [10–12] etc. These algorithms can make use of the prior information of clean speech and noise, so they can achieve better speech enhancement performance in practical noisy environments.

Non-negative Matrix Factorization (NMF)-based [13] [14] SE method can be also viewed as such a kind of supervised speech enhancement strategy. In paper [15], a mask-based NMF SE was proposed, which trained the basis matrix of clean speech and noise during offline stage. On the enhancement stage, the activation matrix could be acquired by combining the trained basis matrix and noisy signal. After that, the mask was estimated for the application of speech enhancement. In paper [16], an NMF-

based denoising scheme was proposed. This method added a heuristic term to the cost function, so the NMF coefficient can be adjusted according to the long-term levels of signals. Smaragdis et al. [17] proposed a supervised and unsupervised NMF speech enhancement method. In [17], the noise basis matrix could be acquired by combining the HMM during the enhancement stage. Thus, this method could mitigate the problem of noise mismatch. Furthermore, a NMF-based source separation approach was proposed in paper [18], which considers the HMM.

Inspired by these previous studies, in this paper, we proposed a novel NMF-HMM speech enhancement algorithm, which applies the Kullback-Leibler (KL) divergence. Compared to most NMF-based methods [13] [14], our method can utilize the temporal dynamics of speech signals to conduct the speech enhancement, so the time information of speech signal can be considered. Moreover, we used the sum of Poisson distribution as the state conditioned likelihood for the HMM rather than the general Gaussian Mixture Model (GMM), because the sum of Poisson distribution leads to the KL divergence measure, which is a mainstream measure in NMF, and its parameter update rule is identical to the multiplicative update rule. This ensures the parameter update is fast and efficient. On the enhancement stage, a minimum mean-square error (MMSE) estimator was derived to conduct SE, which was based on the NMF and HMM. The benefit of this algorithm is that the update of activation matrix can be conducted by parallel computing, which reduces the computation time.

2 NMF-based Speech Enhancement with KL Divergence

In this section, we will briefly review the NMF-based speech enhancement method with KL divergence. In this work, we only consider to achieve speech enhancement in the additive noisy environment. Thus, the noisy signal model can be represented as following:

$$y(t) = s(t) + m(t), \tag{A.1}$$

where $y(t)$, $s(t)$ and $m(t)$ are the noisy speech, clean speech and noise, respectively. The t is the time index. The short time Fourier transform (STFT) of $y(t)$ can be written as

$$Y(f,n) = S(f,n) + M(f,n), \tag{A.2}$$

where $Y(f,n)$, $S(f,n)$ and $M(f,n)$ are the complex STFT parameters of $y(t)$, $s(t)$ and $m(t)$, respectively. The f denotes frequency bin index and the n is the time frame index. For the sake of simplicity, we omit the frequency bin index, so their magnitude can be rewritten as the vectors $\mathbf{Y}_n$, $\mathbf{S}_n$ and $\mathbf{M}_n$.

For the NMF analysis, the magnitude of a signal V can be represented as

$$\mathbf{V} \approx \mathbf{WH}, \tag{A.3}$$

where W denotes the basis matrix and H denotes the activation matrix. Based on KL divergence, W and H can be estimated using iterative multiplicative update rules [14]

$$\mathbf{W} \leftarrow \mathbf{W} \odot \frac{\frac{\mathbf{V}}{\mathbf{WH}}\mathbf{H}^T}{\mathbf{1H}^T}, \tag{A.4}$$

$$\mathbf{H} \leftarrow \mathbf{H} \odot \frac{\mathbf{W}^T \frac{\mathbf{V}}{\mathbf{WH}}}{\mathbf{W}^T \mathbf{1}}, \tag{A.5}$$

where the $\odot$ and all divisions are element-wise multiplication and division operations, respectively. The $\mathbf{1}$ is the matrix of ones with the same size of $\mathbf{V}$ and T is the matrix transpose. For the application of speech enhancement, the speech basis matrix $\overline{\mathbf{W}}$ and noise basis matrix $\ddot{\mathbf{W}}$ can be estimated from clean speech and noise during the training stage. On the enhancement stage, the noisy speech basis matrix can be acquired by $\mathbf{W} = [\overline{\mathbf{W}}, \ddot{\mathbf{W}}]$. Additionally, the activation matrix $\mathbf{H}$ of noisy speech can be estimated by (A.5). After obtaining $\mathbf{H}$ and $\mathbf{W}$, the speech enhancement an be conducted by various algorithms [15] [16] [17] [18].

Furthermore, the [19] proves that the NMF with the KL divergence can be also motivated from the following hierarchical probability model

$$\mathbf{V} = \sum_{k=1}^{K} \mathbf{C}(k), \tag{A.6}$$

$$c_{f,n} \sim \mathcal{PO}(c_{f,n}(k); W_{f,k} H_{k,n}), \tag{A.7}$$

where the Poisson distribution $\mathcal{PO}(x;\lambda) = \frac{\lambda^x e^{-\lambda}}{\Gamma(x+1)}$, and $\Gamma(x+1) = x!$ is the Gamma function, K is the number of basis vectors, and $c_{f,n}$ is the latent variable of $\mathbf{C}(k)$ for Poisson distribution. Note, the $\mathbf{V}$ is assumed Poisson-distributed and integer-valued. In practice, the factorial is approximated by the Gamma function [19]. It has been shown that [19] the iterative update of the parameters $\mathbf{H}$ and $\mathbf{W}$ using Expectation–Maximization (EM) algorithm is identical to the multiplicative update rules (A.4) and (A.5).

3 NMF-HMM-based Speech Enhancement

In this section, the details of the proposed algorithm will be illustrated, which includes the proposed signal model, offline parameter learning and online speech enhancement.

3.1 HMM-based Signal Models with the KL Divergence

In our proposed approach, we need to acquire the three different signal models. They are namely clean speech model, noise model and noisy speech model. They will be separately introduced in this part. We use the overbar and double dots to represent the clean speech and noise, respectively.

In this work, there is the same signal model for the clean speech and the noise signal, so we will illustrate them just using clean speech signal. In order to model clean speech $\mathbf{S}_n$, we propose to a novel NMF-HMM-based method. To acquire a HMM model, there are three parameters [20] to be estimated. They are initial state probability $\overline{\boldsymbol{\pi}}$, transition probability matrix $\overline{\mathbf{A}}$ and state conditioned likelihood function. In addition, there are total $\overline{J}$ hidden states for this model. Thus, based on (A.6), we propose to model $\mathbf{S_n}$ as

$$\mathbf{S}_n = \sum_{k=1}^{\overline{K}} \overline{\mathbf{c}}_n(k), \tag{A.8}$$

By applying the (A.7) and HMM [16], for the jth ($j = 1, 2, \cdots, \overline{J}$) state, we can be defined

$$p(\overline{\mathbf{c}}_n|\overline{x}_n) = \prod_{f=1}^{F} \prod_{k=1}^{\overline{K}} \mathcal{PO}(\overline{c}_{f,n}(k); \overline{W}_{f,k}^{\overline{x}_n} \overline{H}_{k,n}^{\overline{x}_n}), \tag{A.9}$$

where the $\overline{x}_n$ is the hidden state and $\overline{x}_n \in \{1, 2, \cdots, \overline{J}\}$. $\overline{K}$ is the number of basis of clean speech and F is the total number of frequency bins. $\overline{W}_{k,n}^{\overline{x}_n}$ and $\overline{H}_{k,n}^{\overline{x}_n}$ is corresponding to the elements of the basis and activation for clean speech. Thus, the conditioned likelihood function at the jth state can be finally written as

$$p(\mathbf{S}_n|\overline{x}_n) = \prod_{f=1}^{F} \mathcal{PO}(S(f,n); \sum_{k=1}^{\overline{K}} \overline{W}_{f,k}^{\overline{x}_n} \overline{H}_{k,n}^{\overline{x}_n}), \tag{A.10}$$

where we use the superposition property of Poisson random variable. From (A.9), it can be found that there are $\overline{J}$ basis matrices for speech modelling, instead of one basis matrix in the traditional NMF, which is able to effectively capture the temporal dynamics of speech signals. The benefits of choosing the sum of Poission distribution as the state conditioned likelihood function is that its parameters update rules using EM algorithm is identical to the multiplicative update rules leading to low computational complexity. In addition, it is based on non-negative data by comparing with traditional HMM.

To sum up, the proposed model includes four parameters ($\overline{\mathbf{A}}$, $\overline{\boldsymbol{\pi}}$, $\overline{\mathbf{W}}^{\overline{x}_n}$ and $\overline{\mathbf{H}}^{\overline{x}_n}$). The $\overline{\mathbf{H}}^{\overline{x}_n}$ can be estimated by online speech enhancement and the other three parameters can be obtained by offline learning.

Based on proposed clean speech, noise signal model and (A.2), the noisy speech model can be defined. We assume that there are $\ddot{J}$ hidden states for noise and the hidden state of noise is $\ddot{x}_n (\ddot{x}_n \in \{1, 2, \cdots, \ddot{J}\})$. The $\ddot{\boldsymbol{\pi}}$ and $\ddot{\mathbf{A}}$ represent the initial state probability and transition probability matrix of the noise. Thus, there are total $\overline{J} \times \ddot{J}$ hidden states for noisy speech. The initial state and transition probabilities matrix of noisy speech can be expressed as $\overline{\boldsymbol{\pi}} \otimes \ddot{\boldsymbol{\pi}}$ and $\overline{\mathbf{A}} \otimes \ddot{\mathbf{A}}$, where the $\otimes$ denotes the Kronecker product. Finally, the conditioned likelihood function of noisy speech can be written as

$$\begin{aligned} &p(\mathbf{Y}_n|\overline{x}_n, \ddot{x}_n) = \\ &\prod_{f=1}^{F} \mathcal{PO}(Y(f,n); \sum_{k=1}^{\overline{K}} \overline{W}_{f,k}^{\overline{x}_n} \overline{H}_{k,n}^{\overline{x}_n} + \sum_{k=1}^{\ddot{K}} \ddot{W}_{f,k}^{\ddot{x}_n} \ddot{H}_{k,n}^{\ddot{x}_n}), \end{aligned} \tag{A.11}$$

where $\ddot{K}$, $\ddot{W}_{k,n}$ and $\ddot{H}_{k,n}$ is the number of basis, elements of the basis matrices and activation for noise.

3.2 Offline NMF-HMM Parameter Learning

In offline training stage, the aim is to find the parameter set $\boldsymbol{\Phi}$ to maximize the likelihood function, which is based on the HMM and EM algorithm [20]. There is the similar process for the parameter learning of clean speech and noise, so we will use the

clean speech as the example to illustrate this process. At first, we define the complete data set $(\mathbf{S_N}, \overline{\mathbf{X}}_\mathbf{N}, \overline{\mathbf{C}}_\mathbf{N})$, where $\mathbf{S_N} = [\mathbf{S}_1, \mathbf{S}_2, \cdots, \mathbf{S}_N]$, $\overline{\mathbf{X}}_\mathbf{N} = [\overline{x}_1, \overline{x}_2, \cdots, \overline{x}_N]^T$ and $\overline{\mathbf{C}}_\mathbf{N} = [\overline{\mathbf{c}}_1, \overline{\mathbf{c}}_2, \cdots, \overline{\mathbf{c}}_N]$. Thus, the complete data likelihood can be written as

$$p(\mathbf{S_N}, \overline{\mathbf{X}}_\mathbf{N}, \overline{\mathbf{C}}_\mathbf{N}) = \prod_{n=1}^{N} p(\mathbf{S}_n|\overline{\mathbf{c}}_n)p(\overline{\mathbf{c}}_n|\overline{x}_n)p(\overline{x}_n|\overline{x}_{n-1}). \tag{A.12}$$

By applying the EM algorithm in the expectation step, we first calculate the exact posterior state probability and joint posterior probability, which can be written as

$$q(\overline{x}_n) = p(\overline{x}_n|\mathbf{S_N}; \mathbf{\Phi}^{i-1}), \tag{A.13}$$

$$q(\overline{x}_n, \overline{x}_{n-1}) = p(\overline{x}_n, \overline{x}_{n-1}|\mathbf{S_N}; \mathbf{\Phi}^{i-1}), \tag{A.14}$$

where i is the iteration number. The calculation of (A.13) and (A.14) can be performed using forward-backward algorithm [20]. Then, we need to evaluate the posterior Expectation $\mathbb{E}_{\overline{\mathbf{c}}_n|\mathbf{S_N}, \overline{x}_n; \mathbf{\Phi}^{i-1}}(\overline{\mathbf{c}}_n)$, which will be used in M-step. By using Bayesian rule and conditional independence property of the proposed HMM model, combining (A.8), (A.9) and following the derivation in paper [19], we have

$$\begin{aligned} &q(\overline{\mathbf{c}}_n|\overline{x}_n) = \\ &\prod_{f=1}^{F} \mathcal{M}(\overline{c}_{f,n}(1), \cdots, \overline{c}_{f,n}(K); S(f,n), p_{f,n}^{\overline{x}_n}(1), \cdots, p_{f,n}^{\overline{x}_n}(K)), \end{aligned} \tag{A.15}$$

where $\mathcal{M}(\cdot)$ is the multinomial distribution [19].

$$p_{f,n}^{\overline{x}_n}(k) = \frac{\overline{W}_{f,k}^{\overline{x}_n} \overline{H}_{k,n}^{\overline{x}_n}}{\sum_{l=1}^{K} \overline{W}_{f,l}^{\overline{x}_n} \overline{H}_{l,n}^{\overline{x}_n}}. \tag{A.16}$$

Finally, we have

$$\mathbb{E}(\overline{c}_{f,n}(k)|\mathbf{S_N}, \overline{x}_n) = S(f,n) \frac{\overline{W}_{f,k}^{\overline{x}_n} \overline{H}_{k,n}^{\overline{x}_n}}{\sum_{l=1}^{K} \overline{W}_{f,l}^{\overline{x}_n} \overline{H}_{l,n}^{\overline{x}_n}}. \tag{A.17}$$

In the maximization step, the purpose is to find parameters to maximize the expected value of complete data likelihood, i.e.,

$$\mathbf{\Phi}^i = \arg\max_{\mathbf{\Phi}} \mathbb{E}_{\overline{\mathbf{X}}_\mathbf{N}, \overline{\mathbf{C}}_\mathbf{N}|\mathbf{S_N}; \mathbf{\Phi}^{i-1}} [\log p(\mathbf{S_N}, \overline{\mathbf{X}}_\mathbf{N}, \overline{\mathbf{C}}_\mathbf{N})]. \tag{A.18}$$

By using (A.18), the estimation of $\overline{\mathbf{A}}$ and $\overline{\boldsymbol{\pi}}$ is the same as the traditional HMM model [20]. To obtain $\overline{\mathbf{W}}^{\overline{x}_n}$ and $\overline{\mathbf{H}}^{\overline{x}_n}$, we can set the derivatives in (A.18) to zero. Thus, the update of parameters can be written as following:

$$\overline{\pi}_j = \frac{q(\overline{x}_1 = j)}{\sum_{o=1}^{\overline{J}} q(\overline{x}_1 = o)}, \tag{A.19}$$

$$\overline{A}_{o,j} = \frac{\sum_{n=2}^{\overline{N}} q(\overline{x}_n = j, \overline{x}_{n-1} = o)}{\sum_{j=1}^{\overline{J}} \sum_{n=2}^{\overline{N}} q(\overline{x}_n = j, \overline{x}_{n-1} = o)}, \tag{A.20}$$

where $1 \leq o, j \leq \overline{J}$.

$$\overline{\mathbf{W}}^{\overline{x}_n} \leftarrow \overline{\mathbf{W}}^{\overline{x}_n} \odot \frac{\dfrac{\mathbf{S_N}}{\overline{\mathbf{W}}^{\overline{x}_n}\overline{\mathbf{H}}^{\overline{x}_n}}\mathbf{\Lambda}(j)(\overline{\mathbf{H}}^{\overline{x}_n})^T}{\mathbf{1}\mathbf{\Lambda}(j)(\overline{\mathbf{H}}^{\overline{x}_n})^T}, \tag{A.21}$$

$$\overline{\mathbf{H}}^{\overline{x}_n} \leftarrow \overline{\mathbf{H}}^{\overline{x}_n} \odot \frac{(\overline{\mathbf{W}}^{\overline{x}_n})^T\dfrac{\mathbf{S_N}}{\overline{\mathbf{W}}^{\overline{x}_n}\overline{\mathbf{H}}^{\overline{x}_n}}}{(\overline{\mathbf{W}}^{\overline{x}_n})^T\mathbf{1}}, \tag{A.22}$$

where $\mathbf{\Lambda}(j) = \mathrm{diag}(q(\overline{x}_1 = j), q(\overline{x}_2 = j), \cdots, q(\overline{x}_N = j))$. From (A.21) and (A.22), we can find that the parameters update of proposed algorithm is identical to the multiplicative update rule. This ensures that our method is efficient and quick.

3.3 MMSE-based Online Speech Enhancement

In this work, we proposed to combine the NMF-HMM model with MMSE estimator to conduct online speech enhancement. Thus, the estimated signal can be represented as

$$\hat{\mathbf{S}}_n = \mathbb{E}_{\mathbf{S}_n|\mathbf{Y_n}}(\mathbf{S}_n) = \int \mathbf{S}_n p(\mathbf{S}_n|\mathbf{Y_n})\, d\mathbf{S}_n, \tag{A.23}$$

where $\mathbf{Y_n}$ is defined similar to $\mathbf{S_N}$. We ignore specific details of derivation, the enhanced speech can be represented as

$$\hat{\mathbf{S}}_n = \mathbf{Y}_n \odot \left(\sum_{\overline{x}_n, \ddot{x}_n} \omega_{\overline{x}_n, \ddot{x}_n} \mathbf{p}_n(\overline{x}_n, \ddot{x}_n) \right), \tag{A.24}$$

where $\omega_{\overline{x}_n, \ddot{x}_n}$ is the weight, which can be written as

$$\omega_{\overline{x}_n, \ddot{x}_n} = \frac{p(\mathbf{Y}_n|\overline{x}_n, \ddot{x}_n)p(\overline{x}_n, \ddot{x}_n|\mathbf{Y_{n-1}})}{\sum_{\overline{x}_n, \ddot{x}_n} p(\mathbf{Y}_n|\overline{x}_n, \ddot{x}_n)p(\overline{x}_n, \ddot{x}_n|\mathbf{Y_{n-1}})}. \tag{A.25}$$

$$\begin{aligned} &p(\overline{x}_n, \ddot{x}_n|\mathbf{Y_{n-1}}) \\ &= \sum_{\overline{x}_{n-1}, \ddot{x}_{n-1}} p(\overline{x}_n, \ddot{x}_n|\overline{x}_{n-1}, \ddot{x}_{n-1})p(\overline{x}_{n-1}, \ddot{x}_{n-1}|\mathbf{Y_{n-1}}) \end{aligned} \tag{A.26}$$

In (A.26), the first term can be acquired by the transition probabilities matrix of noisy speech and the second term is the forward probability that can be calculated by forward algorithm [20]. Additionally, $\mathbf{p}_n(\overline{x}_n, \ddot{x}_n)$ can be represented as

$$\mathbf{p}_n(\overline{x}_n, \ddot{x}_n) = \frac{\overline{\mathbf{W}}^{\overline{x}_n}\overline{\mathbf{H}}^{\overline{x}_n}}{\overline{\mathbf{W}}^{\overline{x}_n}\overline{\mathbf{H}}^{\overline{x}_n} + \ddot{\mathbf{W}}^{\ddot{x}_n}\ddot{\mathbf{H}}^{\ddot{x}_n}}. \tag{A.27}$$

In enhancement stage, the $\ddot{\mathbf{H}}^{\ddot{x}_n}$ and $\overline{\mathbf{H}}^{\overline{x}_n}$ can be acquired by (A.5). After that, the enhanced speech can be estimated from (A.24) to (A.27). The equation (A.24) shows that there are more than one basic and activation matrix to be applied to acquire gain to conduct speech enhancement. This is because the proposed algorithm utilize the HMM and consider the temporal aspect. Additionally, the update of activation matrix ($\ddot{\mathbf{H}}^{\ddot{x}_n}$ and $\overline{\mathbf{H}}^{\overline{x}_n}$) can be conducted by parallel computing. This means that our algorithm can reduce the time assumption during the online stage.

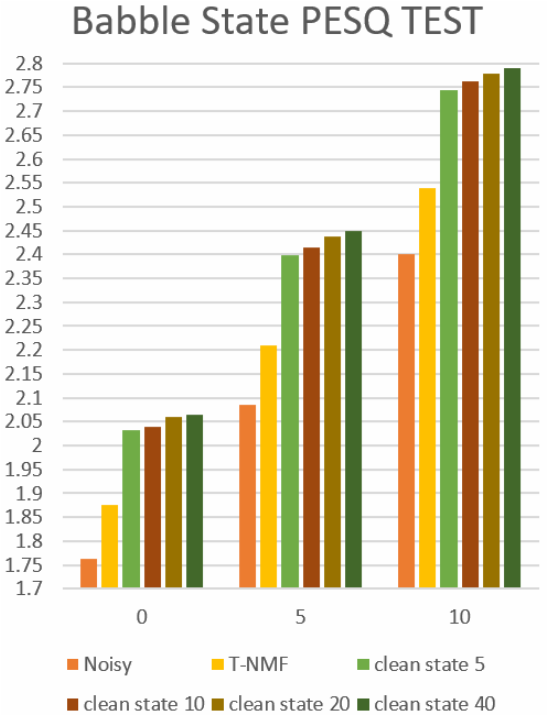


Fig. A.1: PESQ score of proposed algorithm in babble noise with different numbers of state.

4 Experiments and Results

4.1 Experimental Database Preparation

In this study, the proposed algorithm is expected to be evaluated by TIMIT [21] and NOISEX-92 [22] database. During the training stage, all the 4620 utterances from the training TIMIT database are used to train the clean NMF-HMM model. Additionally, the Babble, F16, Factory and White noise from the NOISEX-92 is also used to train the noise NMF-HMM model. During the test stage, the 200 utterances from the TIMIT test set are randomly chosen to build the test database. Then, four types of noise are added at three different SNR levels (0, 5 and 10dB). The test noise types are F16, Babble, Factory, and White.

In our experiments, all the signals are down-sampled to 16 kHz. The frame length is 1024 samples (64 ms) with a frame shift of 512 samples (32 ms). The size of short time Fourier transform (STFT) is 1024 points with a Hanning window.

4.2 Performance Evaluation of Speech Enhancement

In order to evaluate the performance of the proposed algorithm, there are two test stages. In the first stage, we will investigate the effects of different parameters for NMF-

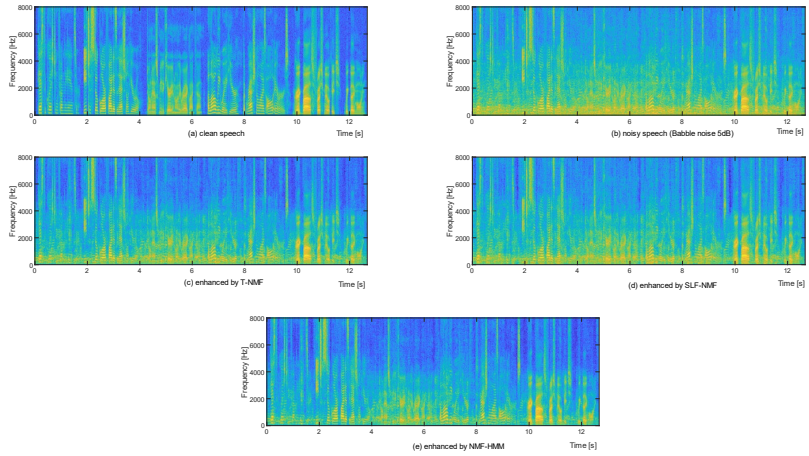


Fig. A.2: Spectrum comparison of various NMF-based methods: (a)clean speech, (b)noisy speech with 5dB Babble noise, (c)(d)(e)enhanced speech by T-NMF, SLP-NMF and NMF-HMM,respectivelys

HMM model. This test will be conducted on the babble noise. More specifically, we will investigate the effect of different numbers of state of clean speech for the performance of speech enhancement. In this experiment, the dimension of clean speech and noise mixture is fixed to 25 and 70, respectively, which is based on the previous research [15]. The state of noise is fixed to 2 because we want to show that the proposed algorithm can apply the different noise state to conduct speech enhancement. In this stage, the test result will be evaluated by PESQ [23] and we apply the traditional NMF-based [15] speech enhancement algorithm (T-NMF) as reference method. The aim of this experiment is to acquire the most suitable parameters of NMF-HMM model. Figure A.1 shows the experimental result. We can find that the proposed method can achieve the better performance the T-NMF. Additionally, the 40 states for clean speech can achieve the highest score under the all three SNRs. In second stage, the proposed algorithm is expected to be conducted on the more types of noise, which is Babble, F16, Factory and White noise, respectively. We apply the traditional NMF-based [15] speech enhancement algorithm (T-NMF), Optimally-Modified Log-Spectral Amplitude (OM-LSA) method [24] with IMCRA noise estimator [25], linear span filters method [26] (SLF-NMF) that applies the parametric NMF [27] and Log-MMSE [28] algorithm as the reference method. STOI [29] is used to evaluate the performance. For the SLF-NMF, the maximum SNR filter is chosen to conduct the speech enhancement. Furthermore, for the SLF-NMF, the codebook size of clean speech and noise is 64 entries and 8 entries, respectively. The dimension of basic matrix for T-NMF is the same as NMF-HMM. Figure A.2 shows the spectrum comparison of various NMF-based methods. It can be found that the proposed NMF-HMM method is able to remove more noise than other NMF-based method. Meanwhile, NMF-HMM can also recover the more speech information. Figure A.3 indicates the average STOI result with the 95% confidential interval (There are four types of noise under three SNRs, each situation includes 200 utterances. Therefore, the average score is acquired by $200\times3\times4=2400$ utterances.) This result shows that NMF-HMM

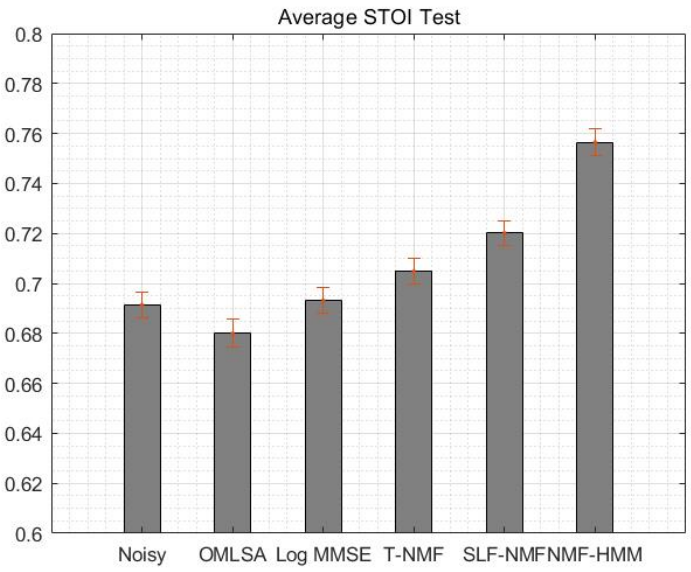


Fig. A.3: Average STOI score for four types of noise under three SNRs.

can effectively improve the more speech intelligibility than T-NMF and other reference methods.

5 Conclusions

In this paper, a novel HMM-NMF speech enhancement method is proposed. The core idea is to apply the sum of Poisson as the observation model for each state of HMM because it can ensure that the parameter update rule is identical to the multiplicative update rule. This is quick and efficient. In addition, this method can consider the temporal dynamics of speech signal because of the application of HMM. Furthermore, we proposed a novel HMM-NMF-based MMSE estimator to conduct the online speech enhancement. The experimental results indicate that the proposed algorithm can achieve better speech enhancement performance than these state-of-the-art statistic-based and NMF-based methods.

References

[1] Jinyu Li, Li Deng, Yifan Gong, and Reinhold Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 22, no. 4, pp. 745–777, 2014.

[2] Israel Cohen and Sharon Gannot, "Spectral enhancement methods," in *Springer Handbook of Speech Processing*, pp. 873–902. Springer, 2008.

[3] Steven Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 27, no. 2, pp. 113–120, 1979.

[4] Jae Lim and Alan Oppenheim, "All-pole modeling of degraded speech," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 26, no. 3, pp. 197–210, 1978.

[5] Firas Jabloun and Benoît Champagne, "Incorporating the human hearing properties in the signal subspace approach for speech enhancement," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 6, pp. 700–708, 2003.

[6] Yariv Ephraim and David Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 32, no. 6, pp. 1109–1121, 1984.

[7] Yariv Ephraim and David Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 33, no. 2, pp. 443–445, 1985.

[8] Sriram Srinivasan, Jonas Samuelsson, and W Bastiaan Kleijn, "Codebook-based bayesian speech enhancement for nonstationary environments," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 15, no. 2, pp. 441–452, 2007.

[9] David Y Zhao and W Bastiaan Kleijn, "HMM-based gain modeling for enhancement of speech in noise," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 15, no. 3, pp. 882–892, 2007.

[10] DeLiang Wang and Jitong Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 26, no. 10, pp. 1702–1726, 2018.

[11] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis, "Joint optimization of masks and deep recurrent neural networks for monaural source separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 23, no. 12, pp. 2136–2147, 2015.

[12] Yang Xiang and Changchun Bao, "A parallel-data-free speech enhancement method using multi-objective learning cycle-consistent generative adversarial network," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 28, pp. 1826–1838, 2020.

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

[14] Daniel D Lee and H Sebastian Seung, "Algorithms for non-negative matrix factorization," in *Proc. Advances in Neural Inform. Process. Syst.*, 2001, pp. 556–562.

[15] Emad M Grais and Hakan Erdogan, "Single channel speech music separation using nonnegative matrix factorization and spectral masks," in *Int. Conf. Digital Signal Process.*, 2011, pp. 1–6.

[16] Kevin W Wilson, Bhiksha Raj, and Paris Smaragdis, "Regularized non-negative matrix factorization with temporal dependencies for speech denoising," in *Proc. Interspeech*, 2008.

[17] Nasser Mohammadiha, Paris Smaragdis, and Arne Leijon, "Supervised and unsupervised speech enhancement using nonnegative matrix factorization," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 21, no. 10, pp. 2140–2151, 2013.

[18] Gautham J Mysore, Paris Smaragdis, and Bhiksha Raj, "Non-negative Hidden Markov modeling of audio with application to source separation," in *International Conference on Latent Variable Analysis and Signal Separation*. Springer, 2010, pp. 140–148.

[19] Ali Taylan Cemgil, "Bayesian inference for nonnegative matrix factorisation models," *Computational intelligence and neuroscience*, vol. 2009, 2009.

[20] Leonard E Baum, "An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes," *Inequalities*, vol. 3, no. 1, pp. 1–8, 1972.

[21] John S Garofolo, Lori F Lamel, William M Fisher, Jonathan G Fiscus, and David S Pallett, "DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1," *NASA STI/Recon technical report n*, vol. 93, 1993.

[22] Andrew Varga and Herman JM Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 12, no. 3, pp. 247–251, 1993.

[23] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2001, vol. 2, pp. 749–752.

[24] Israel Cohen and Baruch Berdugo, "Speech enhancement for non-stationary noise environments," *Signal Process.*, vol. 81, no. 11, pp. 2403–2418, 2001.

[25] Israel Cohen, "Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 5, pp. 466–475, 2003.

[26] Jesper Rindom Jensen, Jacob Benesty, and Mads Græsbøll Christensen, "Noise reduction with optimal variable span linear filters," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 24, no. 4, pp. 631–644, 2015.

[27] Mathew Shaji Kavalekalam, Jesper Kjær Nielsen, Liming Shi, Mads Græsbøll Christensen, and Jesper Boldt, "Online parametric NMF for speech enhancement," in *Proc. European Signal Processing Conf.*, 2018, pp. 2320–2324.

[28] Timo Gerkmann and Richard C Hendriks, "Unbiased MMSE-based noise power estimation with low complexity and low tracking delay," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 20, no. 4, pp. 1383–1393, 2011.

[29] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, 2011.

Paper B

A Speech Enhancement Algorithm Based on a Non-Negative Hidden Markov Model and Kullback-Leibler Divergence

Yang Xiang, Liming Shi, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, Mads Græsbøll Christensen

The paper has been published in the
EURASIP Journal on Audio, Speech, and Music Processing, 2022

The layout has been revised.

Abstract

In this paper, we propose a supervised single-channel speech enhancement method that combines Kullback-Leibler (KL) divergence-based non-negative matrix factorization (NMF) and a hidden Markov model (NMF-HMM). With the integration of the HMM, the temporal dynamics information of speech signals can be taken into account. This method includes a training and enhancement stage. In the training stage, the sum of the Poisson distribution, leading to the KL divergence measure, is used as the observation model for each state of the HMM. This ensures that a computationally efficient multiplicative update can be used for the parameter update of this model. In the online enhancement stage, a novel minimum mean-square error estimator is proposed for the NMF-HMM. This estimator can be implemented using parallel computing, reducing the time complexity. Moreover, compared to the traditional NMF-based speech enhancement methods, the experimental results show that our proposed algorithm improved the short-time objective intelligibility and perceptual evaluation of speech quality by 5% and 0.18, respectively.

1 Introduction

Single-channel speech enhancement technology is being widely used in our daily lives, such as in speech coding, teleconferencing, hearing aids, mobile communication, and automated robust speech recognition (ASR) [1, 2]. In general, the purpose of speech enhancement is to remove background noise from an audio source while preserving clean speech. It aims to improve the quality and intelligibility of noisy speech [3]. Currently, single-channel speech enhancement is an active topic of research.

During the past decades, many different monaural speech enhancement approaches have been proposed [2, 4]. In an environment with additive noise, the simplest approach to speech enhancement is the spectral subtraction algorithm [5], which subtracts the estimated noise spectrum from the observed signal to acquire the desired clean speech. Other unsupervised methods, such as the signal subspace algorithm [6–9], Wiener filtering [10], the minimum mean-square error (MMSE) spectral amplitude estimator [11], and a log-MMSE spectral amplitude estimator [12], are effective strategies for speech enhancement when the noise is stationary. These methods have low computational complexity and have been widely applied in various areas. However, these approaches cannot always achieve satisfactory performance for non-stationary noise and usually introduce musical noise because they do not make the best use of the prior information of the speech and noise [13]. Moreover, most unsupervised methods are based on the statistical properties of the speech and noise signals. However, it is difficult to meet these properties in actual noisy scenarios [14].

Therefore, supervised speech enhancement approaches have been developed. For instance, Kavalekalam [15] proposed a codebook-based Kalman filter speech enhancement method, which performs a listening test and shows significant improvement for speech intelligibility. In addition, Srinivasan [16] proposed a codebook-driven speech enhancement algorithm for non-stationary noise. In this work, the auto-regressive (AR) spectrum shape codebooks of speech and noise were pre-trained. In the enhancement stage, the codebooks could be used to build a Wiener filter to conduct speech enhancement. Inspired by this research, many other codebook-based speech enhancement approaches have been developed [17, 18]. Furthermore, an auto-regressive hid-

den Markov model (ARHMM) [19, 20] has also been shown to be an effective supervised speech enhancement method because it considers the temporal information of the speech signal.

In recent years, advances in deep learning techniques [21, 22], specifically, deep neural networks (DNNs), have significantly promoted the development of speech enhancement [23]. These methods usually rely on fewer assumptions [3, 14, 23] between the noise and clean speech, so they have huge potential to achieve better speech enhancement performance. Xu [3, 14] applied a feedforward multilayer perceptron (MLP) to map log-power spectrum (LPS) features of clean speech given noisy LPS input; the enhanced speech could be obtained directly by waveform reconstruction. Compared to the MMSE estimator [12], this method achieved better performance in various noisy environments. Wang [24, 25] also utilized an MLP to estimate the ideal ratio mask (IRM) and ideal binary mask (IBM) in conducting speech enhancement and also achieved satisfactory performance. Motivated by this work, researchers has used different DNN structures to conduct speech enhancement, such as a fully convolutional neural network (FCN) [26], deep recurrent neural networks (DRNN) [27, 28], and generative adversarial networks (GANs) [29, 30]. These methods could help ASR systems achieve higher recognition accuracy in noisy environments. However, generalization is always a problem that needs to be considered for these DNN-based algorithms [31, 32].

A non-negative matrix factorization (NMF)-based speech enhancement algorithm [33–35] can also be viewed as a kind of supervised speech enhancement method. NMF-based methods usually include a training and enhancement stage. In [36], a mask-based NMF speech enhancement method was proposed. In the training stage, the basis matrix of clean speech and noise was trained. In the enhancement stage, the activation matrix could be acquired by combining the trained basis matrix and noisy signal. The mask was then estimated to conduct the speech enhancement. Additionally, an NMF-based denoising scheme was described in [37, 38], which added a heuristic term to the cost function, so the NMF coefficients could be adjusted according to the long-term levels of the signals. A parametric NMF method for speech enhancement was proposed in [17]. This method applied the AR coefficient and codebook to build the basis matrix. This strategy effectively improved the speech intelligibility. Moreover, some DNN-based NMF methods represent an effective strategy for conducting speech enhancement [39, 40]. In general, the basis matrix could be acquired using the traditional NMF method, and the activation matrix could be estimated by applying a DNN, which improved the accuracy of the estimated activation matrix. Thus, it could achieve a higher perceptual evaluation of speech quality (PESQ) [41] and short-time objective intelligibility (STOI) [42] scores than traditional NMF-based speech enhancement methods. The combination of DNN and NMF could also help the ASR system achieve a lower word error rate (WER) in noisy environments. In [43], a DNN-NMF-based method achieved excellent performance in the Computational Hearing in Multisource Environments (CHiME)-3 challenge. To capture temporal information, some HMM-based NMF speech enhancement methods have been proposed. Mohammadiha [44] proposed a supervised and unsupervised NMF speech enhancement method. In [44], an HMM was used for modeling the temporal change of different noise types. In [45], a non-negative factorial HMM was used to model sound mixtures and showed superior performance in source separation tasks. In [46], an HMM-DNN NMF speech enhancement algorithm was proposed, which applied a clustering method to acquire the HMM-based basis matrix and used the Viterbi algorithm to obtain the ideal state label for the DNN training. In the enhancement stage, the DNN was used to find the corresponding state to conduct

speech enhancement.

In this paper, we propose a novel NMF-HMM speech enhancement method based on the Kullback-Leibler (KL) divergence, expanding on our preliminary work [47]. Our preliminary work has briefly verified the effectiveness of an NMF-HMM for speech enhancement [47, 48], but the effect of the parameters for the model was not considered. This is very important to optimize the algorithm performance. Additionally, its performance in various noisy environments was also not investigated. In this paper, we expand our preliminary research on these two aspects. Compared to other HMM-based methods [44, 45, 49], our method uses the HMM to capture the temporal dynamics of the speech and noise signal. Moreover, we use the sum of the Poisson distribution as the state-conditioned likelihood for the HMM, rather than the general Gaussian mixture model (GMM), because the sum of the Poisson distribution leads to the KL divergence measure. KL divergence is a mainstream measure in NMF, and its parameter update rule is identical to the multiplicative update rule. This ensures that the parameter update is computationally efficient during the training stage. In the enhancement stage, in contrast with previous works [44, 45], we propose a novel NMF-HMM-based MMSE estimator to perform the online enhancement. A major benefit of the proposed algorithm is that the activation matrix could be updated by parallel computing in the online stage. This could effectively reduce computational time. In this paper, we also show a more detailed algorithm derivation towards the preliminary NMF-HMM-based algorithm [47]. Moreover, the proposed method was compared with other state-of-the-art speech enhancement algorithms, which further indicated the advantages of the proposed algorithm.

The rest of this paper is organized as follows. First, we will briefly review the general NMF-based speech enhancement method with KL divergence in Section 2. The proposed HMM-based signal model will be introduced in Section 3, and the more detailed offline parameter learning will be explained in Section 4. The details of the proposed MMSE estimator and online speech enhancement process will be given in Section 4. The experimental comparison and analysis of results will be illustrated in Section 5, and we will draw conclusions in Section 6.

2 NMF-based Speech Enhancement Method with KL Divergence

In this section, we will briefly review NMF-based speech enhancement with KL divergence. Under the additive noise assumption, the noisy signal model can be expressed as

$$y(t) = s(t) + m(t), \tag{B.1}$$

where $y(t)$, $s(t)$, and $m(t)$ denote the noisy signal, clean speech, and noise, respectively, and t is the time index. With (B.1), the short-time Fourier transform (STFT) of $y(t)$ can be written as

$$Y(f,n) = S(f,n) + M(f,n), \tag{B.2}$$

where $Y(f,n)$, $S(f,n)$, and $M(f,n)$ denote the frequency spectrums of $y(t)$, $s(t)$, and $m(t)$, respectively. Here, $f \in [1,F]$ and $n \in [1,N]$ denote the frequency bin and time frame indices, respectively. Collecting the F frequency bins and N time frames, we define the magnitude spectrum matrices $\mathbf{Y}_N$, $\mathbf{S}_N$, and $\mathbf{M}_N$, where

$\mathbf{Y}_N = [\mathbf{y}_1, \cdots, \mathbf{y}_n, \cdots, \mathbf{y}_N]$ and $\mathbf{y}_n = [|Y(1,n)|, \cdots, |Y(f,n)|, \cdots, |Y(F,n)|]^T$ and also $\mathbf{s}_n$ and $\mathbf{m}_n$ are defined similarly to $\mathbf{y}_n$. Additionally, $\mathbf{S}_N$ and $\mathbf{M}_N$ are defined similarly to $\mathbf{Y}_N$; we assume that $\mathbf{Y}_N = \mathbf{S}_N + \mathbf{M}_N$. The classical NMF-based speech enhancement has two stages: training and enhancement. In the training stage, the clean speech basis matrix $\overline{\mathbf{W}}$ and noise basis matrix $\ddot{\mathbf{W}}$ are trained using clean speech and noise databases, respectively. Many cost functions have been proposed for NMF, such as KL divergence [34], Itakura-Saito (IS) divergence [50], β divergence, and Euclidian distance [51]. In this paper, we focus on using the KL divergence measure. There are two reasons for this choice. First, compared with other cost functions, the best speech enhancement performance can be achieved using the KL divergence-based NMF with the magnitude spectrum [52]. Second, the efficient multiplicative update (MU) rule of the KL divergence-based NMF can be also derived statistically using the expectation maximization (EM) algorithm [53]. For the two matrices $\mathbf{B}$ and $\hat{\mathbf{B}}$, the KL divergence measure is defined as

$$\mathrm{KL}(\mathbf{B}|\hat{\mathbf{B}}) = \sum_{i,j}(b_{i,j}\log(b_{i,j}/\hat{b}_{i,j}) - b_{i,j} + \hat{b}_{i,j}), \tag{B.3}$$

where $b_{i,j}$ and $\hat{b}_{i,j}$ denote the elements from the i^{th} row and j^{th} column of the matrices $\mathbf{B}$ and $\hat{\mathbf{B}}$, respectively. Using speech basis matrix training as an example, the cost function of the KL divergence-based NMF for training $\overline{\mathbf{W}}$ can be written as

$$(\overline{\mathbf{W}}, \overline{\mathbf{H}}) = \underset{\overline{\mathbf{W}}, \overline{\mathbf{H}}}{\arg\min}\ \mathrm{KL}(\mathbf{S}_N|\overline{\mathbf{W}} \times \overline{\mathbf{H}}). \tag{B.4}$$

Noise basis matrix training is similar to speech basis matrix training. In [34], it is derived that $\overline{\mathbf{W}}$ and $\overline{\mathbf{H}}$ can be obtained iteratively using the following multiplicative update rules:

$$\overline{\mathbf{W}} \leftarrow \overline{\mathbf{W}} \odot \frac{\dfrac{\mathbf{S}_N}{\overline{\mathbf{W}} \times \overline{\mathbf{H}}}\overline{\mathbf{H}}^T}{\mathbf{1}\overline{\mathbf{H}}^T}, \tag{B.5}$$

$$\overline{\mathbf{H}} \leftarrow \overline{\mathbf{H}} \odot \frac{\overline{\mathbf{W}}^T\dfrac{\mathbf{S}_N}{\overline{\mathbf{W}} \times \overline{\mathbf{H}}}}{\overline{\mathbf{W}}^T\mathbf{1}}, \tag{B.6}$$

where $\odot$ and all divisions are element-wise multiplication and division operations, respectively, and $\mathbf{1}$ is a matrix of ones with the same size as $\mathbf{S}_N$. In the enhancement stage, the noisy speech basis matrix $\mathbf{W}$ can be constructed by concatenating the speech and noise basis matrices, $\mathbf{W} = [\overline{\mathbf{W}}, \ddot{\mathbf{W}}]$. The activation matrix $\mathbf{H}$ of the noisy speech can be estimated iteratively by replacing $\mathbf{S}_N$, $\overline{\mathbf{W}}$, and $\overline{\mathbf{H}}$ in (B.6) with $\mathbf{Y}_N$, $\mathbf{W}$, and $\mathbf{H}$, respectively. The enhanced signal can be obtained using various algorithms [36, 37, 44, 45]. One popular approach is to use the following Wiener-filter-like spectral gain $\mathbf{g}_n^{\mathrm{NMF}}$ function:

$$\mathbf{g}_n^{\mathrm{NMF}} = \frac{\overline{\mathbf{W}}\,\overline{\mathbf{h}}_n}{\overline{\mathbf{W}}\,\overline{\mathbf{h}}_n + \ddot{\mathbf{W}}\,\ddot{\mathbf{h}}_n}, \tag{B.7}$$

$$\begin{aligned}\mathbf{h}_n &= \left[\overline{\mathbf{h}}_n^T, \ddot{\mathbf{h}}_n^T\right]^T \\ &= \underset{\mathbf{h}_n}{\arg\min}\ \mathrm{KL}(\mathbf{y}_n|\mathbf{W}\mathbf{h}_n),\end{aligned} \tag{B.8}$$

where (B.8) can be solved iteratively using (B.6). Apart from the gradient descent derivation of the MU update rules (B.5) and (B.6) presented in [34], it is further shown in [53] that the MU update rules can be derived from a statistical perspective. More specifically, the KL divergence-based NMF can be motivated from the following hierarchical statistical model:

$$\mathbf{S}_N = \sum_{k=1}^{K} \mathbf{C}(k), \tag{B.9}$$

$$c_{f,n}(k) \sim \mathcal{PO}(c_{f,n}(k); \overline{W}_{f,k}\overline{H}_{k,n}), \tag{B.10}$$

where $\mathcal{PO}(x;\lambda) = \dfrac{\lambda^x e^{-\lambda}}{\Gamma(x+1)}$ is the Poisson distribution, $\Gamma(x+1) = x!$ denotes the gamma function for positive integer x, K denotes the number of basis vectors, $\mathbf{C}(k)$ is the latent matrix, and $c_{f,n}(k)$ denotes the element of $\mathbf{C}(k)$ in the f^{th} row and n^{th} column. Note that $c_{f,n}(k)$ is assumed to have a Poisson distribution, which can only be used for discrete variables. However, in practice, this hierarchical statistical model is not limited to discrete variables because the gamma function for continuous variables can be used to replace the factorial calculation [53]. It has been shown in [53] that the iterative update of the parameters $\overline{\mathbf{H}}$ and $\overline{\mathbf{W}}$ using the EM algorithm is identical to the multiplicative update rules shown in (B.5) and (B.6).

One of the advantages of the classical NMF-based method for speech enhancement is that the computational efficient MU rules can be applied. However, the temporal dynamical aspects of speech and noise are not taken into account. To incorporate the temporal dynamical information of audio signals, the HMM model is used in [45] for source separation. However, the parameter update rules are computationally complex. Moreover, this method [45] can only perform the offline enhancement. In this paper, we propose an NMF-based speech enhancement algorithm using the HMM to take the temporal aspects of both the speech and noise into account. The proposed approach can achieve efficient parameter updates. Moreover, an online MMSE estimator for speech enhancement is derived. Although other methods also considered the temporal dynamical information for speech enhancement, such as simply stacking multiple frames to a vector [14, 54], using the DRNN [28], and non-negative matrix deconvolution [55], the high computational complexity and the large model size lead to a high storage complexity. In this paper, the proposed method can achieve a higher PESQ score than the referenced DNN-based method for unseen noise and also has a lower complexity than it.

3 HMM-based Signal Models with the KL Divergence

In this section, we present the details of proposed signal models, including the speech and noise signal models and the noisy signal model.

3.1 Speech and Noise Signal Models

In this work, the same signal model is used for both the clean speech and noise signals, so we will derive the equations using only the clean speech signal. Additionally, we use

the overbar ($\bar{\cdot}$) and double dots ($\ddot{\cdot}$) to represent the clean speech and noise, respectively. To consider the temporal dynamic information of the speech and noise, we use the HMM. Following the conditional independence property of the standard HMM [56], the likelihood function can be expressed as follows:

$$p(\mathbf{S}_N;\mathbf{\Phi}) = \sum_{\overline{\mathbf{x}}_\mathbf{N}} \prod_{n=1}^{N} p(\mathbf{s}_n|\overline{x}_n)p(\overline{x}_n|\overline{x}_{n-1}), \tag{B.11}$$

where $\overline{\mathbf{x}}_\mathbf{N} = [\overline{x}_1, \cdots, \overline{x}_n, \cdots, \overline{x}_N]^T$ is a collection of states, $\overline{x}_n \in \{1, 2, \cdots, \overline{J}\}$ denote the state at the n^{th} frame, and $\overline{J}$ denotes the total number of states. The function $p(\overline{x}_n|\overline{x}_{n-1})$ denotes the state transition probability from state $\overline{x}_{n-1}$ to $\overline{x}_n$ with $p(\overline{x}_1|\overline{x}_0)$ being the initial state probability. $p(\mathbf{S}_n|\overline{x}_n)$ is the state-conditioned likelihood function, and $\mathbf{\Phi}$ is a collection of modeling parameters. Next, we describe the state transition probability and the state-conditioned likelihood function, respectively, for the proposed signal model.

The state transition probability $p(\overline{x}_n|\overline{x}_{n-1})$: Following the standard HMM, we use a first-order Markov chain to model the state transition, that is,

$$p(\overline{x}_n|\overline{x}_{n-1}) = \prod_{i=1}^{\overline{J}} \prod_{j=1}^{\overline{J}} \overline{A}_{i,j}^{\,l(\overline{x}_n=j,\overline{x}_{n-1}=i)}, \tag{B.12}$$

$$p(\overline{x}_1|\overline{x}_0) = p(\overline{x}_1) = \prod_{j=1}^{\overline{J}} \overline{\pi}_j^{\,l(\overline{x}_1=j)}, \tag{B.13}$$

where $l(\cdot)$ denotes an indicator function, which is one when the logic expression in the parentheses is true and zero otherwise. In addition, $\overline{A}_{i,j}$ and $\overline{\pi}_j$ denote the transition probability from state i to state j and the initial probability for the first frame's state $\overline{x}_1$ being state j, respectively. Collecting all the initial and transition probabilities, we can write them into matrix forms, $\overline{\boldsymbol{\pi}} = [\overline{\pi}_1, \cdots, \overline{\pi}_j, \cdots, \overline{\pi}_{\overline{J}}]^T$ and $\overline{\mathbf{A}}$ with $\overline{A}_{i,j}$ being the element at the i^{th} row and j^{th} column. Therefore, the modeling parameters of the HMM can be expressed as $\mathbf{\Phi}_{\text{hmm}} = \{\overline{\mathbf{A}}, \overline{\boldsymbol{\pi}}, \overline{J}\}$. The modeling parameters $\overline{\mathbf{A}}$ and $\overline{\boldsymbol{\pi}}$ with a predefined $\overline{J}$ can be trained through the EM algorithm shown in the next section. In the experiments, we investigate the impact of the total number of states $\overline{J}$.

The state-conditioned likelihood function: Next, we present the proposed state-conditioned likelihood function. Motived by the good speech enhancement performance, the computationally efficient MU rule, and the equivalence between the gradient descent derivation and the EM algorithm for the KL divergence-based NMF, we propose to use the statistical model in (B.9) and (B.10) to build the state-conditioned likelihood function, that is,

$$\mathbf{s}_n = \sum_{k=1}^{\overline{K}} \overline{\mathbf{c}}_n(k), \tag{B.14}$$

$$p(\overline{\mathbf{c}}_n(k)|\overline{x}_n) = \prod_{f=1}^{F} \mathcal{PO}(\overline{c}_{f,n}(k); \overline{W}_{f,k}^{\overline{x}_n} \overline{H}_{k,n}^{\overline{x}_n}), \tag{B.15}$$

where $\overline{K}$ is the number of basis vectors, $\overline{\mathbf{c}}_n(k)$ contains the hidden variables, and $\overline{W}_{k,n}^{\overline{x}_n}$ and $\overline{H}_{k,n}^{\overline{x}_n}$ correspond to the elements of the basis and activation matrices, respectively.

By writing $\overline{\mathbf{c}}_n = [\overline{\mathbf{c}}_n(1)^T, \overline{\mathbf{c}}_n(2)^T, \cdots, \overline{\mathbf{c}}_n(\overline{K})^T]^T$ and integrating $\overline{\mathbf{c}}_n$, the state conditioned likelihood function can be written as

$$\begin{aligned} p(\mathbf{s}_n|\overline{x}_n) &= \int p(\mathbf{s}_n|\overline{\mathbf{c}}_n)p(\overline{\mathbf{c}}_n|\overline{x}_n)\, d\overline{\mathbf{c}}_n \\ &= \prod_{f=1}^{F} \mathcal{PO}(|S(f,n)|; \sum_{k=1}^{\overline{K}} \overline{W}_{f,k}^{\overline{x}_n} \overline{H}_{k,n}^{\overline{x}_n}), \end{aligned} \tag{B.16}$$

where we use the superposition property of the Poisson random variable [53]. Collecting the unknown parameters $\{\overline{W}_{f,k}^{\overline{x}_n}\}$ and $\{\overline{H}_{k,n}^{\overline{x}_n}\}$, we can write them into matrix forms, $\{\overline{\mathbf{W}}^j\}$ and $\{\overline{\mathbf{H}}^j\}$. Therefore, unlike the traditional NMF using only one basis matrix, the proposed model has $\overline{J}$ basis matrices to be trained. Each basis matrix is intended to capture a specific feature (e.g., a phoneme) of the speech signal. The modeling parameters of the proposed state-conditioned likelihood function can be expressed as $\mathbf{\Phi}_{\text{like}} = \{\{\overline{\mathbf{W}}^j\}, \{\overline{\mathbf{H}}^j\}, \overline{K}, \overline{J}\}$. The modeling parameters $\{\overline{\mathbf{W}}^j\}$ and $\{\overline{\mathbf{H}}^j\}$ with predefined $\overline{J}$ and $\overline{K}$ can be trained through the EM algorithm shown in the next section. In the experiments, we investigate the impact of the number of basis vectors $\overline{K}$ and $\overline{J}$. It will also be shown that a multiplicative update rule can be derived for the basis and activation matrices update of the proposed state-conditioned likelihood function.

To summarize, five types of parameters in the parameter set $\mathbf{\Phi}=\mathbf{\Phi}_{\text{hmm}} \cup \mathbf{\Phi}_{\text{like}}$ can be identified. They are the transition matrix $\overline{\mathbf{A}}$, initial state probabilities in $\overline{\boldsymbol{\pi}}$, basis matrices of different states $\{\overline{\mathbf{W}}^j\}$, activation matrices of different states $\{\overline{\mathbf{H}}^j\}$, and modeling parameters $\overline{K}$ and $\overline{J}$. In this paper, the modeling parameters $\overline{K}$ and $\overline{J}$ are predefined, the activation matrices $\{\overline{\mathbf{H}}^j\}$ are estimated by online speech enhancement, and the other three types of parameters are obtained using offline learning.

3.2 Noisy Speech Model

Based on the proposed clean speech and noise signal models (B.1) and (B.2), the noisy speech model can be defined. We assume that there are a total of $\ddot{J}$ hidden states for the noise, and the hidden state of the noise is $\ddot{x}_n (\ddot{x}_n \in \{1, 2, \cdots, \ddot{J}\})$. The notations $\ddot{\boldsymbol{\pi}}$ and $\ddot{\mathbf{A}}$ correspond to the initial state probability and transition probability matrix of the noise. Thus, there are a total of $\overline{J} \times \ddot{J}$ hidden states for the noisy speech. Each composite state consists of a pair of states of clean speech $\overline{x}_n$ and noise $\ddot{x}_n$. Thus, if we list the state space for a noisy signal, we have $(\overline{x}_n = 1, \ddot{x}_n = 1), (\overline{x}_n = 1, \ddot{x}_n = 2), \cdots, (\overline{x}_n = 1, \ddot{x}_n = \ddot{J}); (\overline{x}_n = 2, \ddot{x}_n = 1), (\overline{x}_n = 2, \ddot{x}_n = 2), \cdots, (\overline{x}_n = 2, \ddot{x}_n = \ddot{J}); \cdots; (\overline{x}_n = \overline{J}, \ddot{x}_n = 1), (\overline{x}_n = \overline{J}, \ddot{x}_n = 2), \cdots, (\overline{x}_n = \overline{J}, \ddot{x}_n = \ddot{J})$. Moreover, the initial state and transition probability matrices of the noisy speech can be expressed as $\overline{\boldsymbol{\pi}} \otimes \ddot{\boldsymbol{\pi}}$ and $\overline{\mathbf{A}} \otimes \ddot{\mathbf{A}}$, where $\otimes$ denotes the Kronecker product. Finally, the state conditioned likelihood function of the noisy speech can be written as follows:

$$\begin{aligned} &p(\mathbf{y}_n|\overline{x}_n, \ddot{x}_n) = \\ &\prod_{f=1}^{F} \mathcal{PO}(|(Y(f,n)|; \sum_{k=1}^{\overline{K}} \overline{W}_{f,k}^{\overline{x}_n} \overline{H}_{k,n}^{\overline{x}_n} + \sum_{k=1}^{\ddot{K}} \ddot{W}_{f,k}^{\ddot{x}_n} \ddot{H}_{k,n}^{\ddot{x}_n}), \end{aligned} \tag{B.17}$$

where $\ddot{K}$, $\{\ddot{W}_{f,k}^{\ddot{x}_n}\}$, and $\{\ddot{H}_{f,k}^{\ddot{x}_n}\}$ represent the number of basis vectors, elements of the basis matrices, and the activation matrices for the noise, respectively. We can write

$\{\ddot{W}_{f,k}^{\ddot{x}_n}\}$ and $\{\ddot{H}_{k,n}^{\ddot{x}_n}\}$ into matrix forms as $\{\ddot{\mathbf{W}}^j\}$ and $\{\ddot{\mathbf{H}}^j\}$. Note that we also used the superposition property of Poisson random variables to obtain (B.17).

4 Methods

4.1 Offline NMF-HMM-based Parameter Learning

In the offline training stage, the objective is to find the parameter set $\mathbf{\Phi}$ that maximizes the likelihood function (B.11). In general, the EM algorithm [56] can be used to address this problem. Because we use the same model for the speech and noise, here we use the clean speech as an example to illustrate the offline parameter learning process. First, we define the complete data set $(\mathbf{S}_N, \overline{\mathbf{x}}_N, \overline{\mathbf{C}}_N)$, where $\overline{\mathbf{C}}_N = [\overline{\mathbf{c}}_1, \overline{\mathbf{c}}_2, \cdots, \overline{\mathbf{c}}_N]$. Thus, using the conditional independence property, the complete data likelihood function can be written as

$$p(\mathbf{S}_N, \overline{\mathbf{x}}_N, \overline{\mathbf{C}}_N) = \prod_{n=1}^{N} p(\mathbf{s}_n|\overline{\mathbf{c}}_n)p(\overline{\mathbf{c}}_n|\overline{x}_n)p(\overline{x}_n|\overline{x}_{n-1}). \tag{B.18}$$

Next, we show how the parameter set can be obtained iteratively using the EM algorithm. Moreover, we propose an acceleration strategy to lower the computational and memory complexities. The traditional MU update algorithm for the KL divergence-based NMF can be seen as a special case of the proposed algorithm.

Expectation step: We first calculate the posterior state probability and the joint posterior probability, which can be written as

$$q(\overline{x}_n) = p(\overline{x}_n|\mathbf{S}_N; \mathbf{\Phi}^{i-1}), \tag{B.19}$$

$$q(\overline{x}_n, \overline{x}_{n-1}) = p(\overline{x}_n, \overline{x}_{n-1}|\mathbf{S}_N; \mathbf{\Phi}^{i-1}), \tag{B.20}$$

where i is the iteration number. The calculation of (B.19) and (B.20) can be performed using the forward-backward algorithm [56]. Apart from this, we also need to evaluate the posterior expectation $\mathbb{E}_{\overline{\mathbf{c}}_n|\mathbf{S}_N,\overline{x}_n;\mathbf{\Phi}^{i-1}}(\overline{\mathbf{c}}_n)$, which will be used in the maximization step. By using the Bayes rule and the conditional independence property of the proposed model, we have

$$q(\overline{\mathbf{c}}_n|\overline{x}_n) = p(\overline{\mathbf{c}}_n|\mathbf{S}_N, \overline{x}_n; \mathbf{\Phi}^{i-1}) = \frac{p(\mathbf{s}_n|\overline{\mathbf{c}}_n)p(\overline{\mathbf{c}}_n|\overline{x}_n)}{p(\mathbf{S}_N, \overline{x}_n)}. \tag{B.21}$$

Combining (B.14) and (B.15) and following the derivation in [53], we have

$$\begin{aligned} &q(\overline{\mathbf{c}}_n|\overline{x}_n) = \\ &\prod_{f=1}^{F} \mathcal{M}(\overline{c}_{f,n}(1), \cdots, \overline{c}_{f,n}(\overline{K}); |S(f,n)|, \\ &p_{f,n}^{\overline{x}_n}(1), \cdots, p_{f,n}^{\overline{x}_n}(\overline{K})), \end{aligned} \tag{B.22}$$

where $\mathcal{M}(\cdot)$ denotes the multinomial distribution and

$$p_{f,n}^{\overline{x}_n}(k) = \frac{\overline{W}_{f,k}^{\overline{x}_n}\overline{H}_{k,n}^{\overline{x}_n}}{\sum_{l=1}^{\overline{K}} \overline{W}_{f,l}^{\overline{x}_n}\overline{H}_{l,n}^{\overline{x}_n}}. \tag{B.23}$$

Using the properties of the multinomial distribution, the mean can be written as

$$\mathbb{E}(\overline{c}_{f,n}(k)|\mathbf{S}_N, \overline{x}_n) = |S(f,n)|\frac{\overline{W}_{f,k}^{\overline{x}_n}\overline{H}_{k,n}^{\overline{x}_n}}{\sum_{l=1}^{K}\overline{W}_{f,l}^{\overline{x}_n}\overline{H}_{l,n}^{\overline{x}_n}}. \tag{B.24}$$

Maximization step: In this step, our objective is to find parameters to maximize the expectation of the logarithm of the complete data likelihood, that is,

$$\mathbf{\Phi}^i = \arg\max_{\mathbf{\Phi}} \mathbb{E}_{\overline{\mathbf{x}}_N, \overline{\mathbf{C}}_N|\mathbf{S}_N;\mathbf{\Phi}^{i-1}}[\log p(\mathbf{S}_N, \overline{\mathbf{x}}_N, \overline{\mathbf{C}}_N)]. \tag{B.25}$$

The estimators for $\overline{\mathbf{A}}$ and $\overline{\boldsymbol{\pi}}$ are the same as the traditional HMM [56]. For completeness, the results are shown below:

$$\overline{\pi}_j = \frac{q(\overline{x}_1 = j)}{\sum_{o=1}^{\overline{J}} q(\overline{x}_1 = o)}, \tag{B.26}$$

$$\overline{A}_{o,j} = \frac{\sum_{n=2}^{\overline{N}} q(\overline{x}_n = j, \overline{x}_{n-1} = o)}{\sum_{j=1}^{\overline{J}}\sum_{n=2}^{\overline{N}} q(\overline{x}_n = j, \overline{x}_{n-1} = o)}, \tag{B.27}$$

where $1 \leq o, j \leq \overline{J}$. The estimated basis and activation matrices can be derived by setting the derivatives of (B.25) to zeros, and we can obtain

$$W_{f,k}^j = \frac{\sum_{n=1}^{N} q(\overline{x}_n = j)\mathbb{E}(\overline{c}_{f,n}(k)|\mathbf{S}_N, \overline{x}_n = j)}{\sum_{n=1}^{N} q(\overline{x}_n = j)H_{k,n}^j}, \tag{B.28}$$

$$H_{k,n}^j = \frac{\sum_{f=1}^{F} \mathbb{E}(\overline{c}_{f,n}(k)|\mathbf{S}_N, \overline{x}_n = j)}{\sum_{f=1}^{F} W_{f,k}^j}. \tag{B.29}$$

Acceleration strategy: Although we can directly use the above EM algorithm to update the parameter set, saving the conditional expectation of $\overline{c}_{f,n}(k)$ in (B.24) requires a great deal of memory. Like [53], we substitute (B.24) into (B.28) and (B.29) and can obtain:

$$W_{f,k}^j \leftarrow \frac{\sum_{n=1}^{N} q(\overline{x}_n = j)\frac{|S(f,n)|\overline{H}_{k,n}^j}{\sum_{l=1}^{\overline{K}}\overline{W}_{f,l}^j\overline{H}_{l,n}^j}}{\sum_{n=1}^{N} q(\overline{x}_n = j)H_{k,n}^j}, \tag{B.30}$$

$$H_{k,n}^j \leftarrow \frac{\sum_{f=1}^{F}\frac{\overline{W}_{f,k}^j|S(f,n)|}{\sum_{l=1}^{K}\overline{W}_{f,l}^j\overline{H}_{l,n}^j}}{\sum_{f=1}^{F} H_{k,n}^j}. \tag{B.31}$$

We can further write (B.30) and (B.31) in matrix forms:

$$\overline{\mathbf{W}}^j \leftarrow \overline{\mathbf{W}}^j \odot \frac{\frac{\mathbf{S}_N}{\overline{\mathbf{W}}^j\overline{\mathbf{H}}^j}\mathbf{\Lambda}(j)(\overline{\mathbf{H}}^j)^T}{\mathbf{1}\mathbf{\Lambda}(j)(\overline{\mathbf{H}}^j)^T}, \tag{B.32}$$

$$\overline{\mathbf{H}}^j \leftarrow \overline{\mathbf{H}}^j \odot \frac{(\overline{\mathbf{W}}^j)^T\frac{\mathbf{S}_N}{\overline{\mathbf{W}}^j\overline{\mathbf{H}}^j}}{(\overline{\mathbf{W}}^j)^T\mathbf{1}}, \tag{B.33}$$

Algorithm 1: Offline NMF-HMM-based parameter learning

1: Randomly initiate $\overline{\mathbf{W}}^j$ and $\overline{\mathbf{H}}^j, \overline{j} \in \{1, 2, \cdots, \overline{J}\}$
2: **for** $i = 1, 2, 3, \cdots, I$ **do**
Expectation step:
3: Calculate $p(\mathbf{s}_n|\overline{x}_n)$, $1 \leq n \leq N$ based on (A.14)
4: Obtain (B.19) and (B.20) using the forward-backward algorithm [56]
Maximization step:
5: Re-estimate $\overline{\boldsymbol{\pi}}$ and $\overline{\mathbf{A}}$ based on (B.26) and (B.27)
6: Re-estimate $\overline{\mathbf{W}}^j$ and $\overline{\mathbf{H}}^j$ based on (B.32) and (B.33)
7: **end for**

where $\boldsymbol{\Lambda}(j) = \mathrm{diag}(q(\overline{x}_1 = j), q(\overline{x}_2 = j), \cdots, q(\overline{x}_N = j))$. By using the proposed acceleration strategy, the computing and saving of the conditional expectation of $\overline{c}_{f,n}(k)$ in (B.24) is not required. Moreover, the multiplicative update rules for the basis and activation matrices can be obtained, leading to fast computing. In the other word, there are more than one basis and active matrices to be estimated in the proposed algorithm. Using acceleration strategy, the different basis and active matrices can be simultaneously estimated. We do not need to estimate them one by one. This reduces the time complexity. Comparing the update rules of the proposed method (B.32), (B.33) with the traditional NMF-based method (B.5), (B.6), the difference is that the basis vectors update rule (B.32) for the proposed method takes the posterior state information $\boldsymbol{\Lambda}(j)$ into account. In fact, if the number of the state is set to one (i.e., $\overline{J} = 1$), the proposed training method is identical to the traditional KL divergence-based NMF approach. Thus, the traditional NMF can be seen as a special case of the proposed algorithm. The entire flow of the offline parameter learning is shown in Algorithm 1. Note that, for stability reasons, each column of $\overline{\mathbf{W}}^j$ is normalized to have a unit norm during training.

4.2 Online Speech Enhancement Using the MMSE Estimator

MMSE Estimator for the NMF-HMM

In this section, we provide a detailed derivation for the proposed MMSE-based online speech enhancement algorithm in the proposed NMF-HMM model. Our objective is to obtain the MMSE estimate of the desired clean speech signal from noisy observation:

$$\hat{\mathbf{s}}_n = \mathbb{E}_{\mathbf{s}_n|\mathbf{Y}_n}(\mathbf{s}_n) = \int \mathbf{s}_n p(\mathbf{s}_n|\mathbf{Y}_n)\, d\mathbf{s}_n. \tag{B.34}$$

In (B.34), the posterior probability $p(\mathbf{s}_n|\mathbf{Y}_n)$ can be derived as

$$\begin{aligned} p(\mathbf{s}_n|\mathbf{Y}_n) &= \frac{p(\mathbf{s}_n, \mathbf{y}_n|\mathbf{Y}_{n-1})}{p(\mathbf{y}_n|\mathbf{Y}_{n-1})} \\ &= \frac{\sum_{\overline{x}_n, \ddot{x}_n} p(\mathbf{s}_n, \mathbf{y}_n|\overline{x}_n, \ddot{x}_n) p(\overline{x}_n, \ddot{x}_n|\mathbf{Y}_{n-1})}{p(\mathbf{y}_n|\mathbf{Y}_{n-1})}, \end{aligned} \tag{B.35}$$

where we use the conditional independence property of the HMM. The term $p(\overline{x}_n, \ddot{x}_n|\mathbf{Y}_{n-1})$ in (B.35) can be expressed as

$$\begin{aligned} &p(\overline{x}_n, \ddot{x}_n|\mathbf{Y}_{n-1}) \\ &= \sum_{\overline{x}_{n-1}, \ddot{x}_{n-1}} p(\overline{x}_n, \ddot{x}_n|\overline{x}_{n-1}, \ddot{x}_{n-1}) p(\overline{x}_{n-1}, \ddot{x}_{n-1}|\mathbf{Y}_{n-1}), \end{aligned} \tag{B.36}$$

where the first term after the summation is the state transition probability for a noisy signal, and the second term is the forward probability that can be acquired using the well-known forward algorithm [56]. By applying the Bayes rule, the term $p(\mathbf{s}_n, \mathbf{y}_n|\overline{x}_n, \ddot{x}_n)$ in (B.35) can be further written as

$$p(\mathbf{s}_n, \mathbf{y}_n|\overline{x}_n, \ddot{x}_n) = p(\mathbf{s}_n|\mathbf{y}_n, \overline{x}_n, \ddot{x}_n) p(\mathbf{y}_n|\overline{x}_n, \ddot{x}_n). \tag{B.37}$$

Substituting (B.37) for (B.35), the posterior probability can be re-written as

$$p(\mathbf{s}_n|\mathbf{Y}_n) = \sum_{\overline{x}_{n-1}, \ddot{x}_{n-1}} \omega_{\overline{x}_n, \ddot{x}_n} p(\mathbf{s}_n|\mathbf{y}_n, \overline{x}_n, \ddot{x}_n), \tag{B.38}$$

where the weight $0 \leq \omega_{\overline{x}_n, \ddot{x}_n} \leq 1$ is defined as

$$\omega_{\overline{x}_n, \ddot{x}_n} = \frac{p(\mathbf{y}_n|\overline{x}_n, \ddot{x}_n) p(\overline{x}_n, \ddot{x}_n|\mathbf{Y}_{n-1})}{\sum_{\overline{x}_n, \ddot{x}_n} p(\mathbf{y}_n|\overline{x}_n, \ddot{x}_n) p(\overline{x}_n, \ddot{x}_n|\mathbf{Y}_{n-1})}. \tag{B.39}$$

Thus, by combining (B.34) and (B.38), the proposed HMM-based MMSE estimator can be expressed as

$$\hat{\mathbf{s}}_n = \sum_{\overline{x}_n, \ddot{x}_n} \omega_{\overline{x}_n, \ddot{x}_n} \int \mathbf{s}_n p(\mathbf{s}_n|\mathbf{y}_n, \overline{x}_n, \ddot{x}_n)\, d\mathbf{s}_n. \tag{B.40}$$

Instead of obtaining the posterior probability density function (PDF) $p(\mathbf{s}_n|\mathbf{y}_n, \overline{x}_n, \ddot{x}_n)$ directly, we derive the formula for the joint posterior PDF of the clean speech and noise first, that is,

$$\begin{aligned} &p(\mathbf{s}_n, \mathbf{m}_n|\mathbf{y}_n, \overline{x}_n, \ddot{x}_n) \\ &= \frac{p(\mathbf{y}_n|\mathbf{s}_n, \mathbf{m}_n) p(\mathbf{s}_n, \mathbf{m}_n|\overline{x}_n, \ddot{x}_n)}{p(\mathbf{y}_n|\overline{x}_n, \ddot{x}_n)} \\ &= \frac{p(\mathbf{y}_n|\mathbf{s}_n, \mathbf{m}_n) p(\mathbf{s}_n|\overline{x}_n) p(\mathbf{m}_n|\ddot{x}_n)}{p(\mathbf{y}_n|\overline{x}_n, \ddot{x}_n)}. \end{aligned} \tag{B.41}$$

By using (B.1), we can express the likelihood function $p(\mathbf{y}_n|\mathbf{s}_n, \mathbf{m}_n)$ as $p(\mathbf{y}_n|\mathbf{s}_n, \mathbf{m}_n) = \delta(\mathbf{y}_n - \mathbf{s}_n - \mathbf{m}_n)$, where $\delta(\cdot)$ denotes the Dirac delta function, which is defined by $\delta(0) = +\infty$, and $\delta(x) = 0$ when $x \neq 0$. Furthermore, $\int_{-\infty}^{+\infty} \delta(x)\, dx = 1$. The prior probability $p(\mathbf{s}_n|\overline{x}_n)$ and $p(\mathbf{m}_n|\ddot{x}_n)$ can be estimated by using (B.16). Following the derivation in [53], we can verify that the joint posterior PDF can be expressed in terms of the multinomial distribution as

$$\begin{aligned} &p(\mathbf{s}_n, \mathbf{m}_n|\mathbf{y}_n, \overline{x}_n, \ddot{x}_n) = \\ &\prod_{f=1}^{F} \mathcal{M}(|S(f,n)|, |M(f,n)|; \\ &|Y(f,n)|, p_{f,n}(\overline{x}_n, \ddot{x}_n), q_{f,n}(\overline{x}_n, \ddot{x}_n)), \end{aligned} \tag{B.42}$$

where $p_{f,n}(\overline{x}_n, \ddot{x}_n)$ and $q_{f,n}(\overline{x}_n, \ddot{x}_n)$ are defined as

$$p_{f,n}(\overline{x}_n, \ddot{x}_n) = \frac{\sum_{k=1}^{\overline{K}} \overline{W}_{f,k}^{\overline{x}_n} \overline{H}_{k,n}^{\overline{x}_n}}{\sum_{k=1}^{\overline{K}} \overline{W}_{f,k}^{\overline{x}_n} \overline{H}_{k,n}^{\overline{x}_n} + \sum_{k=1}^{\ddot{K}} \ddot{W}_{f,k}^{\ddot{x}_n} \ddot{H}_{k,n}^{\ddot{x}_n}}, \tag{B.43}$$

where $q_{f,n}(\overline{x}_n, \ddot{x}_n) = 1 - p_{f,n}(\overline{x}_n, \ddot{x}_n)$. Therefore, the integral term in (B.40) can be expressed as

$$\begin{aligned} &\int \mathbf{s}_n p(\mathbf{s}_n|\mathbf{y}_n, \overline{x}_n, \ddot{x}_n)\, d\mathbf{s}_n \\ &= \int \mathbf{s}_n \int p(\mathbf{s}_n, \mathbf{m}_n|\mathbf{y}_n, \overline{x}_n, \ddot{x}_n)\, d\mathbf{m}_n\, d\mathbf{s}_n \\ &= \mathbf{y}_n \odot \mathbf{p}_n(\overline{x}_n, \ddot{x}_n), \end{aligned} \tag{B.44}$$

where $\mathbf{p}_n(\overline{x}_n, \ddot{x}_n) = [p_{1,n}(\overline{x}_n, \ddot{x}_n), \cdots, p_{F,n}(\overline{x}_n, \ddot{x}_n)]^T$, and we used the marginal mean property of the multinomial distribution. Combining (B.40) and (B.44), the MMSE estimator can be expressed as:

$$\hat{\mathbf{s}}_n = \mathbf{y}_n \odot \mathbf{g}_n, \tag{B.45}$$

$$\mathbf{g}_n = \sum_{\overline{x}_n, \ddot{x}_n} \omega_{\overline{x}_n, \ddot{x}_n} \mathbf{p}_n(\overline{x}_n, \ddot{x}_n), \tag{B.46}$$

where $\mathbf{g}_n$ can be viewed as the spectral gain vector for the proposed model. Comparing the proposed gain vector $\mathbf{g}_n$ with the traditional NMF-based gain vector [36], we find that the proposed gain vector is a weighted sum of each state's gain, which is in the Wiener filtering form as the traditional NMF gain (B.7).

Online Estimation of Activation Matrices

After obtaining the trained basis matrices $\overline{W}_{f,k}^{\overline{x}_n}$ and $\ddot{W}_{f,k}^{\ddot{x}_n}$ for both the clean speech and noise in the training stage, we need to obtain the online estimates of the activation parameters $\overline{H}_{f,k}^{\overline{x}_n}$ and $\ddot{H}_{f,k}^{\ddot{x}_n}$ to acquire the gain in (B.45) and (B.46). The activation matrices are estimated by maximizing the logarithm of the state-conditioned likelihood function (B.17), which is equivalent to

$$\mathbf{h}_n(\overline{x}_n, \ddot{x}_n) = \arg\min_{\mathbf{h}_n} \ \mathrm{KL}(\mathbf{y}_n|[\overline{\mathbf{W}}^{\overline{x}_n}, \ddot{\mathbf{W}}^{\ddot{x}_n}]\mathbf{h}_n), \tag{B.47}$$

$$\mathbf{h}_n(\overline{x}_n, \ddot{x}_n) = [\overline{\mathbf{h}}_n(\overline{x}_n, \ddot{x}_n)^T, \ddot{\mathbf{h}}_n(\overline{x}_n, \ddot{x}_n)^T]^T, \tag{B.48}$$

where the clean and noise activation matrices for the state $(\overline{x}_n, \ddot{x}_n)$ are defined as $\overline{\mathbf{h}}_n(\overline{x}_n, \ddot{x}_n) = [\overline{H}_{1,n}^{\overline{x}_n}, \overline{H}_{2,n}^{\overline{x}_n}, \cdots, H_{\overline{K},n}^{\overline{x}_n}]^T$ and $\ddot{\mathbf{h}}_n(\overline{x}_n, \ddot{x}_n) = [\ddot{H}_{1,n}^{\ddot{x}_n}, \ddot{H}_{2,n}^{\ddot{x}_n}, \cdots, H_{\ddot{K},n}^{\ddot{x}_n}]^T$. The activation matrix (B.48) can be obtained iteratively by using the multiplicative update rule in equation (B.6). Note that parallel computing can be used to reduce the time complexity when obtaining the activation matrices for different states. It can be readily shown that when $\overline{J} = \ddot{J} = 1$, the gain vectors for the proposed algorithm (B.46) and the standard NMF (9) are identical, that is, $\mathbf{g}_n = \mathbf{g}_n^{\mathrm{NMF}}$. The entire flow of the proposed MMSE-based online speech enhancement algorithm is illustrated by Algorithm 2.

Algorithm 2: MMSE-based online speech enhancement

1: Input magnitude spectrum: $\mathbf{Y}_n$
2: Initiate $\overline{\boldsymbol{\pi}} \otimes \ddot{\boldsymbol{\pi}}$ and $\overline{\mathbf{A}} \otimes \ddot{\mathbf{A}}$
3: **for** $n = 1, 2, 3, \cdots, N$ **do**
4: Initiate $\mathbf{h}_n(\overline{x}_n, \ddot{x}_n)$
5: Based on (B.6) and (B.48), obtain the iterative estimation $\mathbf{h}_n(\overline{x}_n, \ddot{x}_n)$
6: Calculate $p(\mathbf{y}_n|\overline{x}_n, \ddot{x}_n)$ based on (B.17)
7: Apply the forward algorithm and combine (B.36) and (B.39) to acquire $\omega_{\overline{x}_n, \ddot{x}_n}$
7: Obtain $\mathbf{p}_n(\overline{x}_n, \ddot{x}_n)$ using (B.43)
8: Calculate the spectral gain $\mathbf{g}_n$ using (B.46)
9: By equation(B.45), estimate the clean speech $\hat{\mathbf{s}}_n$
10: **end for**

5 Experimental Results and Discussion

In this section, we report on the investigation and evaluation of the proposed algorithm using various experiments. First, we investigated the effect of different parameter settings for the proposed model, that is, the number of states and basis vectors of clean speech and noise, respectively. Second, we compared the proposed NMF-HMM with other state-of-the-art speech enhancement methods to demonstrate the effectiveness of the proposed algorithm. In this work, the PESQ score [41], ranging from -0.5 to 4.5, was used to quantify the enhanced speech quality. The version of the PESQ model used was the International Telecommunication Union (ITU) standard P.862 [57]. The implementation code was provided by [2]. The STOI score [42], ranging from zero to one, was used to measure speech intelligibility.

5.1 Experimental Data Preparation

In this study, the proposed algorithm was evaluated using the Texas Instruments/Massachusetts Institute of Technology (TIMIT) database [58], 100 environmental noises [59], office noise [1], and the NoiseX-92 database [60]. During the training stage, all 4620 utterances from the TIMIT training database were used to train the proposed NMF-HMM model for clean speech. For the experiments in Section 5.2, the Babble, F16, Factory, and White noises from the NoiseX-92 database were used to train the NMF-HMM model. For the experiments in Section 5.2, 200 utterances from the TIMIT test set, including 1680 utterances, were randomly chosen to build the test database. Four types of noise were then added at four different SNR levels (-5, 0, 5, and 10 dB). The noise types of the testing set were the same as the training set, but there was no overlap between the signals in the two sets. In total, $200 \times 4 \times 4 = 3200$ utterances were used for the evaluation. For the experiments in Section 5.3, we conducted ex-

[1] https://www.youtube.com/watch?v=D7ZZp8XuUTE

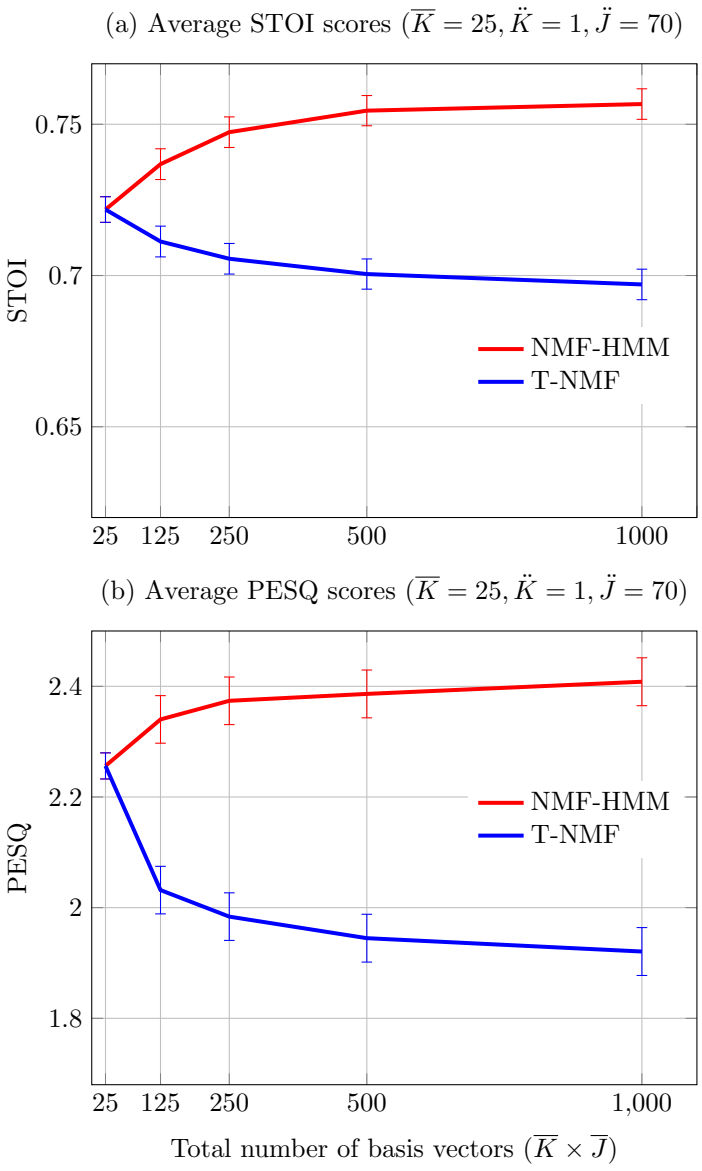


Fig. B.1: Performance of the NMF-HMM and T-NMF using different numbers of clean speech basis vectors.

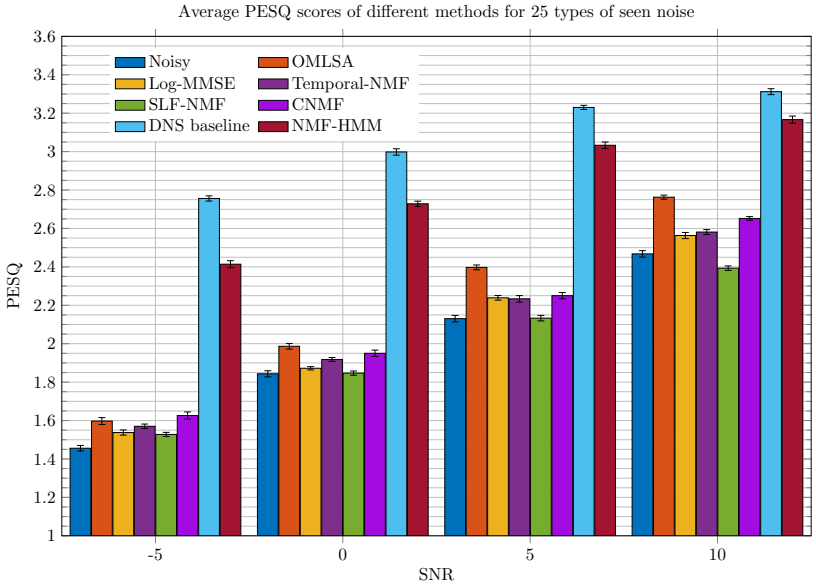


Fig. B.2: Average PESQ scores of different methods for 25 types of seen noise

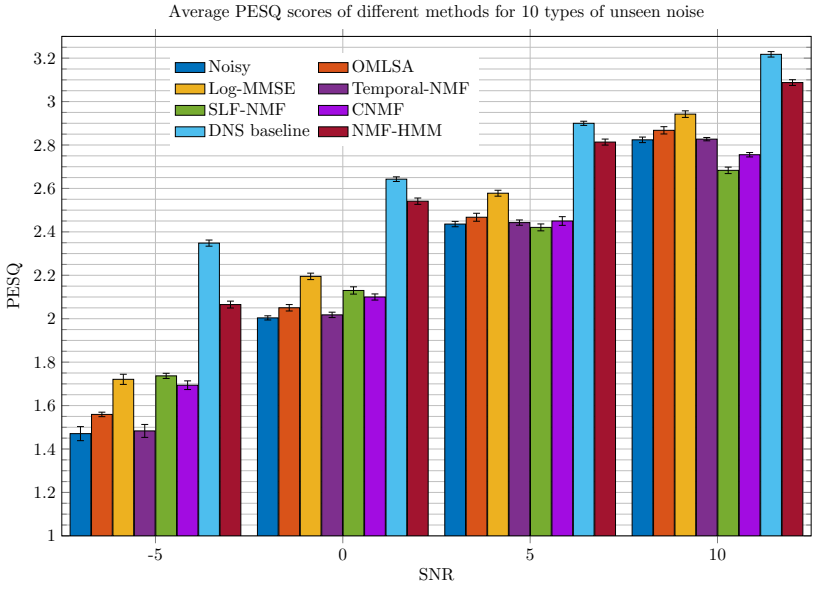


Fig. B.3: Average PESQ scores of different methods for 10 types of unseen noise

Table B.1: Average STOI scores (%) comparisons of different clean speech states and basis vectors ($\ddot{J} = 1, \ddot{K} = 70$)

Parameters	$\overline{K} = 5$	$\overline{K} = 10$
Noisy	69.14 (± 0.51)	
NMF-HMM, $\overline{J} = 1$(T-NMF)	65.00 (± 0.43)	69.29 (± 0.44)
NMF-HMM, $\overline{J} = 5$	68.66 (± 0.42)	71.93 (± 0.45)
NMF-HMM, $\overline{J} = 10$	69.71 (± 0.42)	72.74 (± 0.45)
NMF-HMM, $\overline{J} = 20$	71.14 (± 0.43)	73.51 (± 0.45)
NMF-HMM, $\overline{J} = 40$	71.81 (± 0.44)	73.66 (± 0.45)

Parameters	$\overline{K} = 25$	$\overline{K} = 50$
Noisy	69.14 (± 0.51)	
NMF-HMM, $\overline{J} = 1$(T-NMF)	72.71 (± 0.48)	73.32 (± 0.49)
NMF-HMM, $\overline{J} = 5$	73.94 (± 0.47)	74.02 (± 0.49)
NMF-HMM, $\overline{J} = 10$	74.39 (± 0.47)	74.37 (± 0.50)
NMF-HMM, $\overline{J} = 20$	74.76 (± 0.48)	74.87 (± 0.50)
NMF-HMM, $\overline{J} = 40$	75.00 (± 0.48)	74.73 (± 0.51)

tensive experiments; the Babble and F16 noises from the NoiseX-92 database and 90 environmental noises (N1–N90 in [59]) were used to train the NMF-HMM model for the noise dictionary. In the test stage, 200 utterances from the TIMIT test set, including 1680 utterances, were randomly chosen to build three test databases. The first test database included 10 unseen environmental noises from [59] (N91—N100). The second included unseen office noise, and the third test database was built from 25 seen environmental noises in [59] (N18—N43). In all three test databases, the noise was added at four different SNR levels (-5, 0, 5, and 10 dB). All the algorithms were evaluated using the same test dataset. In all experiments, the sound signals were down-sampled to 16 kHz. The frame length was set to 1024 samples (64 ms) with a frame shift of 512 samples (32 ms). The size of STFT was 1024 points with a Hanning window. Furthermore, the maximum number of iterations was set to 30 in the training stage and 15 in the online speech enhancement stage for the proposed NMF-HMM algorithm.

5.2 Analyses of the Number of States and Basis Vectors

As explained in Sections 3 and 4, four parameters needed to be pre-defined in our proposed NMF-HMM-based speech enhancement algorithm. These parameters were the number of states ($\overline{J}$ and $\ddot{J}$) and basis vectors ($\overline{K}$ and $\ddot{K}$) for the clean speech and noise. In this section, we report on the investigation of the effects of these parameters

Table B.2: Average PESQ scores (%) comparisons of different clean speech states and basis vectors ($\ddot{J} = 1, \ddot{K} = 70$)

Parameters	$\overline{K} = 5$	$\overline{K} = 10$
Noisy	2.02 (± 0.03)	
NMF-HMM, $\overline{J} = 1$(T-NMF)	2.12 (± 0.03)	2.18 (± 0.03)
NMF-HMM, $\overline{J} = 5$	2.27 (± 0.03)	2.31 (± 0.03)
NMF-HMM, $\overline{J} = 10$	2.31 (± 0.03)	2.35 (± 0.03)
NMF-HMM, $\overline{J} = 20$	2.36 (± 0.03)	2.39 (± 0.02)
NMF-HMM, $\overline{J} = 40$	2.38 (± 0.02)	2.41 (± 0.02)

Parameters	$\overline{K} = 25$	$\overline{K} = 50$
Noisy	2.02 (± 0.03)	
NMF-HMM, $\overline{J} = 1$(T-NMF)	2.21 (± 0.02)	2.18 (± 0.02)
NMF-HMM, $\overline{J} = 5$	2.32 (± 0.02)	2.29 (± 0.02)
NMF-HMM, $\overline{J} = 10$	2.35 (± 0.03)	2.30 (± 0.02)
NMF-HMM, $\overline{J} = 20$	2.36 (± 0.02)	2.32 (± 0.02)
NMF-HMM, $\overline{J} = 40$	2.39 (± 0.02)	2.33 (± 0.02)

in our proposed method and the choice of suitable parameters for the later experiments.

HMM States Analysis

First, before the states analysis, we want to indicate that using temporal dynamics can effectively help NMF obtain a better SE performance. To verify this point, we use the traditional NMF-based speech enhancement (T-NMF) [36] as reference method. T-NMF is a special case of NMF-HMM when $\overline{J} = 1$ and $\ddot{J} = 1$. T-NMF does not include the temporal dynamics information. The transition matrix A is a non-informational matrix in T-NMF. For a fair comparison, we keep that the total numbers of clean speech basis vectors ($\overline{K} \times \overline{J}$) for the NMF-HMM and T-NMF method [36] are the same. For the T-NMF, the number of clean speech basis vectors $\overline{K}$ is varied as 25, 125, 250, 500 and 1000. For the NMF-HMM, the $\overline{K}$ is fixed to 25 and $\overline{J}$ is varied as 1, 5, 10, 20 and 40. The number of noise basis vectors for both the proposed NMF-HMM and T-NMF is fixed to 70, and the number of noise states for the NMF-HMM is fixed to 1. In this experiment, we use the average STOI and PESQ scores of 3200 utterances as the performance metrics. The experimental results are shown in Figure B.1. As can be seen, the T-NMF can achieve the best performance when $\overline{K} = 25$. However, its performance degraded with the increasing of number of basis vectors due to overfitting. By contrast, NMF-HMM achieves higher PESQ and STOI scores with an increasing number of the clean speech basis vectors by

Table B.3: Average STOI scores (%) comparisons of different noise states and basis vectors ($\overline{J} = 40, \overline{K} = 25$)

Parameters	$\ddot{K} = 10$	$\ddot{K} = 20$
Noisy	69.14 ($\pm$ 0.51)	
NMF-HMM, $\ddot{J} = 1$	74.51 ($\pm$ 0.51)	74.71 ($\pm$ 0.51)
NMF-HMM, $\ddot{J} = 2$	75.00 ($\pm$ 0.51)	75.30 ($\pm$ 0.50)
NMF-HMM, $\ddot{J} = 5$	75.44 ($\pm$ 0.51)	75.77 ($\pm$ 0.50)
NMF-HMM, $\ddot{J} = 10$	75.56 ($\pm$ 0.50)	76.11 ($\pm$ 0.49)

Parameters	$\ddot{K} = 40$	$\ddot{K} = 70$
Noisy	69.14 ($\pm$ 0.51)	
NMF-HMM, $\ddot{J} = 1$	75.03 ($\pm$ 0.49)	75.00 ($\pm$ 0.48)
NMF-HMM, $\ddot{J} = 2$	75.51 ($\pm$ 0.49)	75.33 ($\pm$ 0.47)
NMF-HMM, $\ddot{J} = 5$	76.05 ($\pm$ 0.47)	75.15 ($\pm$ 0.46)
NMF-HMM, $\ddot{J} = 10$	76.27 ($\pm$ 0.48)	75.70 ($\pm$ 0.46)

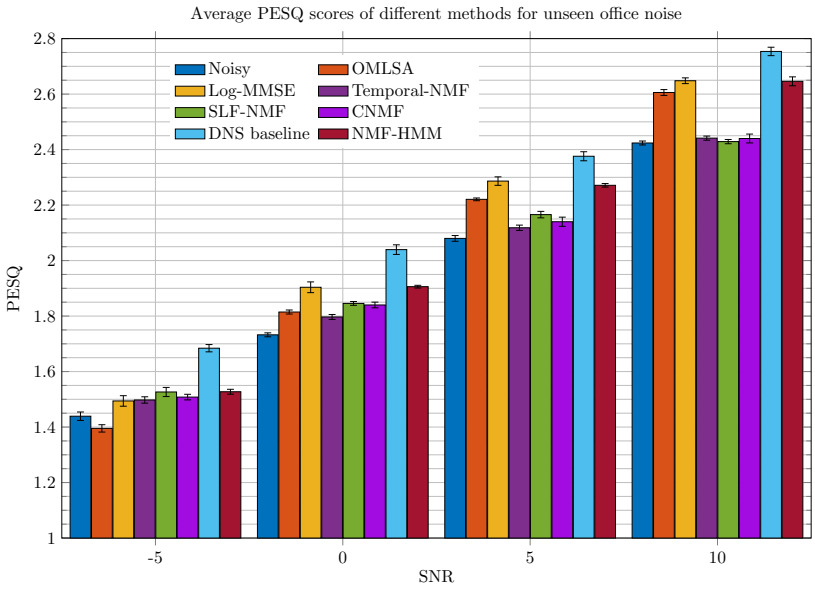


Fig. B.4: Average PESQ scores of different methods for unseen office noise

Table B.4: Average PESE scores (%) comparisons of different noise states and basis vectors ($\overline{J} = 40, \overline{K} = 25$)

Parameters	$\ddot{K} = 10$	$\ddot{K} = 20$
Noisy	2.02 (± 0.03)	
T-NMF, $\ddot{J} = 1$	2.28 (± 0.03)	2.31 (± 0.03)
NMF-HMM, $\ddot{J} = 2$	2.29 (± 0.03)	2.33 (± 0.04)
NMF-HMM, $\ddot{J} = 5$	2.31 (± 0.03)	2.34 (± 0.04)
NMF-HMM, $\ddot{J} = 10$	2.32 (± 0.03)	2.36 (± 0.03)

Parameters	$\ddot{K} = 40$	$\ddot{K} = 70$
Noisy	2.02 (± 0.03)	
T-NMF, $\ddot{J} = 1$	2.36 (± 0.02)	2.39 (± 0.02)
NMF-HMM, $\ddot{J} = 2$	2.37 (± 0.04)	2.40 (± 0.03)
NMF-HMM, $\ddot{J} = 5$	2.39 (± 0.03)	2.40 (± 0.03)
NMF-HMM, $\ddot{J} = 10$	2.40 (± 0.02)	2.41 (± 0.02)

taking the temporal dynamics into account using the HMM model, which indicates that temporal dynamics can improve the NMF's SE performance.

States and basis vector analysis for clean speech

Next, we investigated the effect of the number of clean speech states $\overline{J}$ and basis vector $\overline{K}$ to the proposed model. The number of noise states was set to one (i.e., $\ddot{J} = 1$) for the proposed NMF-HMM. The number of basis vectors for the noise was fixed to $\ddot{K} = 70$, respectively. The number of clean speech states was chosen as 1, 5, 10, 20, and 40. Additionally, the number of clean speech basis vector was chosen as 5, 10, 25, 50. The enhancement performance was evaluated by the PESQ and STOI scores.

Table B.1 and Table B.2 show the average STOI and PESQ score in different SNRs. It can be seen that if the number of basis vectors $\overline{K}$ is fixed, there is a higher PESQ and STOI score with the increasing of clean state $\overline{J}$. This indicated the benefits of using the temporal dynamics in NMF model. Additionally, if the clean state $\overline{J}$ is fixed, we can find that HMM can achieve the best speech enhancement performance when $\overline{K} = 25$. A higher $\overline{K}$ can lead to a worse speech enhancement performance due to overfitting. Therefore, based on these experimental results, we choose $\overline{J} = 40$ and $\overline{K} = 25$ to perform the following experiments.

Table B.5: Comparison of STOI scores (%) for various algorithms under -5dB SNRs using different types of noise.

Test Type	Method	-5dB
Unseen 10 types of noise	**Noisy**	76.97 ($\pm$ 1.45)
	Log-MMSE	75.86 ($\pm$ 1.54)
	OMLSA	75.88 ($\pm$ 1.52)
	Temporal-NMF	77.21 ($\pm$ 1.45)
	SLF-NMF	69.35 ($\pm$ 1.78)
	CNMF	77.12 ($\pm$ 1.51)
	NMF-HMM	78.58 ($\pm$ 1.34)
	DNS baseline	**81.84** ($\pm$ 1.36)
Unseen office noise	**Noisy**	49.91 ($\pm$ 1.33)
	Log-MMSE	46.46 ($\pm$ 1.50)
	OMLSA	44.97 ($\pm$ 1.52)
	Temporal-NMF	49.70 ($\pm$ 1.46)
	SLF-NMF	48.92 ($\pm$ 1.58)
	CNMF	48.43 ($\pm$ 1.47)
	NMF-HMM	50.06 ($\pm$ 1.72)
	DNS baseline	**54.22** ($\pm$ 1.49)
Seen 25 types of noise	**Noisy**	73.65 ($\pm$ 0.82)
	Log-MMSE	71.96 ($\pm$ 1.40)
	OMLSA	73.86 ($\pm$ 1.38)
	Temporal-NMF	75.76 ($\pm$ 1.34)
	SLF-NMF	65.76 ($\pm$ 1.58)
	CNMF	76.23 ($\pm$ 1.38)
	NMF-HMM	81.49 ($\pm$ 1.66)
	DNS baseline	**81.95** ($\pm$ 1.36)

States and Basis Vector Analysis for Noise

In this part, we evaluated the effect of noise states $\ddot{J}$ and basis vector $\ddot{K}$ to the proposed model. Here, the number of clean states and basis vectors was set to 40 and 25 ($\overline{J} = 40$, $\ddot{K} = 25$), respectively, which is based on the previous experimental results. The number of noise states was chosen as 1, 2, 5, and 10. In addition, the number of noise basis vector was chosen as 10, 20, 40, 70.

Table B.3 and Table B.4 show experimental results for the average STOI and PESQ score in different SNRs. We can find that the PESQ and STOI have an increasing trend with the increasing of noise state $\ddot{J}$ when the number of noise basis vectors $\ddot{K}$ is fixed. Moreover, if the $\ddot{J}$ is fixed, $\ddot{K} = 70$ can achieve the highest PESQ score but the STOI score is slightly lower than $\ddot{K} = 40$. Based on the experimental results, we select $\overline{J} = 40, \ddot{J} = 10, \overline{K} = 25, \ddot{K} = 40$ for the rest of the experiments because the model have the less parameters when $\ddot{K} = 40$. Furthermore, there is a higher STOI when $\ddot{K} = 40$ and the PESQ difference is not obvious between the $\ddot{K} = 40$ and $\ddot{K} = 70$.

Table B.6: Comparison of STOI scores (%) for various algorithms under 0dB SNRs using different types of noise.

Test Type	Method	0dB
Unseen 10 types of noise	**Noisy**	84.24 ($\pm$ 0.96)
	Log-MMSE	83.67 ($\pm$ 1.01)
	OMLSA	83.58 ($\pm$ 1.01)
	Temporal-NMF	84.39 ($\pm$ 0.96)
	SLF-NMF	77.01 ($\pm$ 1.28)
	CNMF	83.02 ($\pm$ 1.13)
	NMF-HMM	84.76 ($\pm$ 0.84)
	DNS baseline	**86.91** ($\pm$ 1.09)
Unseen office noise	**Noisy**	61.03 ($\pm$ 1.40)
	Log-MMSE	58.75 ($\pm$ 1.57)
	OMLSA	58.14 ($\pm$ 1.63)
	Temporal-NMF	61.79 ($\pm$ 1.47)
	SLF-NMF	60.84 ($\pm$ 1.54)
	CNMF	60.97 ($\pm$ 1.46)
	NMF-HMM	63.02 ($\pm$ 1.61)
	DNS baseline	**66.46** ($\pm$ 1.01)
Seen 25 types of noise	**Noisy**	81.36 ($\pm$ 1.03)
	Log-MMSE	80.13 ($\pm$ 1.20)
	OMLSA	81.58 ($\pm$ 1.18)
	Temporal-NMF	83.22 ($\pm$ 1.09)
	SLF-NMF	73.49 ($\pm$ 1.33)
	CNMF	84.12 ($\pm$ 1.11)
	NMF-HMM	87.02 ($\pm$ 1.35)
	DNS baseline	**87.34** ($\pm$ 1.15)

5.3 Overall Evaluation

In this section, we report on the comparison of the proposed NMF-HMM speech enhancement method with state-of-the-art speech enhancement methods. We chose the optimally-modified log-spectral amplitude (OM-LSA) method [61] with improved minima controlled recursive averaging (IMCRA) noise estimator [62]; variable span linear filters method [7] (SLF-NMF), which uses the parametric NMF [17] for estimating the statistics; Temporal-NMF [49]; Convolutive NMF (CNMF) [55, 63]; DNN [64]; and Log-MMSE [65] algorithm as the reference methods. For the SLF-NMF, the maximum SNR filter was applied, and the number of eigenvectors was set to one. The variable span linear filters reference code can be found in [7]. The codebook size of clean speech and noise was set to 64 and 8, respectively. The other SLF-NMF parameter settings were the same as NMF-HMM. For the temporal-NMF, all the parameter settings were the same as the work in [49], which ensured that the temporal-NMF could achieve

Table B.7: Comparison of STOI scores (%) for various algorithms under 5dB SNRs using different types of noise.

Test Type	Method	5dB
Unseen 10 types of noise	**Noisy**	90.07 ($\pm$0.68)
	Log-MMSE	89.72 ($\pm$0.70)
	OMLSA	89.51 ($\pm$0.72)
	Temporal-NMF	90.15 ($\pm$0.68)
	SLF-NMF	82.11 ($\pm$1.09)
	CNMF	86.01 ($\pm$1.02)
	NMF-HMM	88.39 ($\pm$0.58)
	DNS baseline	**91.44** ($\pm$0.75)
Unseen office noise	**Noisy**	72.80 ($\pm$1.27)
	Log-MMSE	71.09 ($\pm$1.40)
	OMLSA	71.52 ($\pm$1.44)
	Temporal-NMF	73.48 ($\pm$1.29)
	SLF-NMF	70.95 ($\pm$1.35)
	CNMF	71.45 ($\pm$1.12)
	NMF-HMM	74.56 ($\pm$1.32)
	DNS baseline	**77.58** ($\pm$0.89)
Seen 25 types of noise	**Noisy**	87.64 ($\pm$0.84)
	Log-MMSE	87.04 ($\pm$0.94)
	OMLSA	87.90 ($\pm$0.91)
	Temporal-NMF	89.03 ($\pm$0.88)
	SLF-NMF	79.06 ($\pm$1.18)
	CNMF	89.55 ($\pm$0.97)
	NMF-HMM	90.28 ($\pm$0.77)
	DNS baseline	**91.53** ($\pm$0.75)

the best speech enhancement performance. For the CNMF, the related settings were similar to the CNMF in [40]. For the DNN, we used the DNS-baseline [64] as the reference method, which is one of the state of the art speech enhancement algorithm. The OM-LSA and Log-MMSE were state-of-the-art unsupervised speech enhancement methods. while the SLF-NMF and temporal-NMF were state-of-the-art NMF-based speech enhancement methods. The temporal-NMF also considered the temporal information like our methods.

The performance of the NMF-HMM, DNN, Temporal-NMF, CNMF, SLF-NMF, Log-MMSE, and OM-LSA were evaluated using the test set. Figure B.2 shows the average PESQ scores with 95% confidence intervals of these algorithms for 25 types of seen noise. As can be seen, the SLF-NMF had the worst performance among these algorithms. Temporal-NMF and CNMF achieved a higher score than SLF-NMF, which indicated the benefits of temporal information for speech enhancement. Moreover, except for DNS baseline, the proposed NMF-HMM outperformed other enhancement algorithms in all the SNR scenarios. Furthermore, in low SNR scenarios (e.g., -5–5 dB), the average

Table B.8: Comparison of STOI scores (%) for various algorithms under 10dB SNRs using different types of noise.

Test Type	Method	10dB
Unseen 10 types of noise	**Noisy**	94.16 ($\pm$ 0.49)
	Log-MMSE	93.85 ($\pm$ 0.48)
	OMLSA	93.62 ($\pm$ 0.55)
	Temporal-NMF	94.19 ($\pm$ 0.49)
	SLF-NMF	85.72 ($\pm$ 0.94)
	CNMF	89.44 ($\pm$ 0.91)
	NMF-HMM	90.88 ($\pm$ 0.43)
	DNS baseline	**94.67** ($\pm$ 0.55)
Unseen office noise	**Noisy**	82.57 ($\pm$ 1.05)
	Log-MMSE	81.31 ($\pm$ 1.15)
	OMLSA	82.29 ($\pm$ 1.14)
	Temporal-NMF	83.05 ($\pm$ 1.05)
	SLF-NMF	79.21 ($\pm$ 1.12)
	CNMF	80.03 ($\pm$ 0.97)
	NMF-HMM	82.55 ($\pm$ 0.88)
	DNS baseline	**86.18** ($\pm$ 0.50)
Seen 25 types of noise	**Noisy**	92.48 ($\pm$ 0.60)
	Log-MMSE	92.08 ($\pm$ 0.68)
	OMLSA	92.45 ($\pm$ 0.66)
	Temporal-NMF	93.46 ($\pm$ 0.58)
	SLF-NMF	83.14 ($\pm$ 1.04)
	CNMF	91.06 ($\pm$ 0.62)
	NMF-HMM	91.84 ($\pm$ 0.51)
	DNS baseline	**94.77** ($\pm$ 0.53)

PESQ score improvement of the proposed NMF-HMM was larger than 0.5 against the other algorithms.

Figure B.3 and Figure B.4 show the PESQ result under an unseen noise environment, which indicates that NMF-HMM could always achieve a higher PESQ score than the reference methods at all four SNRs except for DNS baseline.

The results of the STOI scores with 95% confidence intervals for various algorithms are provided in Table B.5, B.6, B.7, and B.8. As can be seen, the Temporal-NMF, CNMF and NMF-HMM had higher STOI scores than SLF-NMF under three different test datasets, which illustrates the benefits of considering speech temporal information. In general, NMF-HMM achieved the highest STOI score, better than the referenced NMF-based methods (Temporal-NMF, CNMF and SLF-NMF) for seen and unseen noise. In addition, the DNS baseline achieved a better STOI score than NMF-HMM.

In general, for these non-DNN-based speech enhancement algorithm, proposed method can achieve the best speech enhancement performance. Moreover, DNS baseline can achieve the highest speech enhancement score. In the future work, we think

that a DNN-based strategy can be combine with proposed algorithm to improve to accuracy of basis vector estimation. As a result, our algorithm can achieve a better speech enhancement performance.

6 Conclusions

In this work, we proposed and analyzed an NMF-HMM-based speech enhancement algorithm that applies the sum of the Poisson distribution, leading to the KL divergence measure, as the observation model for each state of the HMM. The computationally efficient multiplicative update rule is used to conduct parameter updates during the training stage for this proposed method. Moreover, using the HMM, the temporal dynamic information of speech signals can be captured in this method. Furthermore, we detailed the derivation of the proposed NMF-HMM-based MMSE estimator to conduct online speech enhancement. Parallel computation can be applied for the proposed estimator, so we can effectively reduce the time complexity during the online speech enhancement stage. With experiments, a suitable number of state basis vectors for the proposed NMF-HMM were found. Our experimental results also indicated that the proposed algorithm could outperform state-of-the-art NMF-based and unsupervised speech enhancement methods. In the future work, a DNN-based strategy can be considered to improve the accuracy of basis vector estimation. As a result, our algorithm can achieve a better speech enhancement performance.

References

[1] Jinyu Li, Li Deng, Yifan Gong, and Reinhold Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 22, no. 4, pp. 745–777, 2014.

[2] Philipos C Loizou, *Speech enhancement: theory and practice*, CRC press, 2013.

[3] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Process. Lett.*, vol. 21, no. 1, pp. 65–68, 2013.

[4] Israel Cohen and Sharon Gannot, "Spectral enhancement methods," in *Springer Handbook of Speech Processing*, pp. 873–902. Springer, 2008.

[5] Steven Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 27, no. 2, pp. 113–120, 1979.

[6] Knud B Christensen, Mads G Christensen, Jesper B Boldt, and Fredrik Gran, "Experimental study of generalized subspace filters for the cocktail party situation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2016, pp. 420–424.

[7] Jesper Rindom Jensen, Jacob Benesty, and Mads Græsbøll Christensen, "Noise reduction with optimal variable span linear filters," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 24, no. 4, pp. 631–644, 2015.

[8] Yariv Ephraim and Harry L Van Trees, "A signal subspace approach for speech enhancement," *IEEE Trans. Speech Audio Process.*, vol. 3, no. 4, pp. 251–266, 1995.

[9] Firas Jabloun and Benoît Champagne, "Incorporating the human hearing properties in the signal subspace approach for speech enhancement," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 6, pp. 700–708, 2003.

[10] Jae Lim and Alan Oppenheim, "All-pole modeling of degraded speech," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 26, no. 3, pp. 197–210, 1978.

[11] Yariv Ephraim and David Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 32, no. 6, pp. 1109–1121, 1984.

[12] Yariv Ephraim and David Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 33, no. 2, pp. 443–445, 1985.

[13] Amir Hussain, Mohamed Chetouani, Stefano Squartini, Alessandro Bastari, and Francesco Piazza, "Nonlinear speech enhancement: An overview," in *Progress in nonlinear speech processing*, pp. 217–248. Springer, 2007.

[14] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 23, no. 1, pp. 7–19, 2014.

[15] Mathew Shaji Kavalekalam, Jesper Kjær Nielsen, Jesper Bünsow Boldt, and Mads Græsbøll Christensen, "Model-based speech enhancement for intelligibility improvement in binaural hearing aids," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 27, no. 1, pp. 99–113, 2018.

[16] Sriram Srinivasan, Jonas Samuelsson, and W Bastiaan Kleijn, "Codebook-based bayesian speech enhancement for nonstationary environments," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 15, no. 2, pp. 441–452, 2007.

[17] Mathew Shaji Kavalekalam, Jesper Kjær Nielsen, Liming Shi, Mads Græsbøll Christensen, and Jesper Boldt, "Online parametric NMF for speech enhancement," in *Proc. European Signal Processing Conf.*, 2018, pp. 2320–2324.

[18] Qi He, Feng Bao, and Changchun Bao, "Multiplicative update of auto-regressive gains for codebook-based speech enhancement," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 25, no. 3, pp. 457–468, 2016.

[19] David Y Zhao and W Bastiaan Kleijn, "HMM-based gain modeling for enhancement of speech in noise," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 15, no. 3, pp. 882–892, 2007.

[20] Feng Deng, Changchun Bao, and W Bastiaan Kleijn, "Sparse Hidden Markov Models for speech enhancement in non-stationary noise environments," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 23, no. 11, pp. 1973–1987, 2015.

[21] Yoshua Bengio et al., "Learning deep architectures for AI," *Foundations and trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[22] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[23] DeLiang Wang and Jitong Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 26, no. 10, pp. 1702–1726, 2018.

[24] Yuxuan Wang, Arun Narayanan, and DeLiang Wang, "On training targets for supervised speech separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 22, no. 12, pp. 1849–1858, 2014.

[25] Arun Narayanan and DeLiang Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2013, pp. 7092–7096.

[26] Se Rim Park and Jinwon Lee, "A fully convolutional neural network for speech enhancement," *arXiv preprint arXiv:1609.07132*, 2016.

[27] Henrik Jacobsson, "Rule extraction from recurrent neural networks: Ataxonomy and review," *Neural Comput.*, vol. 17, no. 6, pp. 1223–1263, 2005.

[28] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis, "Joint optimization of masks and deep recurrent neural networks for monaural source separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 23, no. 12, pp. 2136–2147, 2015.

[29] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Proc. Advances in Neural Inform. Process. Syst.*, 2014, pp. 2672–2680.

[30] Santiago Pascual, Antonio Bonafonte, and Joan Serrà, "Segan: Speech enhancement generative adversarial network," *Proc. Interspeech*, pp. 3642–3646, 2017.

[31] Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen, "Speech intelligibility potential of general and specialized deep neural network based speech enhancement systems," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 25, no. 1, pp. 153–167, 2016.

[32] Yang Xiang and Changchun Bao, "A parallel-data-free speech enhancement method using multi-objective learning cycle-consistent generative adversarial network," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 28, pp. 1826–1838, 2020.

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

[34] Daniel D Lee and H Sebastian Seung, "Algorithms for non-negative matrix factorization," in *Proc. Advances in Neural Inform. Process. Syst.*, 2001, pp. 556–562.

[35] Kazuki Shimada, Yoshiaki Bando, Masato Mimura, Katsutoshi Itoyama, Kazuyoshi Yoshii, and Tatsuya Kawahara, "Unsupervised speech enhancement based on multichannel nmf-informed beamforming for noise-robust automatic speech recognition," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 27, no. 5, pp. 960–971, 2019.

[36] Emad M Grais and Hakan Erdogan, "Single channel speech music separation using nonnegative matrix factorization and spectral masks," in *Int. Conf. Digital Signal Process.*, 2011, pp. 1–6.

[37] Kevin W Wilson, Bhiksha Raj, and Paris Smaragdis, "Regularized non-negative matrix factorization with temporal dependencies for speech denoising," in *Proc. Interspeech*, 2008.

[38] Shuai Nie, Shan Liang, Hao Li, XueLiang Zhang, ZhanLei Yang, Wen Ju Liu, and Li Ke Dong, "Exploiting spectro-temporal structures using NMF for DNN-based supervised speech separation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2016, pp. 469–473.

[39] Tae Gyoon Kang, Kisoo Kwon, Jong Won Shin, and Nam Soo Kim, "NMF-based target source separation using deep neural network," *IEEE Signal Process. Lett.*, vol. 22, no. 2, pp. 229–233, 2014.

[40] Shuai Nie, Shan Liang, Wenju Liu, Xueliang Zhang, and Jianhua Tao, "Deep learning based speech separation via nmf-style reconstructions," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 26, no. 11, pp. 2043–2055, 2018.

[41] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2001, vol. 2, pp. 749–752.

[42] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, 2011.

[43] Thanh T Vu, Benjamin Bigot, and Eng Siong Chng, "Combining non-negative matrix factorization and deep neural networks for speech enhancement and automatic speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2016, pp. 499–503.

[44] Nasser Mohammadiha, Paris Smaragdis, and Arne Leijon, "Supervised and unsupervised speech enhancement using nonnegative matrix factorization," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 21, no. 10, pp. 2140–2151, 2013.

[45] Gautham J Mysore, Paris Smaragdis, and Bhiksha Raj, "Non-negative Hidden Markov modeling of audio with application to source separation," in *International Conference on Latent Variable Analysis and Signal Separation*. Springer, 2010, pp. 140–148.

[46] Ziteng Wang, Xu Li, Xiaofei Wang, Qiang Fu, and Yonghong Yan, "A DNN-HMM approach to non-negative matrix factorization based speech enhancement.," in *Proc. Interspeech*, 2016, pp. 3763–3767.

[47] Yang Xiang, Liming Shi, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, and Mads Græsbøll Christensen, "An NMF-HMM speech enhancement method based on kullback-leibler divergence," in *Proc. Interspeech*, 2020, pp. 2667–2671.

[48] Yang Xiang, Liming Shi, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, and Mads Græsbøll Christensen, "A novel NMF-HMM speech enhancement algorithm based on poisson mixture model," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2021, pp. 721–725.

[49] Cédric Févotte, Jonathan Le Roux, and John R Hershey, "Non-negative dynamical system with application to speech and audio," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2013, pp. 3158–3162.

[50] Cédric Févotte, Nancy Bertin, and Jean-Louis Durrieu, "Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis," *Neural Comput.*, vol. 21, no. 3, pp. 793–830, 2009.

[51] Cédric Févotte and Jérôme Idier, "Algorithms for nonnegative matrix factorization with the β-divergence," *Neural Comput.*, vol. 23, no. 9, pp. 2421–2456, 2011.

[52] Derry FitzGerald, Matt Cranitch, and Eugene Coyle, "On the use of the beta divergence for musical source separation," 2009.

[53] Ali Taylan Cemgil, "Bayesian inference for nonnegative matrix factorisation models," *Computational intelligence and neuroscience*, vol. 2009, 2009.

[54] Deepak Baby, Jort F Gemmeke, Tuomas Virtanen, et al., "Exemplar-based speech enhancement for deep neural network based automatic speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2015, pp. 4485–4489.

[55] Paris Smaragdis, "Convolutive speech bases and their application to supervised speech separation," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 15, no. 1, pp. 1–12, 2006.

[56] Leonard E Baum, "An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes," *Inequalities*, vol. 3, no. 1, pp. 1–8, 1972.

[57] ITU-T Recommendation, "Perceptual evaluation of speech quality (pesq): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," *Rec. ITU-T P. 862*, 2001.

[58] John S Garofolo, Lori F Lamel, William M Fisher, Jonathan G Fiscus, and David S Pallett, "DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1," *NASA STI/Recon technical report n*, vol. 93, 1993.

[59] Guoning Hu and DeLiang Wang, "A tandem algorithm for pitch estimation and voiced speech segregation," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 18, no. 8, pp. 2067–2079, 2010.

[60] Andrew Varga and Herman JM Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 12, no. 3, pp. 247–251, 1993.

[61] Israel Cohen and Baruch Berdugo, "Speech enhancement for non-stationary noise environments," *Signal Process.*, vol. 81, no. 11, pp. 2403–2418, 2001.

[62] Israel Cohen, "Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 5, pp. 466–475, 2003.

[63] Paul D O'grady and Barak A Pearlmutter, "Discovering speech phones using convolutive non-negative matrix factorisation with a sparseness constraint," *Neurocomputing*, vol. 72, no. 1-3, pp. 88–101, 2008.

[64] Sebastian Braun and Ivan Tashev, "Data augmentation and loss normalization for deep noise suppression," in *International Conference on Speech and Computer*. Springer, 2020, pp. 79–86.

[65] Timo Gerkmann and Richard C Hendriks, "Unbiased MMSE-based noise power estimation with low complexity and low tracking delay," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 20, no. 4, pp. 1383–1393, 2011.

Paper C

A novel NMF-HMM speech enhancement algorithm based on Poisson mixture model

Yang Xiang, Liming Shi, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, Mads Græsbøll Christensen

The paper has been published in the
Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., 2021

The layout has been revised.

Abstract

In this paper, we propose a novel non-negative matrix factorization (NMF) and hidden Markov model (NMF-HMM) based speech enhancement algorithm, which employs a Poisson mixture model (PMM). Compared to the previously proposed NMF-HMM method, the new algorithm, termed PMM-NMF-HMM, uses the Poisson mixture distribution for the state conditional likelihood function for a HMM rather than the single Poisson distribution. This means that there are the more basis matrices that can be used to model the speech and noise signals, so more signal information can be captured by the resulting model. The proposed method is supervised and thus includes a training and an enhancement stage. It is shown that, in the training stage, the proposed method can be implemented efficiently using multiplicative update (MU) for the model parameters, much like the NMF-HMM algorithm. In the speech enhancement stage, which can be performed online, a novel PMM-NMF-HMM minimum mean-square error (MMSE) estimator is developed. The experimental results indicate that the PMM-NMF-HMM method can obtain higher short-time objective intelligibility (STOI) and perceptual evaluation of speech quality (PESQ) score than NMF-HMM. Additionally, the method also outperforms other state-of-the-art NMF-based supervised speech enhancement algorithms.

1 Introduction

In real-word environments,the quality and intelligibility of speech signal is often degraded due to the presence of background noise. To combat such noise, speech enhancement techniques have been developed. The main purpose of speech enhancement is to estimate the speech from the observed noisy speech while attenuating the background noise to improve the quality and intelligibility of the observed signal [1]. Monaural speech enhancement provides a cost-effective strategy to address this problem by utilizing recordings from a single microphone, and by combining it with beamforming it can be extended to multiple microphones. Speech enhancement has a wide rang of important applications, which include as automatic speech recognition (ASR) [2], teleconferencing, hearing-aids, and mobile communication.

During the past decades, many different speech enhancement strategies have been proposed for environments with additive noise (e.g., [3]). These methods can be roughly divided into supervised and unsupervised approaches. For the unsupervised algorithms, the spectral subtraction algorithm [4] is perhaps the simplest strategy to estimate the speech. Furthermore, the minimum mean-square error (MMSE) spectral amplitude estimator [5], the signal subspace method of [6] and the optimally-modified log-spectral amplitude (OM-LSA) method [7] combined with IMCRA noise estimator [8] are all effective strategies to estimate the speech. However, these methods cannot always achieve satisfactory speech enhancement performance in non-stationary noise environment because of inaccurate estimation of noise. Therefore, the supervised speech enhancement method have been proposed like NMF [9]. Among the supervised speech enhancement algorithms, the codebook-driven auto-regressive (AR) model based method [10], the auto-regressive hidden Markov model (ARHMM) method [11] and non-negative matrix factorization (NMF) based methods [12] are noteworthy methods. These algorithms can make good use of prior information about both speech and noise, and, as a result, they can often achieve better speech enhancement performance

than the unsupervised methods, particularly in non-stationary acoustic environments.

With the advances in computation power, increases in the availability of training data combined with advances in the theory and practice of neural networks [13], deep neural networks (DNNs) have become a feasible strategy for speech enhancement. In recent years, various network structures have been used for enhancement, such as feed-forward multilayer perceptron [14], fully convolutional neural network [15], deep recurrent neural networks [16], and generative adversarial networks [17]. These networks can be used to predict the different targets like the speech spectrum [18], ideal ratio mask [19] and time domain waveform [20]. However, the computational complexity, model size and power consumption of these methods may be problematic for some application.

As mentioned above, NMF is an effective speech enhancement method. In general, NMF can be combined with other models to achieve the better speech enhancement performance. For instance, the combination of NMF and DNN can help NMF better model the speech and noise characteristics [21] and improve the generalization ability of the method [22]. Moreover, the NMF can be also combined with HMM [23], which can capture the temporal information of both speech and noise. As a consequence, such methods can often outperform the traditional NMF-based speech enhancement methods [12].

In our previous work [24], we proposed a NMF-HMM-based speech enhancement algorithm. This method applies a single Poisson distribution as the likelihood function for the HMM, which cannot effectively model the speech and noise due to their complex behavior. To address this problem we propose the Poisson Mixture Model-based NMF-HMM (PMM-NMF-HMM) speech enhancement algorithm, which is a more sophisticated statistical model capable of capturing more complex behavior, similarly to Gaussian mixture models [25]. This model makes it possible to better describe the speech and noise because these may be governed by multiple underlying causes, each being responsible for one particular mixture component in the distribution. If such causes are identified, then the PMM-NMF-HMM can be decomposed into a set of cause-dependent or context-dependent component distributions [25]. As a result, the performance can, arguably, be improved by exploiting this. Furthermore, like the NMF-HMM-based speech enhancement algorithm, the proposed method can be implemented using multiplicative updates (MU) of the parameters. For performing the enhancement given the trained speech and noise models, we propose an PMM-NMF-HMM-based MMSE estimator, which can be implemented using online parameter updates suitable for parallel computations. Moreover, compared to typical DNN-based method [14], the proposed method uses a small model size with few degrees of freedom.

2 Signal Model

In this section, we will briefly introduce the signal model that the proposed method is based. In an acoustic environment with additive noise, the observed signal model can be written as

$$y(l) = s(l) + d(l), \tag{C.1}$$

where $y(l)$, $s(l)$ and $d(l)$ represent the observed, speech and noise signals, respectively, and l is the time index. The short-time Fourier transform (STFT) of $y(l)$ can be written

as

$$Y(f,n) = S(f,n) + D(f,n), \tag{C.2}$$

where $Y(f,n)$, $S(f,n)$, and $D(f,n)$ denotes the frequency spectrums of $y(l)$, $s(l)$, and $d(l)$, respectively. The f is the frequency bin index and the n is the time frame index. Collecting F frequency bins and N time frames, the magnitude spectrum matrices can be defined as $\mathbf{Y}_N$, $\mathbf{S}_N$ and $\mathbf{D}_N$, where $\mathbf{Y}_N = [\mathbf{y}_1, \cdots, \mathbf{y}_n, \cdots, \mathbf{y}_N]$ and $\mathbf{y}_n = [|Y(1,n)|, \cdots, |Y(f,n)|, \cdots, |Y(F,n)|]^T$, $\mathbf{s}_n$ and $\mathbf{d}_n$ are defined similarly to $\mathbf{y}_n$. And $\mathbf{S}_N$ and $\mathbf{D}_N$ are defined similarly to $\mathbf{Y}_N$. Additionally, the proposed method is based on the approximation $\mathbf{Y}_N \approx \mathbf{S}_N + \mathbf{D}_N$. The overbar $(\bar{\cdot})$ and double dots $(\ddot{\cdot})$ are used to represent the speech and the noise, respectively. The signal models for the speech and the noise signal are the same, so we will in what follows only shown them for the speech signal. Applying the conditional independence property of the standard HMM, the likelihood function for the speech can be expressed as follows:

$$p(\mathbf{S}_N; \mathbf{\Phi}) = \sum_{\bar{\mathbf{x}}_\mathbf{N}} \prod_{n=1}^{N} p(\mathbf{s}_n|\bar{x}_n) p(\bar{x}_n|\bar{x}_{n-1}), \tag{C.3}$$

where $\bar{\mathbf{x}}_\mathbf{N} = [\bar{x}_1, \cdots, \bar{x}_n, \cdots, \bar{x}_N]^T$ is a collection of states, $\bar{x}_n \in \{1, 2, \cdots, \bar{J}\}$ represents the state at the n^{th} frame and $\bar{J}$ denotes the total number of states. $p(\bar{x}_n|\bar{x}_{n-1})$ is the state transition probability from state $\bar{x}_{n-1}$ to $\bar{x}_n$ with $p(\bar{x}_1|\bar{x}_0)$ being the initial state probability. $p(\mathbf{s}_n|\bar{x}_n)$ is the state-conditioned likelihood function, $\mathbf{\Phi}$ is a collection of modeling parameters. In this work, we propose to apply PMM-NMF-HMM to estimate the $p(\mathbf{s}_n|\bar{x}_n)$, which can be written as

$$p(\mathbf{s}_n|\bar{x}_n) = \int p(\mathbf{s}_n|\bar{z}_n) p(\bar{z}_n|\bar{x}_n)\, d\bar{z}_n, \tag{C.4}$$

$$p(\bar{z}_n|\bar{x}_n) = \prod_{j=1}^{\bar{J}} \prod_{t=1}^{\bar{T}} \bar{P}_{j,t}^{\,l(\bar{x}_n=j, \bar{z}_n=t)}, \tag{C.5}$$

where $\bar{z}_n \in \{1, 2, \cdots, \bar{T}\}$ denotes the mixture state and $\bar{T}$ is the total number of mixture states. Additionally, we define $\bar{\mathbf{z}}_\mathbf{N} = [\bar{z}_1, \cdots, \bar{z}_n, \cdots, \bar{z}_N]^T$, which is a collection of mixture states. The $\bar{P}_{j,t}$ is the mixture weight and there is $\sum_{t=1}^{\bar{T}} \bar{P}_{j,t} = 1 (1 \le j \le \bar{J})$. The $l(\cdot)$ denotes an indicator function, which is 1 when the logical expression in the parentheses is true and 0 otherwise. In [26] it was demonstrated that the Kullback-Leibler (KL) divergence-based NMF can be derived from the following hierarchical statistical model:

$$\mathbf{S}_N = \sum_{k=1}^{\bar{K}} \bar{\mathbf{C}}(k), \tag{C.6}$$

$$\bar{c}_{f,n}(k) \sim \mathcal{PO}(\bar{c}_{f,n}(k); \bar{W}_{f,k} \bar{H}_{k,n}), \tag{C.7}$$

where $\mathcal{PO}(x;\lambda) = \lambda^x e^{-\lambda}/\Gamma(x+1)$ is the Poisson distribution, $\Gamma(x+1) = x!$ denotes the gamma function for positive integer x, $\bar{K}$ denotes the number of basis vectors, $\bar{\mathbf{C}}(k)$ is the latent matrix and $\bar{c}_{f,n}(k)$ denotes the element of $\bar{\mathbf{C}}(k)$ in the f^{th} row and n^{th} column. $\bar{W}_{f,k}$ and $\bar{H}_{k,n}$ correspond to the elements of the basis and activation matrices for the NMF. Based on the (C.6) and (C.7), we propose to apply the following hierarchical model to estimate $p(\mathbf{s}_n|\bar{z}_n)$,

$$\mathbf{s}_n = \sum_{k=1}^{\overline{K}} \overline{\mathbf{c}}_n(k), \tag{C.8}$$

$$p(\overline{\mathbf{c}}_n(k)|\overline{z}_n, \overline{x}_n) = \prod_{j,t,k,f} \{\mathcal{PO}(\overline{c}_{t,f,n}(k); \overline{W}_{f,k}^{\overline{x}_n,\overline{z}_n} \overline{H}_{k,n}^{\overline{x}_n,\overline{z}_n})\}^{l(\overline{x}_n=j,\overline{z_n}=t)}, \tag{C.9}$$

$$p(\mathbf{s}_n|\overline{\mathbf{c}}_n) = \delta(\mathbf{s}_n - \sum_k \overline{\mathbf{c}}_n(k)), \tag{C.10}$$

where $\overline{W}_{f,k}^{\overline{x}_n,\overline{z}_n}$ and $\overline{H}_{k,n}^{\overline{x}_n,\overline{z}_n}$ correspond to the elements of the basis and activation matrices and $\overline{\mathbf{c}}_n(k)$ contains the hidden variables, writing $\overline{\mathbf{c}}_n = [\overline{\mathbf{c}}_n(1)^T, \overline{\mathbf{c}}_n(2)^T, \cdots, \overline{\mathbf{c}}_n(\overline{K})^T]^T$ and integrating $\overline{\mathbf{c}}_n$ out, we obtain

$$\begin{aligned} p(\mathbf{s}_n|\overline{z}_n) &= \int p(\mathbf{s}_n|\overline{\mathbf{c}}_n) p(\overline{\mathbf{c}}_n|\overline{z}_n)\, d\overline{\mathbf{c}}_n \\ &= \prod_{j,t,f} \{\mathcal{PO}(|S(f,n)|; \sum_{k=1}^{\overline{K}} \overline{W}_{f,k}^{\overline{x}_n,\overline{z}_n} \overline{H}_{k,n}^{\overline{x}_n,\overline{z}_n})\}^{l(\overline{x}_n=j,\overline{z}_n=t)}. \end{aligned} \tag{C.11}$$

Finally, combining (C.4) and (C.5), at jth state, the (C.11) can be written as

$$\begin{aligned} &p(\mathbf{s}_n|\overline{x}_n = j) = \\ &\sum_{t=1}^{\overline{T}} \overline{P}_{j,t} \prod_{f=1}^{F} \mathcal{PO}(|S(f,n)|; \sum_{k=1}^{\overline{K}} \overline{W}_{f,k}^{\overline{x}_n,\overline{z}_n} \overline{H}_{k,n}^{\overline{x}_n,\overline{z}_n}) \end{aligned} \tag{C.12}$$

Moreover, we have that

$$\begin{aligned} &p(\mathbf{s}_n|\overline{x}_n, \overline{z}_n) = \\ &\mathcal{PO}(|S(f,n)|; \sum_{k=1}^{\overline{K}} \overline{W}_{f,k}^{\overline{x}_n,\overline{z}_n} \overline{H}_{k,n}^{\overline{x}_n,\overline{z}_n}). \end{aligned} \tag{C.13}$$

We collect the unknown parameters $\{\overline{W}_{f,k}^{\overline{x}_n,\overline{z}_n}\}$ and $\{\overline{H}_{k,n}^{\overline{x}_n,\overline{z}_n}\}$ in matrices $\{\overline{\mathbf{W}}^{\mathbf{j},\mathbf{t}}\}$ and $\{\overline{\mathbf{H}}^{\mathbf{j},\mathbf{t}}\}$. To summarize, there are five parameters to be estimated in our proposed clean speech model. They are the initial state probability matrix $\overline{\boldsymbol{\pi}}$, state transition probability matrix $\overline{\mathbf{A}}$, basis matrix $\overline{\mathbf{W}}^{\mathbf{j},\mathbf{t}}$, activation matrix $\overline{\mathbf{H}}^{\mathbf{j},\mathbf{t}}$ and mixture weight matrix $\overline{\mathbf{P}}$. The activation matrix $\overline{\mathbf{H}}^{\mathbf{j},\mathbf{t}}$ is estimated in the online speech enhancement stage while the other parameters are obtained in the offline training stage. Additionally, the $\overline{K}$ and $\overline{T}$ can be predetermined. For the observed signal, the initial state and transition probabilities matrix can be expressed as $\overline{\boldsymbol{\pi}} \otimes \ddot{\boldsymbol{\pi}}$ and $\overline{\mathbf{A}} \otimes \ddot{\mathbf{A}}$, where the $\otimes$ denotes the Kronecker product. Thus, the conditional likelihood function can be written as

$$\begin{aligned} p(\mathbf{y}_n|\overline{x}_n, \ddot{x}_n) = &\sum_{t=1}^{\overline{T}} \sum_{t=1}^{\ddot{T}} \overline{P}_{j,t} \ddot{P}_{j,t} \prod_{f=1}^{F} \mathcal{PO}(|Y(f,n)|; \\ &\sum_{k=1}^{\overline{K}} \overline{W}_{f,k}^{\overline{x}_n,\overline{z}_n} \overline{H}_{k,n}^{\overline{x}_n,\overline{z}_n} + \sum_{k=1}^{\ddot{K}} \ddot{W}_{f,k}^{\ddot{x}_n,\ddot{z}_n} \ddot{H}_{k,n}^{\ddot{x}_n,\ddot{z}_n}) \end{aligned} \tag{C.14}$$

$$p(\mathbf{y}_n|\overline{z}_n,\ddot{z}_n,\overline{x}_n,\ddot{x}_n)=\prod_{f=1}^{F}\mathcal{PO}(|Y(f,n)|;\\ \sum_{k=1}^{\overline{K}}\overline{W}_{f,k}^{\overline{x}_n,\overline{z}_n}\overline{H}_{k,n}^{\overline{x}_n,\overline{z}_n}+\sum_{k=1}^{\ddot{K}}\ddot{W}_{f,k}^{\ddot{x}_n,\ddot{z}_n}\ddot{H}_{k,n}^{\ddot{x}_n,\ddot{z}_n}). \tag{C.15}$$

3 Offline Parameter Estimation

As mentioned above, the algorithm can be divided into two stages. In the offline training stage, the parameters of speech and noise signal model are estimated by using the speech and noise database, respectively. First, we define the complete data set $(\mathbf{S}_N,\overline{\mathbf{x}}_N,\overline{\mathbf{z}}_N,\overline{\mathbf{C}}_N)$, where $\overline{\mathbf{C}}_N=[\overline{\mathbf{c}}_1,\overline{\mathbf{c}}_2,\cdots,\overline{\mathbf{c}}_N]$. Based on the (C.3) and derivation in Section 2, using the conditional independence property, the complete data likelihood function can be written as

$$\begin{aligned}&p(\mathbf{S}_N,\overline{\mathbf{x}}_N,\overline{\mathbf{z}}_N,\overline{\mathbf{C}}_N)\\&=\left(\prod_{n=1}^{N}p(\mathbf{s}_n|\overline{\mathbf{c}}_n)\right)\left(p(x_1)\prod_{n=2}^{N}p(\overline{\mathbf{x}}_n|\overline{\mathbf{x}}_{n-1})\right)\\&\left(\prod_{n=1}^{N}p(\overline{\mathbf{z}}_n|\overline{\mathbf{x}}_n)\right)\left(\prod_{n=1}^{N}p(\overline{\mathbf{c}}_n|\overline{\mathbf{x}}_n,\overline{\mathbf{z}}_n)\right).\end{aligned} \tag{C.16}$$

Using Expectation–Maximization (EM) algorithm [27], the model parameters can be estimated. For simplicity, we here omit the derivation process. It can be shown that the parameter updates can be written as follows:

$$\overline{\pi}_j=\frac{q(\overline{x}_1=j)}{\sum_{o=1}^{\overline{J}}q(\overline{x}_1=o)}, \tag{C.17}$$

$$\overline{A}_{o,j}=\frac{\sum_{n=2}^{\overline{N}}q(\overline{x}_n=j,\overline{x}_{n-1}=o)}{\sum_{j=1}^{\overline{J}}\sum_{n=2}^{\overline{N}}q(\overline{x}_n=j,\overline{x}_{n-1}=o)}, \tag{C.18}$$

where $1\le o,j\le\overline{J}$. The quantities $q(\overline{x}_n)$ and $q(\overline{x}_n,\overline{x}_{n-1})$ correspond to the posterior state probability and the joint posterior probability, which can be calculated by forward-backward algorithm [24] that combines the (C.12). The $\overline{\pi}_j$ and $\overline{A}_{o,j}$ is the elements of $\overline{\mathbf{A}}$ and $\overline{\boldsymbol{\pi}}$, respectively. The estimation of $\overline{\mathbf{A}}$ and $\overline{\boldsymbol{\pi}}$ is similar to the traditional HMM. In addition, we have the following updates:

$$\overline{\mathbf{W}}^{\mathbf{j,t}}\leftarrow\overline{\mathbf{W}}^{\mathbf{j,t}}\odot\frac{\frac{\mathbf{S_N}}{\overline{\mathbf{W}}^{\mathbf{j,t}}\overline{\mathbf{H}}^{\mathbf{j,t}}}\boldsymbol{\Lambda}(\mathbf{j},\mathbf{t})(\overline{\mathbf{H}}^{\mathbf{j,t}})^T}{\mathbf{1}\boldsymbol{\Lambda}(\mathbf{j},\mathbf{t})(\overline{\mathbf{H}}^{\mathbf{j,t}})^T}, \tag{C.19}$$

$$\overline{\mathbf{H}}^{\mathbf{j,t}}\leftarrow\overline{\mathbf{H}}^{\mathbf{j,t}}\odot\frac{(\overline{\mathbf{W}}^{\mathbf{j,t}})^T\frac{\mathbf{S_N}}{\overline{\mathbf{W}}^{\mathbf{j,t}}\overline{\mathbf{H}}^{\mathbf{j,t}}}}{(\overline{\mathbf{W}}^{\mathbf{j,t}})^T\mathbf{1}}, \tag{C.20}$$

where $\mathbf{\Lambda}(\mathbf{j},\mathbf{t}) = \mathrm{diag}(q(\overline{x}_1 = j, \overline{z}_1 = t), q(\overline{x}_2 = j, \overline{z}_2 = t), \cdots, q(\overline{x}_N = j, \overline{z}_N = t))$. The $q(\overline{x}_n = j, \overline{z}_n = t)$ is the posterior probability when $(\overline{x}_n = j, \overline{z}_n = t)$. Once again, this calculation can be performed using the forward-backward algorithm which uses (C.13). Furthermore, this update is in the form of an multiplicative update, which means that the offline training can be performed efficiently. Moreover, we have

$$\overline{P}_{j,t} = \frac{\sum_{n=1}^{N} q(\overline{x}_n = j, \overline{z}_n = t)}{\sum_{n=1}^{N}\sum_{t=1}^{\overline{T}} q(\overline{x}_n = j, \overline{z}_n = t)} \tag{C.21}$$

This mixture weight $\overline{P}_{j,t}$ determines the importance of each latent cause that is modeled by single Poisson distribution for the whole speech signal.

4 Online Speech Enhancement

In the online enhancement stage, we propose a novel MMSE estimator, which is based on the model produced by the PMM-NMF-HMM algorithm. The MMSE estimate of the speech signal from the noisy observation is

$$\hat{\mathbf{s}}_n = \mathbb{E}_{\mathbf{s}_n|\mathbf{Y}_n}(\mathbf{s}_n) = \int \mathbf{s}_n p(\mathbf{s}_n|\mathbf{Y}_n)\, d\mathbf{s}_n. \tag{C.22}$$

For simplicity, we omit the specific details of this derivation. The enhanced speech can be written as $\hat{\mathbf{s}}_n = \mathbf{y}_n \odot \mathbf{g}_n$ where $\mathbf{g}_n$ can be viewed as a spectral gain vector with

$$\mathbf{g}_n = \sum_{\overline{x}_n, \ddot{x}_n} \omega_{\overline{x}_n, \ddot{x}_n} \left(\sum_{\overline{z}_n, \ddot{z}_n} \overline{P}_{j,t} \ddot{P}_{j,t} \mathbf{p}_n(\overline{x}_n, \ddot{x}_n, \overline{z}_n, \ddot{z}_n) \right), \tag{C.23}$$

where the weight $0 \leq \omega_{\overline{x}_n, \ddot{x}_n} \leq 1$ can be written as

$$\omega_{\overline{x}_n, \ddot{x}_n} = \frac{p(\mathbf{y}_n|\overline{x}_n, \ddot{x}_n) p(\overline{x}_n, \ddot{x}_n|\mathbf{Y}_{n-1})}{\sum_{\overline{x}_n, \ddot{x}_n} p(\mathbf{y}_n|\overline{x}_n, \ddot{x}_n) p(\overline{x}_n, \ddot{x}_n|\mathbf{Y}_{n-1})}. \tag{C.24}$$

The calculation of $p(\mathbf{y}_n|\overline{x}_n, \ddot{x}_n)$ can be conducted using (C.14), and

$$\begin{aligned} & p(\overline{x}_n, \ddot{x}_n|\mathbf{Y}_{n-1}) \\ &= \sum_{\overline{x}_{n-1}, \ddot{x}_{n-1}} p(\overline{x}_n, \ddot{x}_n|\overline{x}_{n-1}, \ddot{x}_{n-1}, \mathbf{Y}_{n-1}) p(\overline{x}_{n-1}, \ddot{x}_{n-1}|\mathbf{Y}_{n-1}) \\ &= \sum_{\overline{x}_{n-1}, \ddot{x}_{n-1}} p(\overline{x}_n, \ddot{x}_n|\overline{x}_{n-1}, \ddot{x}_{n-1}) p(\overline{x}_{n-1}, \ddot{x}_{n-1}|\mathbf{Y}_{n-1}), \end{aligned} \tag{C.25}$$

In (C.25), the first term can be calculated by the transition probabilities matrix of observed signal and the second term is the forward probability which can be calculated by a forward algorithm [24]. In (C.23), $\mathbf{p}_n(\overline{x}_n, \ddot{x}_n, \overline{z}_n, \ddot{z}_n) = [p_{1,n}(\overline{x}_n, \ddot{x}_n, \overline{z}_n, \ddot{z}_n), p_{2,n}(\overline{x}_n, \ddot{x}_n, \overline{z}_n, \ddot{z}_n), \cdots, p_{F,n}(\overline{x}_n, \ddot{x}_n, \overline{z}_n, \ddot{z}_n)]^T$, where

$$\begin{aligned} & p_{f,n}(\overline{x}_n, \ddot{x}_n, \overline{z}_n, \ddot{z}_n) = \\ & \frac{\sum_{k=1}^{\overline{K}} \overline{W}_{f,k}^{\overline{x}_n, \overline{z}_n} \overline{H}_{k,n}^{\overline{x}_n, \overline{z}_n}}{\sum_{k=1}^{\overline{K}} \overline{W}_{f,k}^{\overline{x}_n, \overline{z}_n} \overline{H}_{k,n}^{\overline{x}_n, \overline{z}_n} + \sum_{k=1}^{\ddot{K}} \ddot{W}_{f,k}^{\ddot{x}_n, \ddot{z}_n} \ddot{H}_{k,n}^{\ddot{x}_n, \ddot{z}_n}}, \end{aligned} \tag{C.26}$$

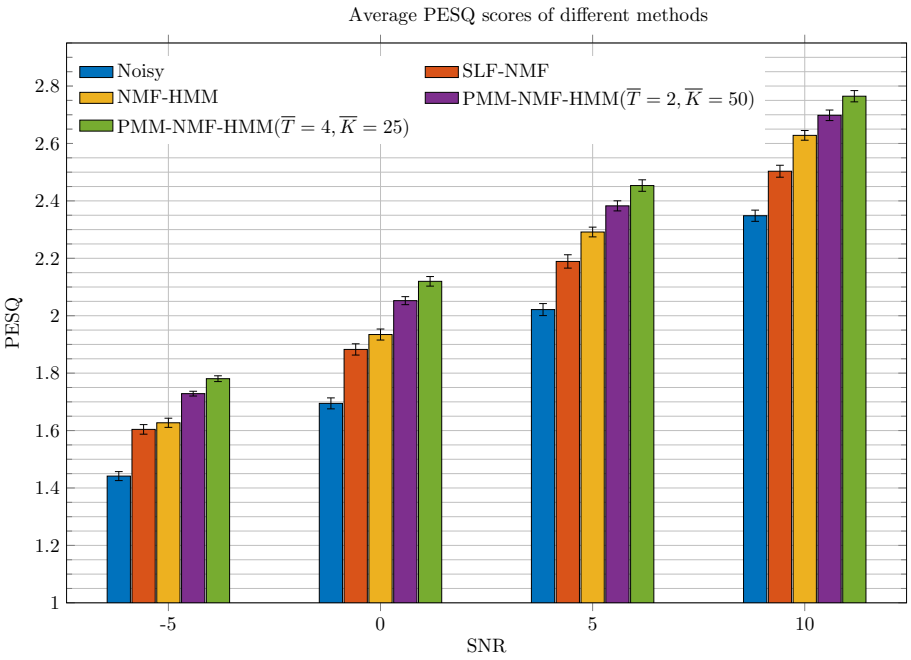


Fig. C.1: Average PESQ scores of different algorithms using six types of noise under four different SNRs.

Comparing the PMM-NMF-HMM-based MMSE estimator with our previous proposed NMF-HMM-based MMSE estimator [24], we can find that there are more than one NMF basis matrices at each HMM hidden state, which means that our algorithm can model more underlying possible causes in the observed signal, so the enhancement performance can likely be improved based on this better model. Furthermore, we also remark that parallel computing can be applied to conduct the online estimation of active matrix $\overline{\mathbf{H}}^{\mathbf{j},\mathbf{t}}$ to effectively reduce the time consumption.

5 EXPERIMENTAL Result and Analysis

In this section, the performance of proposed algorithm was evaluated and compared with state-of-the-art NMF-based speech enhancement algorithms. All the experiments were conducted on the TIMIT [28] and NOISEX-92 [29] databases. In the training stage, all 4620 utterances from the training set of the TIMIT database were used to train the speech PMM-NMF-HMM model. Meanwhile, parts of the Babble, F16, Factory and White noise from the NOISEX-92 database were used to train the noise model. In the test stage, 200 utterances were randomly chosen from the test set of the TIMIT database. After that, the chosen 200 utterances were added to six types of noise at four different SNR levels (i.e., -5, 0, 5, and 10 dB). There were two types of noise (destroyerengine and destroyerops) that were not included in the training database to

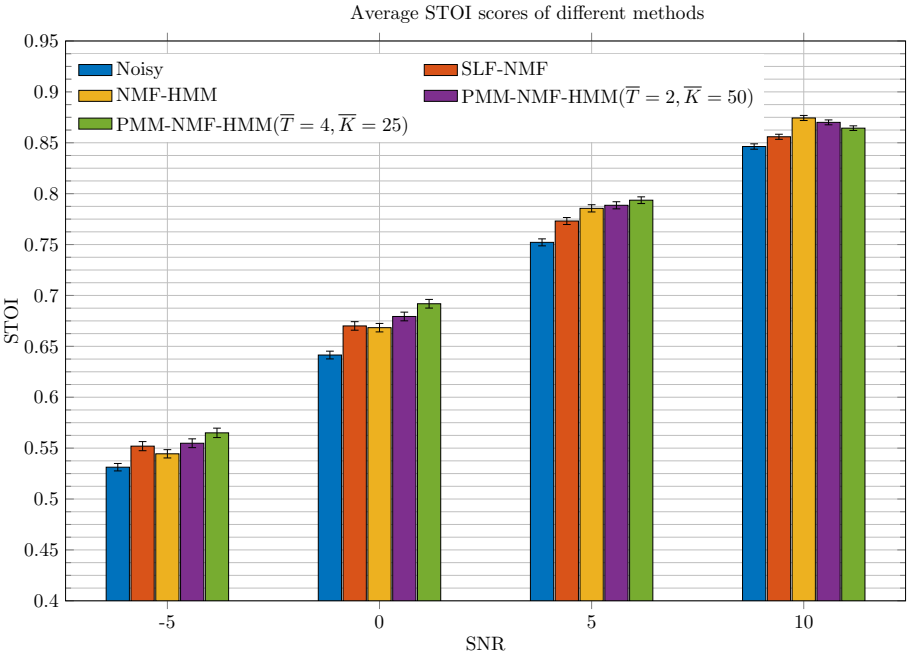


Fig. C.2: Average STOI scores of different algorithms using six types of noise under four different SNRs.

test the generalization ability of the noise model. It must be stressed that for all noise types, disjoint training and test data was used.

To evaluate the performance of the proposed method, we compare to two state-of-the-art methods, namely the NMF-HMM [24] and the variable span linear filters [6] (SLF-NMF) combined with parametric NMF [10] for estimating the noise and speech statsitics.

In the experiments, all the signal waveform was down-sampled to 16 kHz. The frame length was set to 1024 samples with a frame shift of 512 samples . The size of STFT was 1024 points with a Hanning window. Furthermore, the maximum number of iterations was set to 30 in the training stage and 15 in the online speech enhancement stage for these NMF-based methods. In addition, the PESQ [30], ranging from -0.5 to 4.5, was used to evaluate the enhanced speech quality. The STOI [31], ranging from 0 to 1, was used to measure the enhanced speech intelligibility.

For the NMF parameter setting, to better compare the performance of PMM-NMF-HMM and NMF-HMM, we ensure that there are the same total number of basis vector for the two models. For the NMF-HMM, there is no the mixture weight (the NMF-HMM can be seen as a special case of PMM-NMF-HMM when $\overline{T}=1$ and $\ddot{T}=1$), so we only need to set $\overline{J}=10,\overline{K}=100,\ddot{J}=2$ and $\ddot{K}=70$. For the PMM-NMF-HMM, we have $\overline{J}=10,\ddot{J}=2,\ddot{K}=70,\ddot{T}=1$. We investigate the two different $\overline{T}$. When $\overline{T}$ is set to 2 and 4, the $\overline{K}$ corresponds to 50 and 25, which ensures that there is the same number of total basis vector. For the SLF-NMF, we utilize the maximum SNR filter and the codebook size of speech and noise is set to 64 and 8, respectively. Figure C.1 indicates the average

PESQ scores with 95% confidence interval of these algorithms. The NMF-HMM-based methods always achieve higher PESQ scores than SLF-NMF for all four SNRs. Additionally, with increased total number of mixture state $\overline{T}$, PMM-NMF-HMM achieve the better performance. This indicates that PMM-NMF-HMM may effectively better model multiple underlying causes in speech and noise when improving the speech quality. Figure C.2 shows the average STOI scores with 95% confidence interval of the methods. We can see that the PMM-NMF-HMM achieves better speech enhancement performance at low SNRs (-5, 0, 5dB) with increased numbers of mixture state $\overline{T}$. However, for high SNRs, more mixture states does not lead to a better performance.

6 Conclusions

In this work, we have proposed a novel PMM-NMF-HMM-based speech enhancement algorithm. The new method employes a PMM which was used to model the state-conditioned likelihood function for the HMM, whereby multiple underlying causes in the signals could be captured. More specifically, the resulting modal can be decomposed into the different sets of cause-dependent or context-dependent component distributions. Finally, as a result of the new and more sophisticated model, the speech can be estimated more accurately. To enhance the speech, we have proposed a novel MMSE estimator, which is also based on the model of the PMM-NMF-HMM method. This estimator can be implemented efficiently and is thus suitable for online speech enhancement. In general, the experimental results showed that the proposed PMM-NMF-HMM method outperforms the previously proposed NMF-HMM, though the STOI score was slightly lower than NMF-HMM at high SNR (10dB).

References

[1] Philipos C Loizou, *Speech enhancement: theory and practice*, CRC press, 2013.

[2] Jinyu Li, Li Deng, Yifan Gong, and Reinhold Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 22, no. 4, pp. 745–777, 2014.

[3] Israel Cohen and Sharon Gannot, "Spectral enhancement methods," in *Springer Handbook of Speech Processing*, pp. 873–902. Springer, 2008.

[4] Steven Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 27, no. 2, pp. 113–120, 1979.

[5] Yariv Ephraim and David Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 32, no. 6, pp. 1109–1121, 1984.

[6] Jesper Rindom Jensen, Jacob Benesty, and Mads Græsbøll Christensen, "Noise reduction with optimal variable span linear filters," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 24, no. 4, pp. 631–644, 2015.

[7] Israel Cohen and Baruch Berdugo, "Speech enhancement for non-stationary noise environments," *Signal Process.*, vol. 81, no. 11, pp. 2403–2418, 2001.

[8] Israel Cohen, "Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 5, pp. 466–475, 2003.

[9] Daniel D Lee and H Sebastian Seung, "Algorithms for non-negative matrix factorization," in *Proc. Advances in Neural Inform. Process. Syst.*, 2001, pp. 556–562.

[10] Mathew Shaji Kavalekalam, Jesper Kjær Nielsen, Liming Shi, Mads Græsbøll Christensen, and Jesper Boldt, "Online parametric NMF for speech enhancement," in *Proc. European Signal Processing Conf.*, 2018, pp. 2320–2324.

[11] David Y Zhao and W Bastiaan Kleijn, "HMM-based gain modeling for enhancement of speech in noise," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 15, no. 3, pp. 882–892, 2007.

[12] Emad M Grais and Hakan Erdogan, "Single channel speech music separation using nonnegative matrix factorization and spectral masks," in *Int. Conf. Digital Signal Process.*, 2011, pp. 1–6.

[13] Yoshua Bengio et al., "Learning deep architectures for AI," *Foundations and trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[14] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Process. Lett.*, vol. 21, no. 1, pp. 65–68, 2013.

[15] Se Rim Park and Jinwon Lee, "A fully convolutional neural network for speech enhancement," *arXiv preprint arXiv:1609.07132*, 2016.

[16] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis, "Joint optimization of masks and deep recurrent neural networks for monaural source separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 23, no. 12, pp. 2136–2147, 2015.

[17] Santiago Pascual, Antonio Bonafonte, and Joan Serrà, "Segan: Speech enhancement generative adversarial network," *Proc. Interspeech*, pp. 3642–3646, 2017.

[18] Lei Sun, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "Multiple-target deep learning for lstm-rnn based speech enhancement," in *2017 Hands-free Speech Communications and Microphone Arrays (HSCMA)*, 2017, pp. 136–140.

[19] Yuxuan Wang, Arun Narayanan, and DeLiang Wang, "On training targets for supervised speech separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 22, no. 12, pp. 1849–1858, 2014.

[20] Yi Luo and Nima Mesgarani, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 27, no. 8, pp. 1256–1266, 2019.

[21] Thanh T Vu, Benjamin Bigot, and Eng Siong Chng, "Combining non-negative matrix factorization and deep neural networks for speech enhancement and automatic speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2016, pp. 499–503.

[22] Simon Leglaive, Laurent Girin, and Radu Horaud, "Semi-supervised multichannel speech enhancement with variational autoencoders and non-negative matrix factorization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 101–105.

[23] Nasser Mohammadiha, Paris Smaragdis, and Arne Leijon, "Supervised and unsupervised speech enhancement using nonnegative matrix factorization," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 21, no. 10, pp. 2140–2151, 2013.

[24] Yang Xiang, Liming Shi, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, and Mads Græsbøll Christensen, "An NMF-HMM speech enhancement method based on kullback-leibler divergence," in *Proc. Interspeech*, 2020, pp. 2667–2671.

[25] Dong Yu and Li Deng, *Automatic Speech Recognition*, Springer, 2016.

[26] Ali Taylan Cemgil, "Bayesian inference for nonnegative matrix factorisation models," *Computational intelligence and neuroscience*, vol. 2009, 2009.

[27] Leonard E Baum, "An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes," *Inequalities*, vol. 3, no. 1, pp. 1–8, 1972.

[28] John S Garofolo, Lori F Lamel, William M Fisher, Jonathan G Fiscus, and David S Pallett, "DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1," *NASA STI/Recon technical report n*, vol. 93, 1993.

[29] Andrew Varga and Herman JM Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 12, no. 3, pp. 247–251, 1993.

[30] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2001, vol. 2, pp. 749–752.

[31] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, 2011.

Paper D

A Bayesian Permutation training deep representation learning method for speech enhancement with variational autoencoder

Yang Xiang, Jesper Lisby Højvang, Morten Højfeldt Rasmussen,
Mads Græsbøll Christensen

The paper has been published in the
Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., 2022

The layout has been revised.

Abstract

Recently, variational autoencoder (VAE), a deep representation learning (DRL) model, has been used to perform speech enhancement (SE). However, to the best of our knowledge, current VAE-based SE methods only apply VAE to model speech signal, while noise is modeled using the traditional non-negative matrix factorization (NMF) model. One of the most important reasons for using NMF is that these VAE-based methods cannot disentangle the speech and noise latent variables from the observed signal. Based on Bayesian theory, this paper derives a novel variational lower bound for VAE, which ensures that VAE can be trained in supervision, and can disentangle speech and noise latent variables from the observed signal. This means that the proposed method can apply the VAE to model both speech and noise signals, which is totally different from the previous VAE-based SE works. More specifically, the proposed DRL method can learn to impose speech and noise signal priors to different sets of latent variables for SE. The experimental results show that the proposed method can not only disentangle speech and noise latent variables from the observed signal, but also obtain a higher scale-invariant signal-to-distortion ratio and speech quality score than the similar deep neural network-based (DNN) SE method.

1 Introduction

In real-world environments, speech signals are often distorted due to the presence of background noise. To reduce the effects of noise, speech enhancement (SE) techniques have been developed [1, 2] to improve the quality and intelligibility of an observed signal.

Currently, many single-channel SE algorithms have been proposed, which include some unsupervised algorithms [3, 4] and supervised algorithms [5, 6]. However, these methods usually apply linear processes to model complex high-dimensional signal, which is not always reasonable in practical applications [7]. Thus, non-linear deep neural network (DNN) models have been developed. As shown in [2, 7–10], DNN-based methods can achieve better SE performance than traditional linear models. However, their generalization ability is not often satisfactory for the unseen noise conditions [2].

Recently, deep probabilistic generative models have been investigated to improve the DNN's generalization ability for SE, such as generative adversarial networks (GAN) [11] and the variational autoencoder (VAE) [12, 13]. VAE can learn the probability distribution of complex data and perform efficient approximate posterior inference, so VAE-based SE algorithms have been proposed [13–15]. However, the VAE of these methods is trained in an unsupervised manner on speech only, and the noise is modeled by an NMF model because these methods cannot disentangle the speech and noise latent variables from the observed signal. This means that these algorithms are not robust [16], and their SE performance is limited compared to DNN-based supervised methods [13]. To mitigate this problem, supervised VAE-based SE methods have been proposed. In [16, 17], a supervised classifier [16] and a supervised noise-aware training strategy [17] are introduced to the training of speech VAE. The purpose is to obtain a more robust speech latent variable from the observed signal. However, the noise is still modeled by a linear NMF model because it is a difficult task to disentangle the speech and noise latent variables from the observed signal [17].

Learning interpretable latent representation is a challenging but very useful task

because it can explain how different factors influence the speech signal, which is important in speech-related applications [18]. In [18], a latent space arithmetic operation was derived to modify the speech attributes (phonetic content and speaker identity). [19] proposed an unsupervised method to distinguish different latent variables and generate new latent variables for the ASR application. [20] applied VAE to learn the sequence-dependent and sequence-independent representations. However, interpretable latent representation is rarely considered in current SE algorithms [7–10].

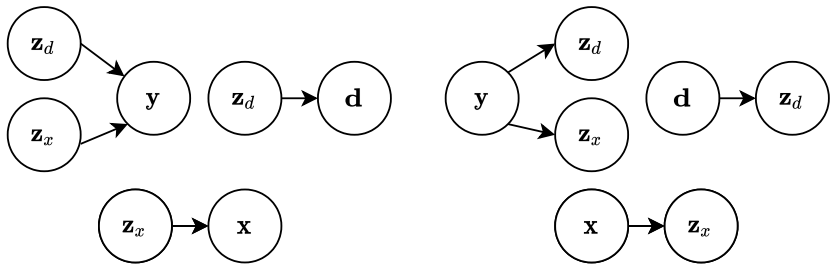
Inspired by previous work, in this paper, we propose a Bayesian permutation training method for SE. The proposed method can disentangle the latent speech and noise variables from the observed signal in a supervised manner and conduct the mapping between latent variables and targets, even though this is a difficult task [17]. We hypothesize that disentangling latent variables can improve the performance of the supervised DNN-based SE method. To achieve this, a clean speech VAE (C-VAE) and a noise VAE (N-VAE) are separately pre-trained without supervision. After that, based on Bayesian theory and our derived variational lower bound, we use the two pre-trained VAEs to train a noisy VAE (NS-VAE) in a supervised manner. The trained NS-VAE can learn the latent representations of the speech and noise signal. When we conduct SE, the trained NS-VAE is first used to predict the latent variables of the speech and noise signal. Then, the two latent variables are independently used as the decoder input of the C-VAE and N-VAE to estimate the corresponding speech and noise. Finally, the enhanced signal can be acquired by direct speech waveform reconstruction or with post-filtering methods. Compared to previous VAE-based SE methods [13–17] and interpretable latent representation learning methods [18–20], the proposed method derives a novel variational lower bound to ensure that supervision training can be used for VAE, and VAE can disentangle different latent variables to model noise for SE. The derived supervised lower bound is very different from previous VAE-based methods [12–20] that are trained on an unsupervised variational lower bound, which increases the robustness of different learned latent variables [17]. Moreover, our learned latent variables are attributed to different types of signal, so each single latent variable that is generated by NS-VAE can be used to generate the corresponding speech or noise signal, and their combination can generate noisy speech.

2 Problem Description

In this work, we aim to perform SE in an additive noisy environment. Thus, the signal model can be written as

$$y(t) = x(t) + d(t), \tag{D.1}$$

where $y(t)$, $x(t)$, and $d(t)$ represent the observed, speech, and noise signal, respectively, and t is the time index. Log-power spectrum (LPS) is suitable for direct signal estimation [7], so we use it as a feature for SE. The LPS of $y(t)$, $x(t)$, and $d(t)$ is written as $Y(f,n)$, $X(f,n)$, and $D(f,n)$, respectively. Here, $f \in [1,F]$ and $n \in [1,N]$ denote the frequency bins and time frame indices, respectively. Collecting F frequency bins and N time frames, we can obtain the LPS dataset $\mathbf{Y}_N$, $\mathbf{X}_N$ and $\mathbf{D}_N$ with N samples, where $\mathbf{Y}_N = [\mathbf{y}_1, \cdots, \mathbf{y}_n, \cdots, \mathbf{y}_N]$ and $\mathbf{y}_n = [Y(1,n), \cdots, Y(f,n), \cdots, Y(F,n)]^T$, $\mathbf{x}_n$ and $\mathbf{d}_n$ are defined similarly to $\mathbf{y}_n$. $\mathbf{X}_N$ and $\mathbf{D}_N$ are defined similarly to $\mathbf{Y}_N$. For simplicity, we use $\mathbf{y}$, $\mathbf{x}$, and $\mathbf{d}$ to represent a sample in dataset $\mathbf{Y}_N$, $\mathbf{X}_N$ and $\mathbf{D}_N$, respectively.In the



(a) Generative model (b) Recognition model

Fig. D.1: Graphical illustration of the proposed model.

proposed VAE model, we assume that $\mathbf{y}$ is generated from a random process involving the speech latent variables $\mathbf{z}_x$ and the noise latent variables $\mathbf{z}_d$, where the observed speech conditional prior distribution can be written as $q(\mathbf{y}|\mathbf{z}_d, \mathbf{z}_x)$. The dimensions of vectors $\mathbf{z}_x$ and $\mathbf{z}_d$ are L_x and L_d, respectively. The dataset of $\mathbf{z}_x$ and $\mathbf{z}_d$ is written as $\mathbf{Z}_{xN}$ and $\mathbf{Z}_{dN}$ with N samples. Here, we assume that the latent variables $\mathbf{z}_x$ and $\mathbf{z}_d$ are independent. Additionally, $\mathbf{x}$ and $\mathbf{d}$ are drawn from the speech prior distribution $q(\mathbf{x}|\mathbf{z}_x)$ and the noise prior distribution $q(\mathbf{d}|\mathbf{z}_d)$, respectively. The whole generative process is illustrated in Fig. D.1(a). In VAE, since exact posterior inference is intractable, we propose that $\mathbf{z}_x$ and $\mathbf{z}_d$ can be estimated from speech and noise posterior distributions $p(\mathbf{z}_x|\mathbf{x})$ and $p(\mathbf{z}_d|\mathbf{d})$, respectively, or they can also be estimated from the noisy speech posterior distributions $p(\mathbf{z}_x|\mathbf{y})$ and $p(\mathbf{z}_d|\mathbf{y})$. Here, we assume $p(\mathbf{z}_x, \mathbf{z}_d|\mathbf{y}) = p(\mathbf{z}_x|\mathbf{y})p(\mathbf{z}_d|\mathbf{y})$, which ensures that noise can be modeled by non-linear VAE rather than NMF. Based on these assumptions and our derivation in section 3, speech and noise latent variables can be obtained from the observed signal. The whole recognition process is shown in Fig. D.1(b).

To sum up, we intend to first estimate the latent variable distributions of the speech $p(\mathbf{z}_x|\mathbf{y})$ and the noise $p(\mathbf{z}_d|\mathbf{y})$ from the observed signal to acquire latent variables $\mathbf{z}_x$ and $\mathbf{z}_d$, respectively. After that, we use the estimated latent variables as the input of the decoder of C-VAE and N-VAE to obtain the probability distribution of $q(\mathbf{x}|\mathbf{z}_x)$ and $q(\mathbf{d}|\mathbf{z}_d)$ for SE.

3 SE with Bayesian Permutation Training

3.1 Variational Autoencoder with Multiple Latent Variables

VAE [12] defines a probabilistic generative process between observed signal and its latent variables and provides a principled method to jointly learn latent variables, generative and recognition models, which is achieved by maximizing variational lower bound using stochastic gradient descent (SGD) algorithm. This optimizing process [12] is equal to minimize Kullback-Leibler (KL) divergence (D_{KL}) between real joint probability distribution $p(\mathbf{y}, \mathbf{z}_x, \mathbf{z}_d)$ and its estimation $q(\mathbf{y}, \mathbf{z}_x, \mathbf{z}_d)$. This process can be written as follows:

$$
\begin{aligned}
&D_{KL}\left(p(\mathbf{y},\mathbf{z}_x,\mathbf{z}_d)||q(\mathbf{y},\mathbf{z}_x,\mathbf{z}_d)\right) = \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\log p(\mathbf{y})\right] \\
&+ \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}[D_{KL}\left(p(\mathbf{z}_x,\mathbf{z}_d|\mathbf{y})\right)||q(\mathbf{y},\mathbf{z}_x,\mathbf{z}_d))].
\end{aligned} \tag{D.2}
$$

In (D.2), the term $\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\log p(\mathbf{y})\right]$ is a constant, so minimizing their KL divergence is equal to minimizing

$$
\begin{aligned}
&\mathcal{L}(\theta_y,\varphi_y,\theta_x,\varphi_x,\theta_d,\varphi_d;\mathbf{y}) \\
&\quad= \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}[D_{KL}\left(p(\mathbf{z}_x,\mathbf{z}_d|\mathbf{y})\right)||q(\mathbf{y},\mathbf{z}_x,\mathbf{z}_d))] \\
&\quad= \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[D_{KL}\left(p(\mathbf{z}_x,\mathbf{z}_d|\mathbf{y})\right)||q(\mathbf{z}_x,\mathbf{z}_d))\right] \\
&\qquad- \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\mathbb{E}_{\mathbf{z}_d,\mathbf{z}_x\sim p(\mathbf{z}_d,\mathbf{z}_x|\mathbf{y})}\left[\log q(\mathbf{y}|\mathbf{z}_x,\mathbf{z}_d)\right]\right],
\end{aligned} \tag{D.3}
$$

where $\theta_y,\varphi_y,\theta_x,\varphi_x,\theta_d,\varphi_d$ are the parameters that are used to conduct the related probability estimation. The details will be presented later. Here, $-\mathcal{L}$ can be seen as the VAE variational lower bound with multiple latent variables ($\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}[\log q(\mathbf{y})] \geq -\mathcal{L}$) [12]. Minimizing $\mathcal{L}$ is equal to maximize this variational lower bound. Based on our assumptions in section 2 ($\mathbf{z}_x$ and $\mathbf{z}_d$ are independent and $p(\mathbf{z}_x,\mathbf{z}_d|\mathbf{y}) = p(\mathbf{z}_x|\mathbf{y})p(\mathbf{z}_d|\mathbf{y})$), (D.3) can be further written as

$$
\begin{aligned}
&\mathcal{L}(\theta_y,\varphi_y,\theta_x,\varphi_x,\theta_d,\varphi_d;\mathbf{y}) \\
&\quad= \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[D_{KL}\left(p(\mathbf{z}_x|\mathbf{y})||q(\mathbf{z}_x)\right)\right] \\
&\qquad+ \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[D_{KL}\left(p(\mathbf{z}_d|\mathbf{y})||q(\mathbf{z}_d)\right)\right] \\
&\qquad- \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\mathbb{E}_{\mathbf{z}_d,\mathbf{z}_x\sim p(\mathbf{z}_d,\mathbf{z}_x|\mathbf{y})}\left[\log q(\mathbf{y}|\mathbf{z}_x,\mathbf{z}_d)\right]\right],
\end{aligned} \tag{D.4}
$$

3.2 DRL with Bayesian Permutation Training

To estimate the speech and noise latent variables from the observed signal using (D.4), we propose a Bayesian permutation training process between NS-VAE, C-VAE, and N-VAE. First, the C-VAE and N-VAE are separately pre-trained using the general VAE training method [12] without supervision. The purpose is to acquire the posterior estimates $p(\mathbf{z}_x|\mathbf{x})$ and $p(\mathbf{z}_d|\mathbf{d})$. Then, the NS-VAE is trained with the supervision of C-VAE and N-VAE.

In (D.4), the calculation of the first and second term is similar, so we will only use the first term to show the Bayesian permutation process. To achieve supervision learning, we add an attention mechanism ($p(\mathbf{z}_x|\mathbf{x})/p(\mathbf{z}_x|\mathbf{x})$) for the calculation of first term in (D.4). Thus, its calculation can be written as (D.5)

$$
\begin{aligned}
&\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[D_{KL}\left(p(\mathbf{z}_x|\mathbf{y})||q(\mathbf{z}_x)\right)\right] \\
&\quad= \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\int p(\mathbf{z}_x|\mathbf{y})\log\frac{p(\mathbf{z}_x|\mathbf{y})p(\mathbf{z}_x|\mathbf{x})}{q(\mathbf{z}_x)p(\mathbf{z}_x|\mathbf{x})}\,d\mathbf{z}_x\right] \\
&\quad= \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y}),\mathbf{x}\sim p(\mathbf{x})}\left[D_{KL}\left(p(\mathbf{z}_x|\mathbf{y})||p(\mathbf{z}_x|\mathbf{x})\right)\right] \\
&\qquad+ \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{y})}[\log\frac{p(\mathbf{z}_x|\mathbf{x})}{q(\mathbf{z}_x)}]\right],
\end{aligned} \tag{D.5}
$$

$$
\begin{aligned}
&\mathcal{L}(\theta_y, \varphi_y, \theta_x, \varphi_x, \theta_d, \varphi_d; \mathbf{y}) \\
&\quad = \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y}), \mathbf{x}\sim \mathbf{p}(\mathbf{x})}\{D_{KL}\left(p(\mathbf{z}_x|\mathbf{y})||p(\mathbf{z}_x|\mathbf{x})\right) \\
&\qquad + \mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{y})}[\log \frac{p(\mathbf{z}_x|\mathbf{x})}{q(\mathbf{z}_x)}]\} \\
&\qquad + \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y}), \mathbf{d}\sim p(\mathbf{d})}\{D_{KL}\left(p(\mathbf{z}_d|\mathbf{y})||p(\mathbf{z}_d|\mathbf{d})\right) \\
&\qquad + \mathbb{E}_{\mathbf{z}_d\sim p(\mathbf{z}_d|\mathbf{y})}[\log \frac{p(\mathbf{z}_d|\mathbf{d})}{q(\mathbf{z}_d)}]\} \\
&\qquad - \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\mathbb{E}_{\mathbf{z}_d, \mathbf{z}_x\sim p(\mathbf{z}_d, \mathbf{z}_x|\mathbf{y})}\left[\log q(\mathbf{y}|\mathbf{z}_x, \mathbf{z}_d)\right]\right].
\end{aligned}
\tag{D.6}
$$

In (D.5), we introduce posterior $p(\mathbf{z}_x|\mathbf{x})$ estimated from C-VAE to conduct supervised latent variable learning. The purpose is to obtain speech latent variables from observed signal. Finally, substituting (D.5) into (D.4), the final loss function can be written as (D.6). In (D.6), we can find KL divergence constraints for speech and noise latent variables, which ensures that we can estimate the desired posterior distributions from noisy signal in a supervision way. This is also why our method can disentangle latent variables, and the noise can be estimated by nonlinear VAE rather than linear NMF, which is different from the previous VAE-based SE methods [13–17]. Moreover, in (D.6), $-\mathcal{L}$ can be used as a novel variational lower bound to perform supervised VAE training in other VAE-related applications. To better minimize (D.6), we introduce C-VAE and N-VAE to conduct joint training, which forms a Bayesian permutation training process between the three VAEs. Finally, the NS-VAE's training loss is

$$
\begin{aligned}
\mathcal{L}_{total} = &\mathcal{L}(\theta_y, \varphi_y, \theta_x, \varphi_x, \theta_d, \varphi_d; \mathbf{y}) \\
&+ \mathcal{L}_c(\theta_x, \varphi_x; \mathbf{x}) + \mathcal{L}_n(\theta_d, \varphi_d; \mathbf{d}),
\end{aligned}
\tag{D.7}
$$

where $\mathcal{L}_c(\theta_x, \varphi_x; \mathbf{x})$ and $\mathcal{L}_n(\theta_d, \varphi_d; \mathbf{d})$ are the general VAE loss function for speech and noise, which can be written as

$$
\begin{aligned}
\mathcal{L}_c(\theta_x, \varphi_x; \mathbf{x}) = &\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\{D_{KL}\left(p(\mathbf{z}_x|\mathbf{x})||q(\mathbf{z}_x)\right) \\
&- \mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{x})}[\log q(\mathbf{x}|\mathbf{z}_x)]\},
\end{aligned}
\tag{D.8}
$$

$$
\begin{aligned}
\mathcal{L}_n(\theta_d, \varphi_d; \mathbf{d}) = &\mathbb{E}_{\mathbf{d}\sim p(\mathbf{d})}\{D_{KL}\left(p(\mathbf{z}_d|\mathbf{d})||q(\mathbf{z}_d)\right) \\
&- \mathbb{E}_{\mathbf{z}_d\sim p(\mathbf{z}_d|\mathbf{d})}[\log q(\mathbf{d}|\mathbf{z}_d)]\}.
\end{aligned}
\tag{D.9}
$$

In (D.7), it can be found that the NS-VAE's training also includes the training of C-VAE and N-VAE, which improves NS-VAE's ability to disentangle latent variables. Minimizing $\mathcal{L}_{total}$ is our final target.

Fig. D.2 shows the proposed framework. To summarize, the proposed method includes a training and an enhancement stage. The whole training process can be described as follows: first, C-VAE and N-VAE are separately pre-trained without supervision using (D.8) and (D.9). Then, the LPS features of speech, noise and observed signal are separately used as the encoder input of C-VAE, N-VAE, and NS-VAE to estimate posterior distributions $p(\mathbf{z}_x|\mathbf{y}), p(\mathbf{z}_d|\mathbf{y}), p(\mathbf{z}_x|\mathbf{x}), p(\mathbf{z}_d|\mathbf{d})$ and prior distribution $q(\mathbf{y}|\mathbf{z}_x, \mathbf{z}_d)$, $q(\mathbf{x}|\mathbf{z}_x)$, $q(\mathbf{d}|\mathbf{z}_d)$. Finally, (D.7) is used as a loss function to perform related parameters update with the Adam algorithm [21]. The training is completed when the neural networks converge. In the online SE stage, we assume that the $\mathbf{z}_x$ sampled from $p(\mathbf{z}_x|\mathbf{x})$ is approximately equal to the sample $\mathbf{z}_x$ sampled from $p(\mathbf{z}_x|\mathbf{y})$. Therefore, we can separately use the NS-VAE encoder's two outputs as input of C-VAE and N-VAE to obtain the prior distributions $q(\mathbf{x}|\mathbf{z}_x)$ and $q(\mathbf{d}|\mathbf{z}_d)$. After that, using the reparameterization trick

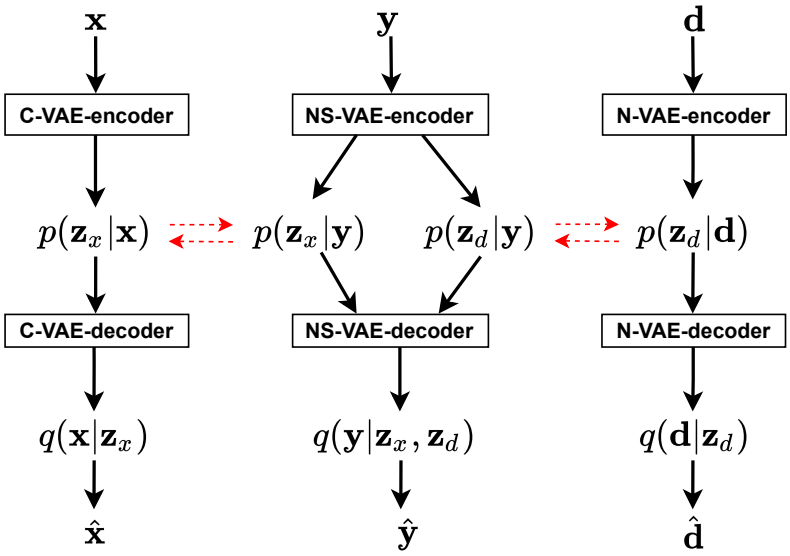


Fig. D.2: Model of Bayesian permutation training for SE.

and Monte Carlo estimate (MCE) [12], the speech and noise signal can be obtained. The enhanced speech is acquired by direct waveform reconstruction or post-filtering methods. This enhanced process is shown in Fig. D.3 (a).

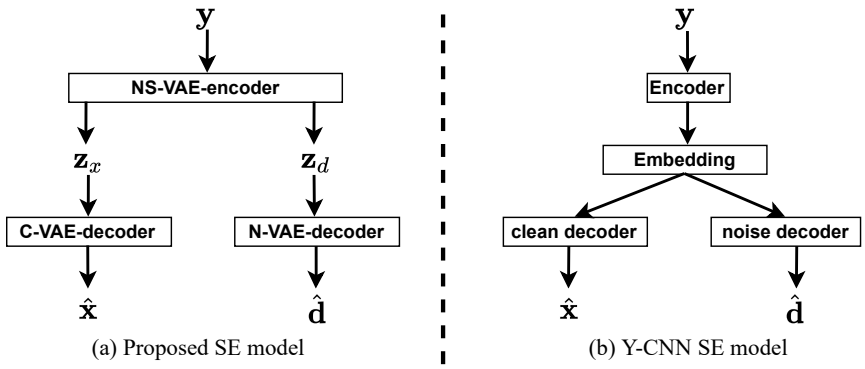
3.3 Calculation of Loss Function

In (D.7), the related posterior and prior distributions need to be determined, and $q(\mathbf{z}_x)$ and $q(\mathbf{z}_d)$ need to be predefined for the calculation. Here, for the simplicity of calculation, we assume that all the posterior and prior distributions are multivariate normal distributions with diagonal covariance [12], which is similar to the previous VAE-based SE methods [13–17]. For the NS-VAE, we have

$$\begin{aligned} p(\mathbf{z}_x|\mathbf{y}) &= \mathcal{N}\left(\mathbf{z}_x; \mu_{\theta_{yx}}(\mathbf{y}), \sigma^2_{\theta_{yx}}(\mathbf{y})\mathbf{I}\right), \\ p(\mathbf{z}_d|\mathbf{y}) &= \mathcal{N}\left(\mathbf{z}_d; \mu_{\theta_{yd}}(\mathbf{y}), \sigma^2_{\theta_{yd}}(\mathbf{y})\mathbf{I}\right), \\ q(\mathbf{y}|\mathbf{z}_x,\mathbf{z}_d) &= \mathcal{N}\left(\mathbf{y}; \mu_{\varphi_y}(\mathbf{z}_x,\mathbf{z}_d), \sigma^2_{\varphi_y}(\mathbf{z}_x,\mathbf{z}_d)\mathbf{I}\right), \end{aligned} \tag{D.10}$$

where $\mathbf{I}$ is the identity matrix. $\mu_{\theta_{yx}}(\mathbf{y}), \sigma^2_{\theta_{yx}}(\mathbf{y}), \mu_{\theta_{yd}}(\mathbf{y})$, $\sigma^2_{\theta_{yd}}(\mathbf{y})$ can be estimated by NS-VAE's encoder $G_{\theta_y}(\mathbf{y})$ with parameter $\theta_y = \{\theta_{yx}, \theta_{yd}\}$, and $\sigma^2_{\varphi_y}(\mathbf{z}_x,\mathbf{z}_d)$ and $\mu_{\varphi_y}(\mathbf{z}_x,\mathbf{z}_d)$ can be estimated by NS-VAE's decoder $G_{\varphi_y}(\mathbf{z}_x,\mathbf{z}_d)$ with parameter φ_y. Due to the space limitation and the fact that the frameworks of C-VAE and N-VAE are similar, we only give the details of C-VAE. Here, we have

$$\begin{aligned} p(\mathbf{z}_x|\mathbf{x}) &= \mathcal{N}\left(\mathbf{z}_x; \mu_{\theta_x}(\mathbf{x}), \sigma^2_{\theta_x}(\mathbf{x})\mathbf{I}\right) \\ q(\mathbf{x}|\mathbf{z}_x) &= \mathcal{N}\left(\mathbf{x}; \mu_{\varphi_x}(\mathbf{z}_x), \sigma^2_{\varphi_x}(\mathbf{z}_x)\mathbf{I}\right), \end{aligned} \tag{D.11}$$



(a) Proposed SE model (b) Y-CNN SE model

Fig. D.3: Enhancement framework comparison.

where $\mu_{\theta_x}(\mathbf{x}), \sigma^2_{\theta_x}(\mathbf{x})$ are obtained by C-VAE's encoder $G_{\theta_x}(\mathbf{x})$ with parameter θ_x, and $\mu_{\varphi_x}(\mathbf{z}_x)$, $\sigma^2_{\varphi_x}(\mathbf{z}_x)$ can be estimated by C-VAE's decoder $G_{\varphi_x}(\mathbf{z}_x)$ with parameter φ_x. $q(\mathbf{z}_d)$ and $q(\mathbf{z}_x)$ are pre-defined as a centered isotropic multivariate Gaussian $q(\mathbf{z}_x) = \mathcal{N}(\mathbf{z}_x; \mathbf{0}, \mathbf{I})$ and $q(\mathbf{z}_d) = \mathcal{N}(\mathbf{z}_d; \mathbf{0}, \mathbf{I})$. Finally, when all the distributions are determined, we can apply loss function (D.7) and the Adam algorithm to perform related parameters update for SE.

4 Experiment and Result Analysis

In this section, the proposed algorithm is evaluated. At first, we will use an example to verify that the proposed method can disentangle different latent variables from the observed signal. After that, an experiment will show the SE performance of our method.

Dataset: In this work, we use the TIMIT database [22], 100 environmental noises [23], DEMAND database [24] and NOISEX-92 database [25] to evaluate the performance of the proposed algorithm. In the training stage, the Babble, F16 noise from the NOISEX-92 database [25] and 90 environmental noise (N1–N90) [23] are used to conduct experiments. All 4620 utterances from the TIMIT database are corrupted by 92 types of noise at four different signal-to-noise ratio (SNR) levels, i.e., -5, 0, 5, and 10 dB. The utterances are randomly selected from these corrupted utterances, and they are connected to a 12-hour noisy speech database. Meanwhile, the corresponding speech and noise databases are also obtained. In the test stage, 200 utterances from the TIMIT test set, including 1680 utterances, are randomly chosen to build the test database. 13 types of noise (office [24], factory [25], and 10 unseen environmental noise (N91—N100) [23]) are randomly added to the 200 utterances at four SNR levels (i.e., -5, 0, 5, and 10 dB) to conduct experiment. In our experiments, all the signals are down-sampled to 16 kHz. The frame length is 512 samples with a frame shift of 256 samples.

Baseline: To evaluate the performance of the proposed method, we use a supervised SE model as a reference method (referred to Y-CNN) [26]. This is similar to the proposed method and can perform SE by direct waveform reconstruction [7] or estimated mask [8]. For a fair comparison, we use a convolutional neural network (CNN)

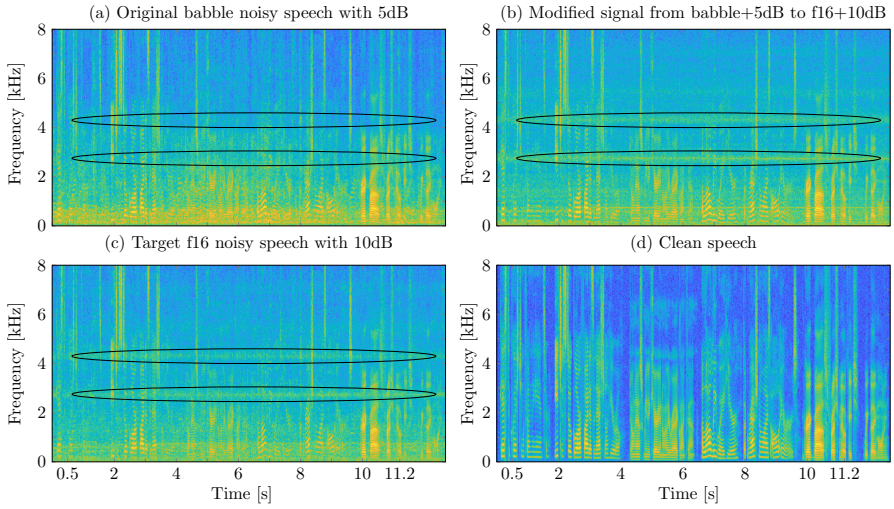


Fig. D.4: Experiment for disentangling latent variables.

to replace the original DNN [26] to improve its performance. Fig. D.3 shows the framework comparison of enhancement. Y-CNN has the same encoder and decoder as the proposed model. The only difference between Y-CNN and our method is the training loss function. The loss function of the proposed method applies deep representation learning (DRL) and reasonable assumptions to disentangle latent variables, which is not achieved by Y-CNN [26].

Experimental Setups: There are three VAEs in our proposed method. The C-VAE and N-VAE have the same structure. 1D CNN which is widely used in SE [9] is adopted in the experiment. C-VAE' encoder includes four hidden 1D convolutional layers. The number of channels in each layer is 32, 64, 128, and 256. The size of each convolving kernel is 3. The two output layers of the encoders are fully connected layers with 128 nodes. By using the reparameterization trick, the decoders' input size can also be set as 128. The decoder consists of four hidden 1D convolutional layers (the channel number of each layer is 256, 128, 64, and 32 with 3 kernel) and two fully connected output layers with 257 nodes. Moreover, the activation functions for the hidden and output layer are ReLU and linear activation function, respectively. For NS-VAE, its encoder also includes four 1D convolutional layers with ReLU as the activation function. The other parameter setting is the same as C-VAE. Additionally, its encoder has four output layers with 128 nodes and a linear activation function. The input size of the NS-VAE decoder is 256, which includes the latent speech and the noise variables. The decoder structure of NS-VAE's decoder is the same as that of C-VAE (Y-CNN's encoder and decoders only have one output layer with 128 and 257 nodes, respectively, because it does not disentangle latent variables. The other settings are the same as that of NS-VAE). In the training stage, all networks are trained by the Adam algorithm with a 128 mini-batch size. The learning rate is 0.001.

Experimental Results: Firstly, we will verify the ability of the proposed method to effectively disentangle the speech and noise latent variables from observed signals. Based on our assumption, the observed signal $\mathbf{y}$ is determined by $\mathbf{z}_x$ and $\mathbf{z}_d$. Thus, if we

Table D.1: Average SI-SDR comparison of different methods

SNR	SI-SDR (dB)		
	Noisy	Y-L	PVAE-L
-5	-5.67(± 0.22)	1.25(± 0.67)	2.84(± 0.72)
0	-0.69(± 0.22)	4.52(± 0.47)	6.32(± 0.48)
5	4.30(± 0.23)	6.76(± 0.29)	8.67(± 0.31)
10	7.30(± 0.23)	8.05(± 0.18)	10.03(± 0.23)

SNR	SI-SDR (dB)		
	Noisy	Y-M	PVAE-M
-5	-5.67(± 0.22)	2.04(± 0.68)	**4.01(± 0.88)**
0	-0.69(± 0.22)	7.40 (± 0.68)	**8.59(± 0.75)**
5	4.30(± 0.23)	11.74(± 0.62)	**12.33(± 0.61)**
10	7.30(± 0.23)	15.17(± 0.54)	**15.41(± 0.50)**

Table D.2: Average PESQ comparison of different methods

SNR	PESQ		
	Noisy	Y-L	PVAE-L
-5	1.43(± 0.02)	1.59(± 0.03)	**1.87(± 0.03)**
0	1.78(± 0.02)	2.02(± 0.02)	2.24(± 0.03)
5	2.13(± 0.02)	2.43(± 0.02)	2.57(± 0.02)
10	2.46(± 0.01)	2.76(± 0.02)	2.80(± 0.02)

SNR	PESQ		
	Noisy	Y-M	PVAE-M
-5	1.43(± 0.02)	1.68(± 0.03)	1.86(± 0.03)
0	1.78(± 0.02)	2.11(± 0.03)	**2.27(± 0.03)**
5	2.13(± 0.02)	2.53(± 0.03)	**2.63(± 0.02)**
10	2.46(± 0.01)	2.86(± 0.02)	**2.91(± 0.02)**

use different $\mathbf{z}_x$ or $\mathbf{z}_d$ as the input of NS-VAE's decoder, we can obtain the different ob-

Table D.3: Average STOI comparison of different methods

SNR	Noisy	Y-L	PVAE-L
-5	57.62($\pm$ 1.31)	57.63($\pm$ 1.67)	60.00($\pm$ 1.33)
0	70.02($\pm$ 1.24)	69.80($\pm$ 1.48)	70.68($\pm$ 1.12)
5	80.20($\pm$ 0.90)	79.20($\pm$ 1.18)	79.87($\pm$ 0.87)
10	86.32($\pm$ 0.50)	85.60 ($\pm$ 0.72)	84.32($\pm$ 0.54)

SNR	Noisy	Y-M	PVAE-M
-5	57.62($\pm$ 1.31)	59.72($\pm$ 1.70)	60.32($\pm$ 1.40)
0	70.02($\pm$ 1.24)	72.02($\pm$ 1.43)	71.75($\pm$ 1.19)
5	80.20($\pm$ 0.90)	81.96($\pm$ 1.02)	80.78($\pm$ 0.91)
10	86.32($\pm$ 0.50)	88.80($\pm$ 0.63)	87.24($\pm$ 0.58)

served signal. $\mathbf{z}_x$ and $\mathbf{z}_d$ can be acquired by different NS-VAE encoders. Fig. D.4 shows the experimental result. In this example, we first disentangle the latent variables of the observed signal (babble noise with 5dB). Then, we keep the speech latent variable $\mathbf{z}_x$ and replace the noise latent variable with another noise latent variable (f16 noise with 10dB). Finally, the new combination of latent variables is used as the input of NS-VAE's decoder to acquire the modified signal. Fig. D.4(b) shows the modified signal. Comparing Fig. D.4(a), (b), (c), and (d), it can be found that the modified signal successfully removes the babble noise character, and the original noise character is replaced by f16 noise character. (The modified signal has the constant noise around 3000 and 4000 Hz as shown in the black circle area, which is the same as the target signal.) Furthermore, the modified signal also preserves the original speech character. Therefore, this experiment indicates that the proposed method can effectively disentangle different latent variables.

In the second experiment, all algorithms are evaluated by the scale-invariant signal-to-distortion ratio (SI-SDR) in decibel (dB) [27], short-time objective intelligibility (STOI) [28], and perceptual evaluation of speech quality (PESQ) [29]. The enhanced speech is obtained by direct waveform reconstruction [7] or soft time-frequency mask estimation [26]. We use Y-M and Y-L to represent that the enhanced speech is acquired by mask estimation and direct waveform reconstruction using Y-CNN, respectively. Similarly, PVAE-L and PVAE-M denote that the enhanced speech is obtained by the proposed method using direct waveform reconstruction and mask estimation, respectively. Table. D.1, D.2, and D.3 show the PESQ, SI-SDR, and STOI comparisons with a 95% confidence interval. The results verify that our method can learn latent speech and noise variables from observed signals because PVAE-L achieves better PESQ and SI-SDR performance than Y-L. This means that C-VAE's decoder can recognize speech latent variables that are disentangled by NS-VAE. Moreover, PVAE-M significantly achieves better PESQ and SI-SDR performance than Y-M, which shows that our method can estimate a more accurate mask for SE. This result also illustrates that our approach has better noise

estimation performance than the reference method. Additionally, the results also show that Y-CNN's performance can be improved by the proposed loss function. Table. D.3 shows that the STOI score is competitive between the proposed and the reference algorithms. We think that the STOI score of the proposed method can be further improved by improving PVAE's disentangling performance [30]. Overall, PVAE-M achieves the best SE performance across the three evaluation criteria. Here, we only use a basic neural network to verify our algorithm. Its performance can be further improved by using more advanced neural networks and other speech features [2, 9, 10].

5 Conclusion

In this paper, a supervised Bayesian permutation training DRL method is proposed to disentangle latent speech and noise variables from the observed signal for SE. The proposed method is based on VAE and Bayesian theory. The experimental results show that our method cannot only successfully disentangle different latent variables but also obtain higher SI-SDR and PESQ scores than the state-of-the-art reference method. Moreover, the results also illustrate that the SE performance of the reference method can be improved by introducing the proposed DRL algorithm. In future work, some other strategies can be considered to further improve the disentangling performance of latent variables. In addition, the proposed method can also be applied in other speech generative tasks, e.g., voice conversion and ASR.

References

[1] Philipos C Loizou, *Speech enhancement: theory and practice*, CRC press, 2013.

[2] DeLiang Wang and Jitong Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 26, no. 10, pp. 1702–1726, 2018.

[3] Timo Gerkmann and Richard C Hendriks, "Unbiased MMSE-based noise power estimation with low complexity and low tracking delay," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 20, no. 4, pp. 1383–1393, 2011.

[4] Jesper Rindom Jensen, Jacob Benesty, and Mads Græsbøll Christensen, "Noise reduction with optimal variable span linear filters," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 24, no. 4, pp. 631–644, 2015.

[5] Yang Xiang, Liming Shi, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, and Mads Græsbøll Christensen, "An NMF-HMM speech enhancement method based on kullback-leibler divergence," in *Proc. Interspeech*, 2020, pp. 2667–2671.

[6] Mathew Shaji Kavalekalam, Jesper Kjær Nielsen, Liming Shi, Mads Græsbøll Christensen, and Jesper Boldt, "Online parametric NMF for speech enhancement," in *Proc. European Signal Processing Conf.*, 2018, pp. 2320–2324.

[7] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 23, no. 1, pp. 7–19, 2014.

[8] Yuxuan Wang, Arun Narayanan, and DeLiang Wang, "On training targets for supervised speech separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 22, no. 12, pp. 1849–1858, 2014.

[9] Yi Luo and Nima Mesgarani, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 27, no. 8, pp. 1256–1266, 2019.

[10] Andong Li, Wenzhe Liu, Xiaoxue Luo, Chengshi Zheng, and Xiaodong Li, "Icassp 2021 deep noise suppression challenge: Decoupling magnitude and phase optimization with a two-stage deep network," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2021, pp. 6628–6632.

[11] Yang Xiang and Changchun Bao, "A parallel-data-free speech enhancement method using multi-objective learning cycle-consistent generative adversarial network," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 28, pp. 1826–1838, 2020.

[12] Diederik P Kingma and Max Welling, "Auto-encoding variational Bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[13] Yoshiaki Bando, Masato Mimura, Katsutoshi Itoyama, Kazuyoshi Yoshii, and Tatsuya Kawahara, "Statistical speech enhancement based on probabilistic integration of variational autoencoder and non-negative matrix factorization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 716–720.

[14] Simon Leglaive, Laurent Girin, and Radu Horaud, "A variance modeling framework based on variational autoencoders for speech enhancement," in *Proc. IEEE Workshop Machine Learning. Signal Process.*, 2018, pp. 1–6.

[15] Simon Leglaive, Laurent Girin, and Radu Horaud, "Semi-supervised multichannel speech enhancement with variational autoencoders and non-negative matrix factorization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 101–105.

[16] Guillaume Carbajal, Julius Richter, and Timo Gerkmann, "Guided variational autoencoder for speech enhancement with a supervised classifier," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 681–685.

[17] Huajian Fang, Guillaume Carbajal, Stefan Wermter, and Timo Gerkmann, "Variational autoencoder for speech enhancement with a noise-aware encoder," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 676–680.

[18] Wei-Ning Hsu, Yu Zhang, and James Glass, "Learning latent representations for speech generation and transformation," *Proc. Interspeech*, pp. 1273–1277, 2017.

[19] Wei-Ning Hsu, Yu Zhang, and James Glass, "Unsupervised domain adaptation for robust speech recognition via variational autoencoder-based data augmentation," in *Proc. IEEE Workshop Automatic. Speech Recognition. and Understanding.*, 2017, pp. 16–23.

[20] Wei-Ning Hsu, Yu Zhang, and James Glass, "Unsupervised learning of disentangled and interpretable representations from sequential data," in *Proc. Advances in Neural Inform. Process. Syst.*, 2017, pp. 1876–1887.

[21] Diederik P Kingma and Jimmy Ba, "ADAM: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[22] John S Garofolo, Lori F Lamel, William M Fisher, Jonathan G Fiscus, and David S Pallett, "DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1," *NASA STI/Recon technical report n*, vol. 93, 1993.

[23] Guoning Hu and DeLiang Wang, "A tandem algorithm for pitch estimation and voiced speech segregation," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 18, no. 8, pp. 2067–2079, 2010.

[24] Joachim Thiemann, Nobutaka Ito, and Emmanuel Vincent, "The diverse environments multi-channel acoustic noise database (demand): A database of multichannel environmental noise recordings," in *Proceedings of Meetings on Acoustics ICA2013*. Acoustical Society of America, 2013, vol. 19, p. 035081.

[25] Andrew Varga and Herman JM Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 12, no. 3, pp. 247–251, 1993.

[26] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis, "Deep learning for monaural speech separation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 1562–1566.

[27] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R Hershey, "Sdr–half-baked or well done?," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 626–630.

[28] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, 2011.

[29] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2001, vol. 2, pp. 749–752.

[30] Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner, "Understanding disentangling in β-vae," *arXiv preprint arXiv:1804.03599*, 2018.

Paper E

A deep representation learning speech enhancement method using β-VAE

Yang Xiang, Jesper Lisby Højvang, Morten Højfeldt Rasmussen,
Mads Græsbøll Christensen

The paper has been published in the
Proc. Eur. Signal Process. Conf. (EUSIPCO) 2022

The layout has been revised.

Abstract

In previous work, we proposed a variational autoencoder-based (VAE) Bayesian permutation training speech enhancement (SE) method (PVAE) which indicated that the SE performance of the traditional deep neural network-based (DNN) method could be improved by deep representation learning (DRL). Based on our previous work, we in this paper propose to use β-VAE to further improve PVAE's ability of representation learning. More specifically, our β-VAE can improve PVAE's capacity of disentangling different latent variables from the observed signal without the trade-off problem between disentanglement and signal reconstruction. This trade-off problem widely exists in previous β-VAE algorithms. Unlike the previous β-VAE algorithms, the proposed β-VAE strategy can also be used to optimize the DNN's structure. This means that the proposed method can not only improve PVAE's SE performance but also reduce the number of PVAE training parameters. The experimental results show that the proposed method can acquire better speech and noise latent representation than PVAE. Meanwhile, it also obtains a higher scale-invariant signal-to-distortion ratio, speech quality, and speech intelligibility.

1 Introduction

The aim of speech enhancement (SE) is to remove background noise from the observed speech signal. In general, SE is mainly used to reduce the word error rate of the automatic speech recognition system [1] or improve speech quality and intelligibility for human listening [2]. Recently, with the wide application of online meeting systems, SE is required to reduce the WER for accurate live caption when providing high-quality speech audio under various complex noise conditions [3]. Thus, SE research is becoming more and more challenging.

During the past decades, many single-channel SE algorithms have been developed, including signal subspace methods [4], non-negative matrix factorization methods [5, 6], and codebook-based methods [7]. In recent years, deep neural networks (DNN) have shown great potential for SE [2, 8–14] because DNNs can use a non-linear process to model complex high-dimensional signals, which is more reasonable in practical applications [15]. Thus, DNN-based methods usually have a better SE performance than these previous linear models [4–7].

However, most of the regression-based SE algorithms [2, 8–10] do not consider applying DNNs to obtain better speech representations when conducting SE. Instead, they usually use DNNs to directly predict pre-defined targets for SE [2]. Although this approach can avoid inaccurate assumptions [8], it cannot ensure that these methods always work in environments with complex noise [2]. In general, deep representation learning (DRL) is important for DNN because DRL can obtain good signal representations in an unsupervised way and can, potentially, improve DNN's ability to extract useful information in complex environments [15, 16]. Additionally, a better signal representation usually leads to better predictions for DNNs [15]. Thus, DRL has a huge potential for DNN-based SE algorithms and makes them more robust. Moreover, the lack of a good DRL strategy may cause poor generalization of DNN-based SE algorithms [2, 15]. A good DRL algorithm can also disentangle various latent representations [15] of speech signals (e.g., speaker and phoneme information), which can also help DNNs achieve a better SE performance.

Recently, to improve traditional DNN's generalization ability, DRL-based SE algorithms are proposed [17–22]. The basic idea of these methods is that they use a variational autoencoder (VAE) [23] to learn speech representations when modeling speech, and apply a non-negative matrix factorization (NMF) to model noise. VAE is a DRL model and can perform efficient approximate posterior inference. Additionally, VAE can also learn the probability distribution of complex data. Thus, VAE is suitable for various speech generative tasks [23–25]. These VAE-based algorithms can effectively improve DNN's generalization ability, but they have difficulty obtaining good speech representations from the observed signal because they cannot disentangle speech representations from other latent representations [15, 17–22]. This causes the need to use a linear NMF to model noise, so their noise modeling ability is limited compared with these non-linear DNN-based methods [23]. And their SE performance is not always satisfactory in a complex noisy environment [18].

To obtain a better speech representation from the observed signal, a novel VAE-based SE method (named PVAE) is proposed [26]. This method applies an unsupervised method to learn signal representations and derives a novel VAE lower bound, which ensures that VAE can disentangle different latent variables from the observed signal. Compared to the previous VAE-based SE algorithms, PVAE can use non-linear DNNs to model noise, which improves the noise modeling ability. Additionally, this method can adopt various DNN structures [2], so the DNN-based SE algorithms [2] can be directly optimized by PVAE. This is not achieved by VAE-NMF-based algorithms [17–22]. The experimental results [26] indicate that the SE performance of the traditional DNN-based methods can be improved by introducing this PVAE-based DRL algorithm.

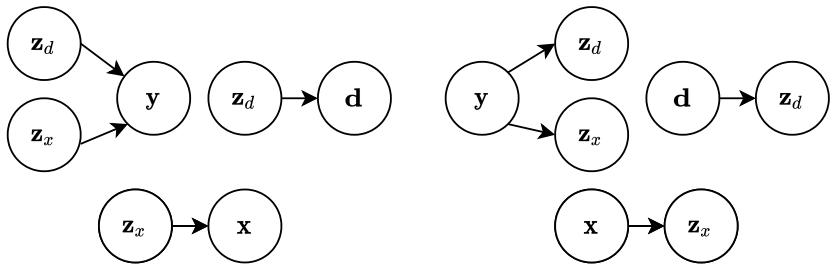
Inspired by previous works, in this paper we propose a novel β-VAE strategy to improve PVAE's representation learning and disentangling performance [15] with fewer DNN parameters. β-VAE [27, 28] is originally designed to push VAE to learn a more efficient latent representation of the data, which is disentangled if the data contains at least some underlying factors of variation [27]. However, in general, β-VAE has a trade-off problem [28]. A better disentanglement within the latent representations usually causes worse signal reconstruction. In this work, based on the VAE's application in SE [26], we propose a strategy to address this trade-off problem to obtain better speech and noise representation. As a result, our β-VAE can improve disentangling and representation performance without signal reconstruction loss. Moreover, the proposed β-VAE can also optimize the neural network structure of the original PVAE. This means that the proposed β-VAE (named β-PVAE) can possibly achieve a better SE performance with fewer training parameters compared to PVAE.

2 Related Work

Signal Model: in an additive noisy environment, using the short-time Fourier transform, the observed signal $y_{f,n} \in \mathbb{C}$, speech signal $x_{f,n} \in \mathbb{C}$, and noise $d_{f,n} \in \mathbb{C}$ can be written as

$$y_{f,n} = x_{f,n} + d_{f,n}, \tag{E.1}$$

where frequency bin $f \in [1, F]$ and time frame index $n \in [1, N]$. N and F denote the number of time frames and frequency bins, respectively. Their log-power spectrum (LPS) vector [8] at each frame can be represented as $\mathbf{y}$, $\mathbf{x}$, and $\mathbf{d}$, respectively, where



(a) Generative model (b) Recognition model

Fig. E.1: Graphical illustration of the proposed signal model.

we omit the frequency and time frame index for simplicity. In [26], we assume that $\mathbf{y}$ is generated from a random process involving the speech latent variables $\mathbf{z}_x \in \mathbb{R}^L$ and the noise latent variables $\mathbf{z}_d \in \mathbb{R}^L$. L is the dimension of latent variables. The latent variables $\mathbf{z}_x$ and $\mathbf{z}_d$ are independent. Similarly, $\mathbf{x}$ and $\mathbf{d}$ are independently generated by $\mathbf{z}_x$ and $\mathbf{z}_d$, respectively. Fig. E.1(a) shows the generative process. In [26], it is assumed that $\mathbf{z}_x$ and $\mathbf{z}_d$ can be estimated from speech and noise posterior distributions $p(\mathbf{z}_x|\mathbf{x})$ and $p(\mathbf{z}_d|\mathbf{d})$, respectively, and that they can also be estimated from the noisy speech posterior distributions $p(\mathbf{z}_x|\mathbf{y})$ and $p(\mathbf{z}_d|\mathbf{y})$. To disentangle latent variables, we assume that $p(\mathbf{z}_x, \mathbf{z}_d|\mathbf{y}) = p(\mathbf{z}_x|\mathbf{y})p(\mathbf{z}_d|\mathbf{y})$. Although this assumption is not always accurate in practical environments, it simplifies derivations, and helps us obtain a better signal model. Additionally, its effect towards signal estimation is not significant [26] (related analysis will be also given in Section 4). Fig. E.1(b) shows the recognition process.

VAE and β-VAE: the original VAE [23] defines a probabilistic generative process between the observed signal and its latent variables, and provides a principled method to jointly learn latent variables, generative and recognition models. The generative and recognition models are jointly trained by maximizing the evidence lower bound [23]

$$\begin{aligned} &\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}[\log q(\mathbf{y})] \geq -\mathcal{L}_n, \\ &\mathcal{L}_n = \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[D_{KL}\left(p(\mathbf{z}_y|\mathbf{y})\right)||q(\mathbf{z}_y)\right)] \\ &\qquad - \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\mathbb{E}_{\mathbf{z}_y\sim p(\mathbf{z}_y|\mathbf{y})}\left[\log q(\mathbf{y}|\mathbf{z}_y)\right]\right], \end{aligned} \tag{E.2}$$

where $D_{KL}(||)$ denotes the Kullback-Leibler (KL) divergence. $\mathbf{z}_y \in \mathbb{R}^L$ is the noisy latent variable. Maximizing this lower bound is equivalent to minimizing $\mathcal{L}_n$.

β-VAE [27] is a modification of the original VAE framework, which introduces an adjustable hyperparameter β in the KL divergence term:

$$\begin{aligned} \mathcal{L}_n = &\beta\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[D_{KL}\left(p(\mathbf{z}_y|\mathbf{y})\right)||q(\mathbf{z}_y)\right)] \\ &- \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\mathbb{E}_{\mathbf{z}_y\sim p(\mathbf{z}_y|\mathbf{y})}\left[\log q(\mathbf{y}|\mathbf{z}_y)\right]\right]. \end{aligned} \tag{E.3}$$

In general, $\beta > 1$ results in more disentangled latent representations [27]. Higher values of β can encourage learning a more disentangled representation. However, β-VAE usually has a trade-off problem between the latent representation disentanglement and signal reconstruction.

Bayesian permutation training VAE (PVAE) for SE: Although the VAE-based algorithms [23, 27] can learn signal representations and disentangle latent representations

in a self-supervised way, their performance is limited when disentangling desired latent representations for SE application. Therefore, a Bayesian permutation training VAE (PVAE) [26] is proposed for SE. PVAE is a semi-supervised DRL method, which introduces multiple latent variables in VAE and disentangles them in a semi-supervised way. Fig. E.2 shows the PVAE framework. It can be seen that PVAE includes three VAE structures: clean speech VAE (C-VAE), noise VAE (N-VAE), and noisy VAE (NS-VAE). C-VAE and N-VAE are trained without supervision to obtain speech and noise latent representations and their posterior estimates $p(\mathbf{z}_x|\mathbf{x})$, $p(\mathbf{z}_d|\mathbf{d})$, respectively. This is achieved by minimizing the following VAE loss function:

$$\begin{aligned}\mathcal{L}_c(\theta_x,\varphi_x;\mathbf{x}) = \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\{&D_{KL}\left(p(\mathbf{z}_x|\mathbf{x})||q(\mathbf{z}_x)\right)\\ &-\mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{x})}[\log q(\mathbf{x}|\mathbf{z}_x)]\},\end{aligned} \tag{E.4}$$

$$\begin{aligned}\mathcal{L}_d(\theta_d,\varphi_d;\mathbf{d}) = \mathbb{E}_{\mathbf{d}\sim p(\mathbf{d})}\{&D_{KL}\left(p(\mathbf{z}_d|\mathbf{d})||q(\mathbf{z}_d)\right)\\ &-\mathbb{E}_{\mathbf{z}_d\sim p(\mathbf{z}_d|\mathbf{d})}[\log q(\mathbf{d}|\mathbf{z}_d)]\},\end{aligned} \tag{E.5}$$

where $\theta_x, \varphi_x, \theta_d, \varphi_d$ are the DNN parameters for the related probability estimation [26]. Additionally, NS-VAE is trained under the supervision of C-VAE and N-VAE's encoders. Based on the derivation in [26], the NS-VAE's training loss function can be written as

$$\begin{aligned}&\mathcal{L}_p(\theta_y,\varphi_y;\mathbf{y})\\ &=\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y}),\mathbf{x}\sim\mathbf{p}(\mathbf{x})}\{D_{KL}\left(p(\mathbf{z}_x|\mathbf{y})||p(\mathbf{z}_x|\mathbf{x})\right)\\ &\quad+\mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{y})}[\log\frac{p(\mathbf{z}_x|\mathbf{x})}{q(\mathbf{z}_x)}]\}\\ &\quad+\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y}),\mathbf{d}\sim p(\mathbf{d})}\{D_{KL}\left(p(\mathbf{z}_d|\mathbf{y})||p(\mathbf{z}_d|\mathbf{d})\right)\\ &\quad+\mathbb{E}_{\mathbf{z}_d\sim p(\mathbf{z}_d|\mathbf{y})}[\log\frac{p(\mathbf{z}_d|\mathbf{d})}{q(\mathbf{z}_d)}]\}\\ &\quad-\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\mathbb{E}_{\mathbf{z}_d,\mathbf{z}_x\sim p(\mathbf{z}_d,\mathbf{z}_x|\mathbf{y})}\left[\log q(\mathbf{y}|\mathbf{z}_x,\mathbf{z}_d)\right]\right],\end{aligned} \tag{E.6}$$

where θ_y, φ_y are the NS-VAE's network parameters.

In the online SE stage, we assume that the $\mathbf{z}_x, \mathbf{z}_d$ sampled from $p(\mathbf{z}_x|\mathbf{x})$ and $p(\mathbf{z}_d|\mathbf{d})$ are approximately equal to the sample $\mathbf{z}_x$, $\mathbf{z}_d$ sampled from $p(\mathbf{z}_x|\mathbf{y})$, $p(\mathbf{z}_d|\mathbf{y})$, respectively. So, we separately use the NS-VAE encoder's two outputs as input of C-VAE and N-VAE to estimate related signals for SE.

3 β-VAE-based Speech Enhancement

Inspired by β-VAE, we propose a novel β-VAE strategy (named β-PVAE) to further improve PVAE's SE performance. More specifically, β-VAE is used to improve PVAE's representation learning ability that can better disentangle speech and noise latent variables from the observed signal, which can help PVAE obtain better SE performance. In PVAE, all the PVAE's decoders are trained in an unsupervised way [26]. The accuracy of the restored signal depends on the quality of latent representations. This means that the SE performance in PVAE is determined by the quality of speech and noise latent variables.

In [26], we derived a novel evidence lower bound (ELBO) ($\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}[\log q(\mathbf{y})] \geq -\mathcal{L}_p$). Additionally, β-VAE [27] applies an adjustable hyperparameter β in original

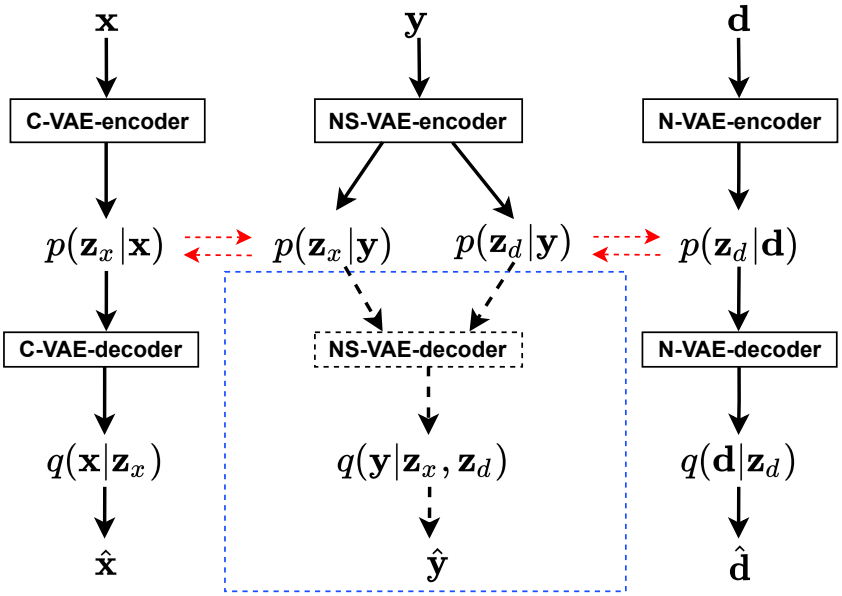


Fig. E.2: Model illustration for PVAE and β-PVAE.

VAE's [23] KL divergence term. Following β-VAE's property and PVAE's derivation [26], we apply this hyperparameter in the derived ELBO [26], the (E.6) can be written as

$$\begin{aligned}
&\mathcal{L}_p(\theta_y, \varphi_y; \mathbf{y}) \\
&\quad = \beta \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y}), \mathbf{x}\sim \mathbf{p}(\mathbf{x})} \{ D_{KL}\left(p(\mathbf{z}_x|\mathbf{y}) || p(\mathbf{z}_x|\mathbf{x})\right) \\
&\qquad + \mathbb{E}_{\mathbf{z}_x \sim p(\mathbf{z}_x|\mathbf{y})} [\log \frac{p(\mathbf{z}_x|\mathbf{x})}{q(\mathbf{z}_x)}] \} \\
&\qquad + \beta \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y}), \mathbf{d}\sim p(\mathbf{d})} \{ D_{KL}\left(p(\mathbf{z}_d|\mathbf{y}) || p(\mathbf{z}_d|\mathbf{d})\right) \\
&\qquad + \mathbb{E}_{\mathbf{z}_d \sim p(\mathbf{z}_d|\mathbf{y})} [\log \frac{p(\mathbf{z}_d|\mathbf{d})}{q(\mathbf{z}_d)}] \} \\
&\qquad - \alpha \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})} \left[\mathbb{E}_{\mathbf{z}_d, \mathbf{z}_x \sim p(\mathbf{z}_d, \mathbf{z}_x|\mathbf{y})} \left[\log q(\mathbf{y}|\mathbf{z}_x, \mathbf{z}_d) \right] \right].
\end{aligned} \tag{E.7}$$

In (E.7), we introduce a hyperparameter α in the restoration term. The purpose is to better analyze β-VAE [27] in PVAE. Note, α will not generate any effects for the original β-VAE's property because what is important in (E.7) is the weight ratio $\beta : \alpha$. This weight ratio can also be written as: $\gamma = \beta : \alpha = (\beta/\alpha) : 1$, which is equal to the original β-VAE's loss function in (E.3). β-VAE [27] indicates that a higher value of β encourages VAE learning a more disentangled representation. Thus, we hypothesize that a higher value of $\beta : \alpha$ in (E.7) can cause a better disentangling performance for speech and noise latent variables. This point will be verified by later experiments.

β-VAE usually has a trade-off problem between the disentanglement and signal reconstruction [27], which means that a good disentangled representation usually leads to poor signal reconstruction performance. In NS-VAE (as shown in Fig. E.2), this trade-off is between the quality of observed signal reconstruction and the disentanglement of

speech and noise latent variables. In SE application, we only need NS-VAE's disentanglement function, observed signal reconstruction is not useful (dashed part in Fig. E.2). This means that we should set a very high weight ratio γ to obtain a better disentanglement performance [27]. Ideally, $\gamma \to +\infty$. One strategy to achieve this purpose is to set $\alpha = 0$, so the loss function (E.7) can be rewritten as

$$\begin{aligned}\mathcal{L}_\beta(\theta_y; \mathbf{y}) = \beta\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y}),\mathbf{x}\sim\mathbf{p}(\mathbf{x})}\{&D_{KL}\left(p(\mathbf{z}_x|\mathbf{y})||p(\mathbf{z}_x|\mathbf{x})\right) \\ &+ \mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{y})}[\log\frac{p(\mathbf{z}_x|\mathbf{x})}{q(\mathbf{z}_x)}]\} \\ &+ \beta\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y}),\mathbf{d}\sim p(\mathbf{d})}\{D_{KL}\left(p(\mathbf{z}_d|\mathbf{y})||p(\mathbf{z}_d|\mathbf{d})\right) \\ &+ \mathbb{E}_{\mathbf{z}_d\sim p(\mathbf{z}_d|\mathbf{y})}[\log\frac{p(\mathbf{z}_d|\mathbf{d})}{q(\mathbf{z}_d)}]\}.\end{aligned} \tag{E.8}$$

In (E.8), it can be found that there is no reconstruction term. This means that we do not need to train the NS-VAE's decoder, which reduces the PVAE's training parameters. The dashed part in Fig. E.2 is removed in the proposed β-PVAE framework. Comparing the PVAE and proposed β-PVAE, we can find that the β-VAE can be used to optimize the PVAE's network structure and β-PVAE also addresses the β-VAE's trade-off problem for SE application. All in all, the combination of β-VAE and PVAE can not only improve PVAE's disentanglement performance, but also simplify its framework.

To summarize, the proposed β-PVAE includes a training and an enhancement stage for the SE application, which is similar to PVAE [26]. In the training stage, C-VAE and N-VAE are separately pre-trained by self-supervision using (E.4) and (E.5). After that, we apply (E.8) to train NS-VAE. In the enhancement stage, we can separately use the NS-VAE encoder's two outputs as input of C-VAE and N-VAE to obtain the prior distributions $q(\mathbf{x}|\mathbf{z}_x)$ and $q(\mathbf{d}|\mathbf{z}_d)$ for SE. Moreover, to calculate (E.8), related prior and posterior distributions need to be determined. Here, all the estimations of these distributions are the same as PVAE. More details can be found in [26].

4 Experiments

In this section, we report two experiments. First, we will investigate the disentanglement ability of the latent variables in the proposed algorithm. In addition, β-PVAE's SE performance will be indicated.

Datasets: In this work, we use the DNS challenge 2021 corpus [29] to evaluate the performance of the proposed algorithm. We select English speakers and randomly split 70% of speakers for training, 20% for validation, and 10% for evaluation. Then, all the noise from the DNS noise corpus are randomly divided into training, validation, and test noise in a proportion similar to that used for speech utterances. Next, the corresponding training, validation, and test corpus for speech and noise are randomly mixed using DNS script [29] with random signal-to-noise ratio (SNR) levels (SNR range is from –10dB to 15dB). Other parameters of signal mixing are the default values in the DNS script [29]. Finally, we randomly choose 20 hours mixed training utterances, 5 hours mixed validation utterances, and 1 hour mixed test utterances to build experimental dataset. All signals are down-sampled to 16 kHz.

Experimental settings: In the experiments, the neural structures for C-VAE and N-VAE are the same. Their encoders include four hidden 1D convolutional layers [11]. The

Table E.1: Average STOI, PESQ, and SI-SDR comparison for β-PVAE under different γ with a 95% confidence interval (β-PVAE is equal to PVAE when $\gamma = 1$)

Method	STOI	PESQ	SI-SDR
Noisy	88.94($\pm$ 1.77)	2.29($\pm$ 0.02)	8.36($\pm$ 1.13)
Oracle	98.12($\pm$ 0.35)	4.19($\pm$ 0.00)	19.84($\pm$ 0.92)
PVAE ($\gamma = 1$)	89.33($\pm$ 1.72)	2.59($\pm$ 0.03)	10.31($\pm$ 1.03)
$\gamma = 2$	89.81($\pm$ 1.67)	2.69($\pm$ 0.02)	11.84($\pm$ 0.97)
$\gamma = 5$	89.76($\pm$ 1.64)	2.70($\pm$ 0.02)	12.23($\pm$ 0.93)
$\gamma = 10$	89.94($\pm$ 1.70)	2.71($\pm$ 0.02)	12.31($\pm$ 0.94)
$\gamma = 100$	89.98($\pm$ 1.70)	2.72($\pm$ 0.02)	12.45($\pm$ 0.94)
$\gamma = 1000$	90.02($\pm$ 1.71)	2.74($\pm$ 0.01)	12.55($\pm$ 0.94)
$\gamma = +\infty$	90.05($\pm$ 1.71)	2.75($\pm$ 0.01)	13.20($\pm$ 0.95)

number of channels in each layer is 32, 64, 128, and 256. The size of each convolving kernel is 3. The two output layers of the encoders are fully connected layers with 128 nodes. Their decoders consist of four hidden 1D convolutional layers (the channel number of each layer is 256, 128, 64, and 32 with 3 kernel) and two fully connected output layers with 257 nodes. For NS-VAE, its encoder's hidden layer setting is the same as C-VAE. NS-VAE's encoder has four output layers with 128 nodes. For C-VAE, N-VAE, and NS-VAE, their activation functions in the hidden and output layer are ReLU and linear activation function, respectively. All networks are trained by the Adam algorithm with a 128 mini-batch size.

Experimental results: To evaluate the SE performance of various algorithms, we will use scale-invariant signal-to-distortion ratio (SI-SDR) in decibel (dB) [30], short-time objective intelligibility (STOI) [31], and perceptual evaluation of speech quality (PESQ) [32] as evaluation metrics.

First, we will investigate β-PVAE's performance in disentangling speech and noise latent variables. Based on our previous derivation and analysis [26], β-PVAE's SE performance is determined by disentanglement performance. Table. E.1 'Oracle' shows the SE performance with a 95% confidence interval if latent variables are completely disentangled. Here, the signals are reconstructed by mask estimation [9]. The complete disentanglement means that they have the same posterior forms: $p(\mathbf{z}_x|\mathbf{x}) = p(\mathbf{z}_x|\mathbf{y})$ and $p(\mathbf{z}_d|\mathbf{d}) = p(\mathbf{z}_d|\mathbf{y})$. This is because $p(\mathbf{z}_x|\mathbf{x})$ and $p(\mathbf{z}_d|\mathbf{d})$ are learned in an unsupervised way with speech or noise only, which ensures that their latent representation only contains speech or noise representation. 'Oracle' results indicate that β-PVAE achieves a very satisfactory SE performance in SI-SDR, STOI, and PESQ, which shows the importance of disentangling latent variable for achieving excellent SE performance. The NS-VAE's purpose is to disentangle different representations from the observed signal and obtain the closest possible speech and noise posterior. Next, we use KL divergence to evaluate the practical disentanglement performance in latent space. A better disentanglement can lead to a lower KL divergence ($D_{KL}\left(p(\mathbf{z}_d|\mathbf{y})||p(\mathbf{z}_d|\mathbf{d})\right)$ and

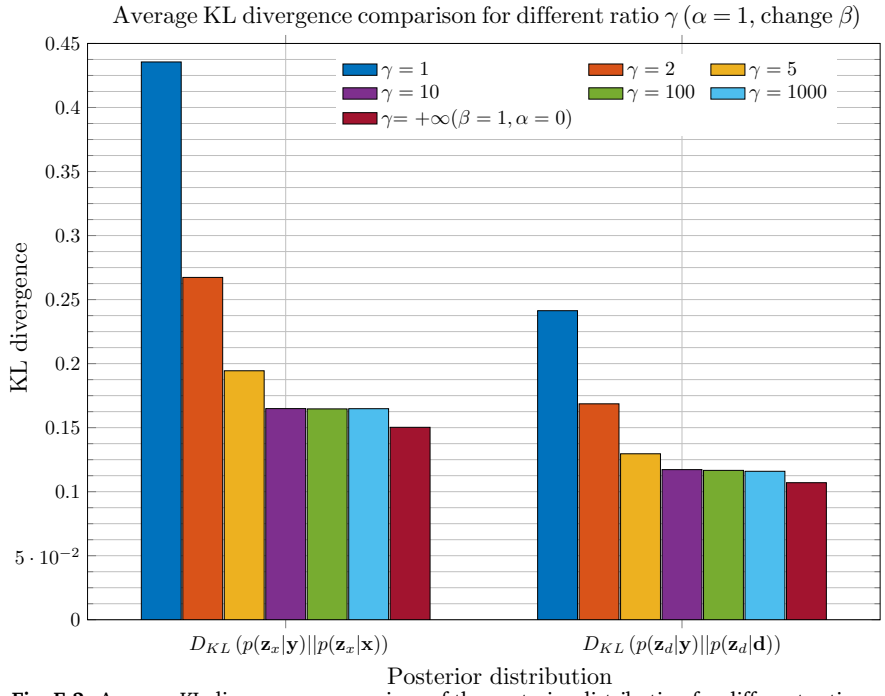


Fig. E.3: Average KL divergence comparison of the posterior distribution for different ratio γ.

$D_{KL}\left(p(\mathbf{z}_x|\mathbf{y})||p(\mathbf{z}_x|\mathbf{x})\right)$). Fig. E.3 shows the average KL divergence comparison of validation samples for using different ratios $\gamma = \beta : \alpha$ in loss function (E.7) to train NS-VAE. In (E.7), we keep $\alpha = 1$ and change different β to determine ratio γ, and $\gamma = +\infty$ means that $\alpha = 0, \beta = 1$, which is equal to (E.8). In Fig. E.3, we see that the KL divergence decreases with the increase of γ for both speech and noise latent variables, which means that the disentangled posteriors get closer to the true posteriors and the NS-VAE achieves a better disentanglement performance. When NS-VAE's decoder is removed ($\gamma = +\infty, \alpha = 0, \beta = 1$), NS-VAE can acquire the best posterior estimation. This verifies our hypothesis and deduction in Section 3. Additionally, although we have an inaccurate posterior conditional assumption $p(\mathbf{z}_x, \mathbf{z}_d|\mathbf{y}) = p(\mathbf{z}_x|\mathbf{y})p(\mathbf{z}_d|\mathbf{y})$, Fig. E.3 shows that NS-VAE can still estimate a satisfactory posterior with a low KL divergence. However, this inaccurate assumption may hinder NS-VAE from obtaining a lower KL divergence when $\gamma = +\infty$.

Next, we will evaluate the SE performance of the proposed β-PVAE. We use basic PVAE [26] as the reference method, which can be more direct to find the effects of β-VAE for the previous PVAE. The enhanced speech is obtained by mask estimation [9]. Table. E.1 shows the experimental results. We find that β-PVAE achieves a very significant STOI, PESQ, and SI-SDR improvement over PVAE (from $\beta = 1$ to $\beta = 2$). This indicates that good disentanglement performance in latent space can directly lead to an improvement in speech quality and intelligibility. In addition, β-PVAE achieves the best SE performance when $\beta = +\infty$. This illustrates that the proposed β-PVAE can effectively improve PVAE's SE performance with a simpler network structure.

5 Conclusions

In this paper, a β-PVAE-based SE method is proposed to improve previous PVAE's SE performance. More specifically, β-PVAE can improve PVAE's ability to disentangle speech and noise latent variables from the observed signal. In addition, based on VAE's application in SE, the proposed β-PVAE addresses the trade-off problem between disentanglement and signal reconstruction, which widely exists in β-VAE. Compared with the previous PVAE algorithm, β-PVAE also simplifies its neural network and reduces the number of training parameters when improving the SE performance. Experimental results indicate that a good signal representation can achieve a very satisfactory SE performance. Moreover, β-PVAE obtains a better disentanglement performance and achieves higher SI-SDR, PESQ, and STOI scores than PVAE. In future work, we believe that β-PVAE can achieve better SE performance by improving the latent space disentanglement performance or the decoder's signal reconstruction ability.

References

[1] Jinyu Li, Li Deng, Yifan Gong, and Reinhold Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 22, no. 4, pp. 745–777, 2014.

[2] DeLiang Wang and Jitong Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 26, no. 10, pp. 1702–1726, 2018.

[3] Sefik Emre Eskimez, Xiaofei Wang, Min Tang, Hemin Yang, Zirun Zhu, Zhuo Chen, Huaming Wang, and Takuya Yoshioka, "Human Listening and Live Captioning: Multi-Task Training for Speech Enhancement," in *Proc. Interspeech*, 2021, pp. 2686–2690.

[4] Jesper Rindom Jensen, Jacob Benesty, and Mads Græsbøll Christensen, "Noise reduction with optimal variable span linear filters," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 24, no. 4, pp. 631–644, 2015.

[5] Yang Xiang, Liming Shi, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, and Mads Græsbøll Christensen, "A novel NMF-HMM speech enhancement algorithm based on poisson mixture model," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2021, pp. 721–725.

[6] Yang Xiang, Liming Shi, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, and Mads Græsbøll Christensen, "An NMF-HMM speech enhancement method based on kullback-leibler divergence," in *Proc. Interspeech*, 2020, pp. 2667–2671.

[7] Mathew Shaji Kavalekalam, Jesper Kjær Nielsen, Jesper Bünsow Boldt, and Mads Græsbøll Christensen, "Model-based speech enhancement for intelligibility improvement in binaural hearing aids," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 27, no. 1, pp. 99–113, 2018.

[8] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 23, no. 1, pp. 7–19, 2014.

[9] Yuxuan Wang, Arun Narayanan, and DeLiang Wang, "On training targets for supervised speech separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 22, no. 12, pp. 1849–1858, 2014.

[10] Zhong-Qiu Wang, Gordon Wichern, and Jonathan Le Roux, "On the compensation between magnitude and phase in speech separation," *IEEE Signal Processing Letters*, vol. 28, pp. 2018–2022, 2021.

[11] Yi Luo and Nima Mesgarani, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 27, no. 8, pp. 1256–1266, 2019.

[12] Yang Xiang and Changchun Bao, "A parallel-data-free speech enhancement method using multi-objective learning cycle-consistent generative adversarial network," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 28, pp. 1826–1838, 2020.

[13] Yanxin Hu, Yun Liu, Shubo Lv, Mengtao Xing, Shimin Zhang, Yihui Fu, Jian Wu, Bihong Zhang, and Lei Xie, "DCCRN: Deep Complex Convolution Recurrent Network for Phase-Aware Speech Enhancement," in *Proc. Interspeech*, 2020, pp. 2472–2476.

[14] Andong Li, Wenzhe Liu, Xiaoxue Luo, Chengshi Zheng, and Xiaodong Li, "Icassp 2021 deep noise suppression challenge: Decoupling magnitude and phase optimization with a two-stage deep network," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2021, pp. 6628–6632.

[15] Yoshua Bengio, Aaron Courville, and Pascal Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[16] Yuying Xie, Thomas Arildsen, and Zheng-Hua Tan, "Disentangled speech representation learning based on factorized hierarchical variational autoencoder with self-supervised objective," in *proc. IEEE International Workshop on Machine Learning for Signal Processing*, 2021, pp. 1–6.

[17] Simon Leglaive, Laurent Girin, and Radu Horaud, "A variance modeling framework based on variational autoencoders for speech enhancement," in *Proc. IEEE Workshop Machine Learning. Signal Process.*, 2018, pp. 1–6.

[18] Yoshiaki Bando, Masato Mimura, Katsutoshi Itoyama, Kazuyoshi Yoshii, and Tatsuya Kawahara, "Statistical speech enhancement based on probabilistic integration of variational autoencoder and non-negative matrix factorization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 716–720.

[19] Simon Leglaive, Laurent Girin, and Radu Horaud, "Semi-supervised multichannel speech enhancement with variational autoencoders and non-negative matrix factorization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 101–105.

[20] Guillaume Carbajal, Julius Richter, and Timo Gerkmann, "Guided variational autoencoder for speech enhancement with a supervised classifier," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 681–685.

[21] Huajian Fang, Guillaume Carbajal, Stefan Wermter, and Timo Gerkmann, "Variational autoencoder for speech enhancement with a noise-aware encoder," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 676–680.

[22] G. Carbajal, J. Richter, and T. Gerkmann, "Disentanglement learning for variational autoencoders applied to audio-visual speech enhancement," in *Proc. IEEE Workshop Appl. of Signal Process. to Aud. and Acoust.* IEEE, 2021, pp. 126–130.

[23] Diederik P Kingma and Max Welling, "Auto-encoding variational Bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[24] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," *arXiv preprint arXiv:2106.06103*, 2021.

[25] Yongmao Zhang, Jian Cong, Heyang Xue, Lei Xie, Pengcheng Zhu, and Mengxiao Bi, "Visinger: Variational inference with adversarial learning for end-to-end singing voice synthesis," *arXiv preprint arXiv:2110.08813*, 2021.

[26] Yang Xiang, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, and Mads Græsbøll Christensen, "A bayesian permutation training deep representation learning method for speech enhancement with variational autoencoder," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2022, pp. 381–385.

[27] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," *International Conference on Learning Representations*, 2017.

[28] Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner, "Understanding disentangling in β-vae," *arXiv preprint arXiv:1804.03599*, 2018.

[29] Chandan KA Reddy, Harishchandra Dubey, Kazuhito Koishida, Arun Nair, Vishak Gopal, Ross Cutler, Sebastian Braun, Hannes Gamper, Robert Aichner, and Sriram Srinivasan, "Interspeech 2021 deep noise suppression challenge," in *Proc. Interspeech*, 2021.

[30] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R Hershey, "Sdr–half-baked or well done?," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 626–630.

[31] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, 2011.

[32] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2001, vol. 2, pp. 749–752.

Paper F

A Two-Stage Deep Representation Learning-Based Speech Enhancement Method Using Variational Autoencoder and Adversarial Training

Yang Xiang, Jesper Lisby Højvang, Morten Højfeldt Rasmussen,
Mads Græsbøll Christensen

The paper has been submitted to
IEEE/ACM Trans. Audio,Speech, and Lang. Process, 2022

The layout has been revised.

Abstract

This paper focuses on leveraging deep representation learning (DRL) for speech enhancement (SE). In general, the performance of the deep neural network (DNN) is heavily dependent on the learning of data representation. However, the DRL's importance is often ignored in many DNN-based SE algorithms. To obtain a higher quality enhanced speech, we propose a two-stage DRL-based SE method through adversarial training. In the first stage, we disentangle different latent variables because disentangled representations can help DNN generate a better enhanced speech. Specifically, we use the β-variational autoencoder (VAE) algorithm to obtain the speech and noise posterior estimations and related representations from the observed signal. However, since the posteriors and representations are intractable and we can only apply a conditional assumption to estimate them, it is difficult to ensure that these estimations are always pretty accurate, which may potentially degrade the final accuracy of the signal estimation. To further improve the quality of enhanced speech, in the second stage, we introduce adversarial training to reduce the effect of the inaccurate posterior towards signal reconstruction and improve the signal estimation accuracy, making our algorithm more robust for the potentially inaccurate posterior estimations. As a result, better SE performance can be achieved. The experimental results indicate that the proposed strategy can help similar DNN-based SE algorithms achieve higher short-time objective intelligibility (STOI), perceptual evaluation of speech quality (PESQ), and scale-invariant signal-to-distortion ratio (SI-SDR) scores. Moreover, the proposed algorithm can also outperform recent competitive SE algorithms.

1 Introduction

In real-world environments, speech signals are usually degraded by various environmental noise. To counter these degradations, speech enhancement (SE) techniques have been developed during the past decades [1]. The main purpose of SE is to remove background noise from an observed signal and improve speech quality and intelligibility in a noisy environment. SE has been widely applied in speech coding, teleconferencing, hearing aids, mobile communication, and robust automatic speech recognition (ASR) [2]. Due to the recent COVID-19 pandemic, there has been an increasing need for online meeting systems [3], where SE can help the system to reduce the word error rate (WER) for accurate live captioning when transmitting high-quality speech audio in various complex-noise conditions [4, 5]. Therefore, SE is an increasingly prominent research topic.

There is a considerable amount of literature published on SE algorithms. Classic SE methods include signal subspace methods [6–8], codebook-based methods [9–11], and non-negative matrix factorization (NMF) methods [11–14]. Most of these methods perform SE by applying short-time Fourier transform (STFT) to analyze the time–frequency (T–F) representation of the observed signal or directly using waveform. Recently, with the development of deep neural network (DNN) techniques, DNNs have shown a great potential for SE [15–23]. Unlike classic algorithms [7, 10–14], DNNs can learn the disentangled representations of the data [24], and can use the learned representations to generate related data. Thus, we hypothesize that one of the reason of why DNN can perform SE is that DNN can extract useful speech representation [25] from the observed signal and generate corresponding speech data. DNNs' advantage for SE is

that DNN can extract underlying information (e.g., phoneme or emotional information) from high dimension features [26–29]. Moreover, DNN can also represent the different underlying information by different vector forms, and can disentangle different information. As a result, DNNs can effectively analyze more signal representations and achieve a better SE performance. Additionally, one of the DNNs' principle is that DNNs are based on data representation learning [24, 30, 31], so it can avoid the speech-phase estimation problem (only DNN's input contains the all signal information) [32–34] in traditional T–F processes (STFT analysis). More specifically, recent research [34] has indicated that DNN can directly leverage the speech waveform to achieve excellent SE performance [35]. Furthermore, compared to T–F representations, DNNs can easily combine different information to perform the signal analysis (find underlying relationships of different signals), so the audio–visual-based SE has also been developed in recent years [36, 37].

Currently, although DNNs have significantly promoted the development of SE techniques [17], there are still some problems in DNN-based SE algorithms. The DNNs' potential for SE is not completely explored. For example, most of the present DNN-based SE methods [15–17, 19–23, 38] focus on the learning of the training target and apply DNNs only to predict pre-defined targets (e.g., various masks [16], speech spectrum [38], and speech present probability [39]). However, these algorithms ignore the importance of reliable representations for DNN-based methods [30] and do not consider using DNN to obtain better signal representations. Although direct prediction of pre-defined targets can prevent inaccurate signal assumptions [38], the lack of a good representation learning strategy means that these algorithms do not achieve constant satisfactory SE performance in complex noisy environments [17]. On the contrary, an efficient deep representation learning (DRL) method may not only improve DNNs' ability to extract useful information in complex environments [29, 30] but can also lead to a better prediction ability of the DNN [30]. Moreover, a good representation can place less demand on the learning machine in order to perform a task successfully [17]. Therefore, DRL has potential to help DNN-based SE algorithms improve their robustness and generalization ability [25, 30]. Furthermore, DRL can disentangle different latent representations of the speech signal (e.g., content and acoustic representation) [26–28], so more related information (e.g., phonetic information of a speech signal) can be included to analyze the speech signal when performing SE, which has a significant potential to improve the quality and intelligibility of the enhanced speech.

Due to the importance of DRL for DNN [30, 31], DRL-based SE algorithms have been investigated in recent research works [37, 40–44]. These methods mainly use a variational autoencoder (VAE) [45] to learn speech representations and improve the generalization ability of the algorithms. VAE is a DRL model that can make efficient approximate posterior inferences and learn the probability distribution of complex data. Therefore, VAE can help DNN extract useful information from the signals [45]. Currently, VAE has been widely applied in various tasks related to representation learning [46, 47]. Although such VAE-based SE algorithms effectively improve DNN's generalization ability, they only consider the speech representation of the observed signal and do not attempt to disentangle the speech representation with latent noise representations. Instead, they use NMF to model the noise signal [37, 40–44]. This directly results in inaccurate obtained speech representations and possibly unsatisfactory SE performance [41].

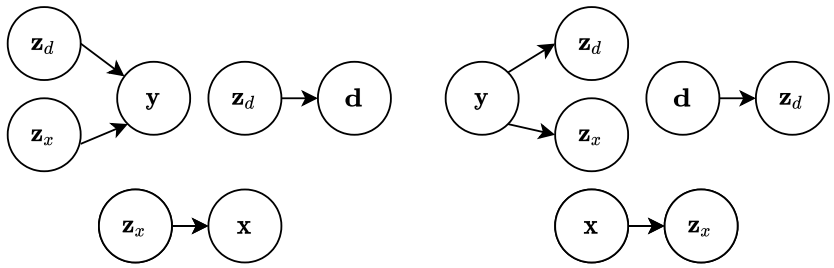
To obtain a more accurate speech representation, a novel VAE-based SE method [48], named Bayesian permutation training variational autoencoder (PVAE), was pro-

posed in our preliminary research. This method leverages a conditional posterior assumption to derive a novel evidence lower bound (ELBO) that enables the VAE to disentangle different signal representations in a very effective way. In addition, the derived ELBO also leads to a novel VAE model for SE. Compared to previous VAE-based SE models [37, 40–44], this model first extracts a more accurate speech representation from the observed signal, because different latent representations are disentangled [48] and these representations are expressed in a low-dimension space; the extracted representations are then used as the input of different decoders for SE. PVAE [48] can be directly adopted by many current SE DNN structures [17] and also directly used to optimize DNN-based SE algorithms [17]. Conducted experiments [48] indicate that this DRL strategy can help the traditional DNN-based SE method [49] achieve a better SE performance.

To further help PVAE to achieve better SE performance, we propose to leverage β-VAE [50, 51] to improve PVAE's representation learning ability. More specifically, the proposed β-PVAE [52] algorithm improves PVAE's capacity to disentangle different latent variables from the observed signal, which means that β-PVAE can obtain a better signal representation for SE. Moreover, β-PVAE optimizes the PVAE's network structure by setting β to infinity, which ensures that β-PVAE can not only improve PVAE's SE performance but also reduce the number of PVAE training parameters.

Both the speech and noise signal representations obtained by PVAE and β-PVAE are based on speech and noise posterior estimations [48]. An experimental analysis in [52] indicated that an accurate posterior estimation is crucial for β-PVAE because β-PVAE's decoders rely heavily on the accurate representation as input to reconstruct signals. Therefore, an accurate posterior estimation can lead to high SE performance [52]. On the other hand, an inaccurate posterior estimate can undermine the decoder's SE performance. However, obtaining pretty accurate posterior estimations is difficult since posteriors are intractable. In addition, another possible reason for the potential inaccurate posterior estimation is that the posterior estimations in [48] rely on a conditional assumption [52]. Although this conditional assumption results in a good signal model and ensures that various signal representations can be disentangled, it is difficult to validate that this assumption is consistently correct in a complex noisy environment. As a result, it potentially leads to β-PVAE to have inaccurate speech signal estimate and its SE performance is limited.

To mitigate the effect of inaccurate posterior estimations for the signal estimation and improve the SE performance of our preliminary work [52], we extend our DRL-based SE framework [48, 52] and propose in this paper a two-stage DRL-based SE method consisting of a representation learning stage [30] and an adversarial training stage [53]. In the first representation learning stage, we leverage the β-PVAE [52] to disentangle different signal representations from the observed signal to obtain speech and noise representations from the observed signal. To further obtain a better SE performance, in the second adversarial training stage, we propose to leverage generative adversarial networks (GANs) to improve the decoders' robustness for any possible inaccurate posterior estimation. Because we cannot ensure that the obtained posterior estimations are always accurate using β-PVAE, we instead attempt to make the decoders more robust. GAN is a probability generative model which can perform exact sampling from the desired distribution given random variables as input, using different f-divergence as training metrics [53, 54]. Unlike the β-PVAE's decoder, this model can generate a desired sample without having precise knowledge of the distribution of the input sample. Moreover, adversarial training can usually improve VAE decoder's signal



(a) Generative model (b) Recognition model

Fig. F.1: Graphic illustration of the proposed signal model.

reconstruction ability and help the VAE obtain higher quality signals [46, 47, 55–57]. Therefore, we introduce adversarial training to improve β-PVAE decoders' generative ability.

Recently, a combination of VAE and GAN (VAE–GAN) [55–57] has been widely applied in various speech synthesis tasks [46, 47]. VAE–GAN can achieve better performance than independent GAN or VAE-based methods [55], which usually use VAE to obtain a reliable signal representation and then involve the GAN to generate a high-quality signal. However, unlike our VAE–GAN-based SE algorithm, most of the current VAE–GAN-based methods [46, 47, 55] do not disentangle various representations in the VAE training stage. To the best of our knowledge, this is the first attempt to investigate VAE–GAN's application in the SE field. Furthermore, compared to the current state-of-the-art (SOTA) GAN-based SE methods [58, 59], VAE–GAN can obtain a disentangled signal representation as the GAN's input. A discriminative input can place less demand on the learning machine in order to perform a task successfully [17], which means that our VAE-GAN can help current GAN-based SE algorithms generate a higher quality speech signal.

This paper is organized as follows. First, in Section II, we will briefly review related VAE and GAN works. Then, we will proceed to illustrate the proposed two-stage VAE–GAN-based SE method in Section III and the experimental preparation, comparison, and analysis in Section IV. Finally, we draw conclusions in Section V.

2 Fundamentals

2.1 Signal Model

In this work, we assume that the noisy speech is additive, so the signal model can be written as follows:

$$y(t) = x(t) + d(t), \tag{F.1}$$

where $y(t)$, $x(t)$, and $d(t)$ represent the observed, speech, and noise signal, respectively, and t is the time index. Using the STFT, the observed signal $y_{f,n} \in \mathbb{C}$, speech signal $x_{f,n} \in \mathbb{C}$, and noise $d_{f,n} \in \mathbb{C}$ can be represented as

$$y_{f,n} = x_{f,n} + d_{f,n}, \tag{F.2}$$

where time frame index $n \in [1, N]$, and the frequency bin $f \in [1, F]$. N and F are the number of time frames and frequency bins, respectively.

We use the log-power spectrum (LPS) as the DNN's input feature since LPS is thought to offer perceptually relevant parameters for DNN-based SE algorithms [15, 17, 60, 61]. At present, LPS, as the input feature, has been widely applied in the DNN-based SE algorithms [15, 17, 60, 61]. The LPS vector [15] at each frame is written as $\mathbf{y}$, $\mathbf{x}$, and $\mathbf{d}$, respectively (we omit the frequency and time frame index for simplicity). Moreover, in the following derivations of our algorithm, the additive assumption in models (F.1) and (F.2) are not used. The purpose of (F.1) and (F.2) is used to generate noisy signal. Furthermore, (F.1) is a simple noisy signal model, so it is convenient to verify the correctness of our methods. Our framework has potential to analyze more challenging noisy signal models.

We assume that $\mathbf{y}$ can be generated from a random process involving the speech latent variables $\mathbf{z}_x \in \mathbb{R}^L$ and the noise latent variables $\mathbf{z}_d \in \mathbb{R}^L$ (L is the dimension of latent variables). The latent variables $\mathbf{z}_x$ and $\mathbf{z}_d$ are independent representations of the speech and noise signal, respectively. The combination of $\mathbf{z}_x$ and $\mathbf{z}_d$ is the representation of the observed signal [30, 45]. The $\mathbf{x}$ and $\mathbf{d}$ can be independently generated by $\mathbf{z}_x$ and $\mathbf{z}_d$, respectively: the generative process is shown in Fig. F.1(a). To obtain the latent variables $\mathbf{z}_x$ and $\mathbf{z}_d$, we assume that $\mathbf{z}_x$ and $\mathbf{z}_d$ can be estimated from the speech and noise posterior distributions $p(\mathbf{z}_x|\mathbf{x})$ and $p(\mathbf{z}_d|\mathbf{d})$, respectively, or from the noisy speech posterior distributions $p(\mathbf{z}_x|\mathbf{y})$ and $p(\mathbf{z}_d|\mathbf{y})$ [48], based on the VAE's property [45]. Fig. F.1(b) shows the recognition process [45]. To perform SE, it is necessary to disentangle the different latent variables from the observed signal. To simplify the disentanglement, we assume that $p(\mathbf{z}_x, \mathbf{z}_d|\mathbf{y}) = p(\mathbf{z}_x|\mathbf{y})p(\mathbf{z}_d|\mathbf{y})$ in [48].

2.2 VAE and β-VAE

The original VAE is a probabilistic generative model [45] which defines a probabilistic generative process between the observed signal and its latent variables and provides a principled method to jointly learn latent variables and generative and recognition models. Generative and recognition models are jointly trained by maximizing the ELBO or minimizing the Kullback–Leibler (KL) divergence between their real joint distribution and the corresponding estimation [45] using the stochastic gradient descent (SGD) or Adagrad [62] algorithm. Maximized, the ELBO can be written as follows:

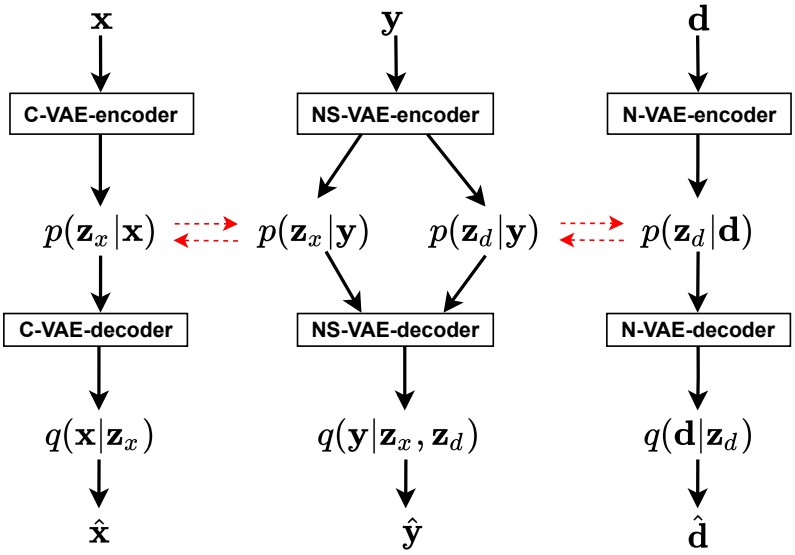
$$\begin{aligned} &\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}[\log q(\mathbf{y})] \geq -\mathcal{L}_n, \\ &\mathcal{L}_n = \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[D_{KL}\left(p(\mathbf{z}_y|\mathbf{y})||q(\mathbf{z}_y)\right)\right] \\ &\qquad - \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\mathbb{E}_{\mathbf{z}_y\sim p(\mathbf{z}_y|\mathbf{y})}\left[\log q(\mathbf{y}|\mathbf{z}_y)\right]\right], \end{aligned} \tag{F.3}$$

where $D_{KL}(||)$ denotes the KL divergence; $\mathbf{z}_y \in \mathbb{R}^L$ is the noisy latent variable. Maximizing this lower bound is equivalent to minimizing $\mathcal{L}_n$.

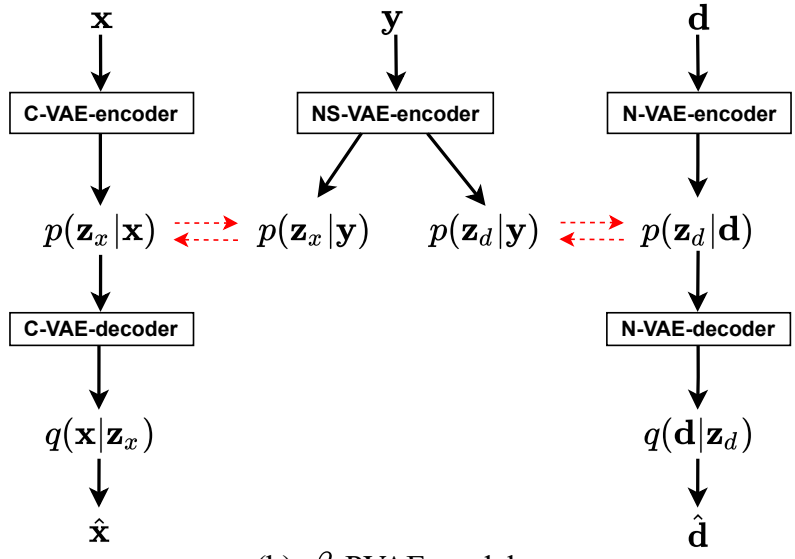
Furthermore, β-VAE [50, 51] is a modification of the original VAE framework, which introduces an adjustable hyperparameter β in the KL divergence term:

$$\begin{aligned} \mathcal{L}_\beta = &\beta\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[D_{KL}\left(p(\mathbf{z}_y|\mathbf{y})||q(\mathbf{z}_y)\right)\right] \\ &- \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\mathbb{E}_{\mathbf{z}_y\sim p(\mathbf{z}_y|\mathbf{y})}\left[\log q(\mathbf{y}|\mathbf{z}_y)\right]\right]. \end{aligned} \tag{F.4}$$

β-VAE aims to help the original VAE [45] to obtain a better signal representation. In general, $\beta > 1$ results in more disentangled latent representations [50]. A higher value of β can encourage learning a more disentangled representation.



(a) PVAE model



(b) β-PVAE model

Fig. F.2: Model illustration of PVAE and β-PVAE.

2.3 PVAE

Our preliminary work proposed a PVAE-based SE algorithm [48] and indicated that PVAE can help the current DNN-based SE method [49] obtain better signal representations (because different latent representations are disentangled [48] and these representations are expressed in a low-dimension space [30]) and achieve better SE performance. PVAE is a semi-supervised DRL-based SE method which introduces multiple latent variables in VAE and disentangles them in a semi-supervised way for SE application. Fig. F.2(a) shows the PVAE framework. We can see that PVAE includes three VAE structures: clean speech VAE (C-VAE), noise VAE (N-VAE), and noisy VAE (NS-VAE). C-VAE and N-VAE are trained without supervision to obtain speech and noise latent representations and their posterior estimates $p(\mathbf{z}_x|\mathbf{x})$, and $p(\mathbf{z}_d|\mathbf{d})$, respectively. This is achieved by minimizing the following VAE loss function [45]:

$$\begin{aligned}\mathcal{L}_c(\theta_x,\varphi_x;\mathbf{x}) = \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\{&D_{KL}\left(p(\mathbf{z}_x|\mathbf{x})||q(\mathbf{z}_x)\right)\\ &-\mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{x})}[\log q(\mathbf{x}|\mathbf{z}_x)]\},\end{aligned} \tag{F.5}$$

$$\begin{aligned}\mathcal{L}_d(\theta_d,\varphi_d;\mathbf{d}) = \mathbb{E}_{\mathbf{d}\sim p(\mathbf{d})}\{&D_{KL}\left(p(\mathbf{z}_d|\mathbf{d})||q(\mathbf{z}_d)\right)\\ &-\mathbb{E}_{\mathbf{z}_d\sim p(\mathbf{z}_d|\mathbf{d})}[\log q(\mathbf{d}|\mathbf{z}_d)]\},\end{aligned} \tag{F.6}$$

where $\theta_x, \varphi_x, \theta_d, \varphi_d$ are the DNN parameters for the related probability estimation [48]: θ_x and φ_x are the C-VAE's encoder and decoder parameters, respectively; θ_d and φ_d are the N-VAE's encoder and decoder parameters, respectively. NS-VAE is trained under the supervision of C-VAE and N-VAE's encoders and is meant to disentangle speech and noise latent variables from the observed signal for SE application. Based on the derivation in [48], the NS-VAE's training loss function is expressed as follows:

$$\begin{aligned}&\mathcal{L}_p(\theta_y,\varphi_y;\mathbf{y})\\ &\quad=\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y}),\mathbf{x}\sim\mathbf{p}(\mathbf{x})}\{D_{KL}\left(p(\mathbf{z}_x|\mathbf{y})||p(\mathbf{z}_x|\mathbf{x})\right)\\ &\qquad+\mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{y})}[\log\frac{p(\mathbf{z}_x|\mathbf{x})}{q(\mathbf{z}_x)}]\}\\ &\qquad+\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y}),\mathbf{d}\sim p(\mathbf{d})}\{D_{KL}\left(p(\mathbf{z}_d|\mathbf{y})||p(\mathbf{z}_d|\mathbf{d})\right)\\ &\qquad+\mathbb{E}_{\mathbf{z}_d\sim p(\mathbf{z}_d|\mathbf{y})}[\log\frac{p(\mathbf{z}_d|\mathbf{d})}{q(\mathbf{z}_d)}]\}\\ &\qquad-\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\mathbb{E}_{\mathbf{z}_d,\mathbf{z}_x\sim p(\mathbf{z}_d,\mathbf{z}_x|\mathbf{y})}\left[\log q(\mathbf{y}|\mathbf{z}_x,\mathbf{z}_d)\right]\right],\end{aligned} \tag{F.7}$$

where θ_y and φ_y are the NS-VAE's encoder and decoder parameters, respectively.

There are two stages for the PVAE-based SE algorithm. In the training stage, C-VAE and N-VAE are separately pre-trained by self-supervision using (F.5) and (F.6). After that, the C-VAE and N-VAE are frozen, and NS-VAE is trained by (F.7). In the enhancement stage, the NS-VAE encoder's two outputs can be used as the input of C-VAE and N-VAE to obtain the prior distributions $q(\mathbf{x}|\mathbf{z}_x)$ and $q(\mathbf{d}|\mathbf{z}_d)$ for SE.

2.4 β-PVAE

To further improve PVAE's SE performance, we propose to leverage β-VAE to improve PVAE's disentangling ability [52] in our another preliminary work. Furthermore, the proposed β-PVAE makes the best use of the β-VAE's trade-off property to simplify the PVAE's network structure and training parameters by setting β to infinity and discarding

the noisy speech restoration term [52], which means that β-PVAE can achieve a better disentangling and enhancement performance than PVAE with a simpler structure. Based on our derivations [48, 52], the β-PVAE's optimization target for $\beta \to +\infty$ is [52]

$$\begin{aligned}\mathcal{L}_{\beta p}(\theta_y; \mathbf{y}) = &\mathbb{E}_{\mathbf{y}\sim p(\mathbf{y}),\mathbf{x}\sim\mathbf{p}(\mathbf{x})}\{D_{KL}\left(p(\mathbf{z}_x|\mathbf{y})||p(\mathbf{z}_x|\mathbf{x})\right) \\ &+ \mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{y})}[\log\frac{p(\mathbf{z}_x|\mathbf{x})}{q(\mathbf{z}_x)}]\} \\ &+ \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y}),\mathbf{d}\sim p(\mathbf{d})}\{D_{KL}\left(p(\mathbf{z}_d|\mathbf{y})||p(\mathbf{z}_d|\mathbf{d})\right) \\ &+ \mathbb{E}_{\mathbf{z}_d\sim p(\mathbf{z}_d|\mathbf{y})}[\log\frac{p(\mathbf{z}_d|\mathbf{d})}{q(\mathbf{z}_d)}]\}.\end{aligned} \tag{F.8}$$

Comparing (F.8) with (F.7), we can find that there is no reconstruction term in β-PVAE. Thus, β-PVAE's framework can be simplified by removing the NS-decoder part (Fig. F.2(b)). The β-PVAE's training process is similar to PVAE; the only difference is that the β-PVAE's training optimization target is (F.8) rather than (F.7).

2.5 Generative Adversarial Network (GAN)

A GAN [53] consists of two networks: a generator network and a discriminator network. The generator network $G(\mathbf{z})$ maps latent $\mathbf{z}$ ($\mathbf{z} \sim q(\mathbf{z})$) to the data space (e.g., observed signal data). Typically, there are no rigid restrictions for the distribution $q(\mathbf{z})$ [54]. The discriminator network $D(\cdot)$ is used to determine whether $\mathbf{y}$ is an actual training sample ($D(\mathbf{y})$) or it is generated by the model through $\mathbf{y} = G(\mathbf{z})$ ($D(G(\mathbf{z}))$). GANs can be optimized by different f-divergences [54]. In Jensen–Shannon (JS) divergence, GANs is optimized by the minimax of the loss function [53]:

$$\begin{aligned}&\min_G \max_D \mathcal{L}_{gan}(G, D) = \\ &\mathbb{E}_{\mathbf{y}\sim q_{data}(\mathbf{y})}[\log(D(\mathbf{y}))] + \mathbb{E}_{\mathbf{z}\sim q(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))].\end{aligned} \tag{F.9}$$

GANs have been applied in SE [58, 59, 63, 64], but the researched methods do not consider how a good speech representation can be obtained as the input of the GAN for SE. Instead, they use the observed signal as the GAN's input to generate the speech signal [58, 59]. Although there are no set restrictions for the GAN's input, an accurate and discriminative signal representation [17] can usually lead to better generative performance for the GAN [46, 47].

3 Speech Enhancement with VAE and GAN

To obtain a higher quality enhanced speech, in this paper, we extend DRL-based SE framework [52]. We propose a VAE-GAN SE algorithm which introduces adversarial training to increase the decoders' robustness and signal restoration ability. In this algorithm, we split the training process into two stages: the representation learning and the adversarial training. In the first stage, we leverage β-PVAE to disentangle speech and noise latent representations from the observed signal. The purpose is to obtain a good signal representation, making the clean speech generation easier. In the second, adversarial training, stage, we freeze the β-PVAE's encoders and leverage adversarial training to optimize β-PVAE's decoders. GANs can generate desired samples without

Algorithm 1 Representation Learning.

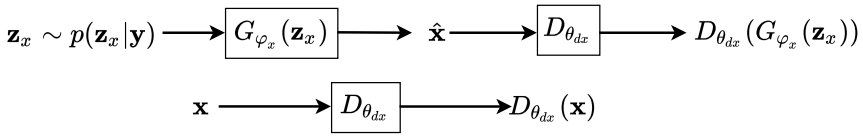
Pre-train 1: Using the speech dataset and loss function (F.5) to train a general speech VAE (C-VAE) [45].
Pre-train 2: Using the noise dataset and loss function (F.6) to train a general noise VAE (N-VAE) [45].
Repeat:
1. Choose random M samples from the speech, noise, and observed signal dataset and build a corresponding mini-batch;
2. Use the chosen speech, noise, and observed signal samples as the encoders' input of C-VAE, N-VAE, and NS-VAE, respectively;
3. Estimate the related posterior probability $p(\mathbf{z}_x|\mathbf{y})$, $p(\mathbf{z}_d|\mathbf{y}), p(\mathbf{z}_x|\mathbf{x})$, and $p(\mathbf{z}_d|\mathbf{d})$ using the equations:
(1) $\mu_{\theta_{yx}}(\mathbf{y}), \sigma^2_{\theta_{yx}}(\mathbf{y}), \mu_{\theta_{yd}}(\mathbf{y}), \sigma^2_{\theta_{yd}}(\mathbf{y}) = G_{\theta_y}(\mathbf{y})$,
(2) $\mu_{\theta_x}(\mathbf{x}), \sigma^2_{\theta_x}(\mathbf{x}) = G_{\theta_x}(\mathbf{x})$,
(3) $\mu_{\theta_d}(\mathbf{d}), \sigma^2_{\theta_d}(\mathbf{d}) = G_{\theta_d}(\mathbf{d})$;
4. Calculate loss function (E.8);
5. Freeze C-VAE and N-VAE and apply the SGD algorithm to update the NS-VAE's parameters θ_y [45];
until the convergence of the loss function.
Return: The trained NS-VAE (G_{θ_x}).

accurate knowledge of the input sample distribution [53, 54] (it only needs samples) and it can also improve VAE decoder's generative performance [55–57], so GANs can mitigate the effect of potentially inaccurate posterior estimation for β-PVAE's decoders and improve decoder's generative ability. As a result, β-PVAE can achieve a satisfactory SE performance even if the posterior estimation is inaccurate. In this section, we will first show the details of representation learning. Then, we will explain the adversarial training processes. After that, we will indicate how to apply the proposed VAE-GAN to conduct online SE.

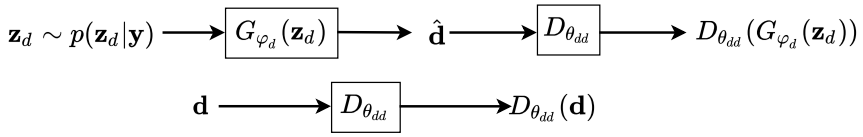
3.1 Stage 1: Representation Learning

In the first stage, we aim to disentangles speech and noise latent variables from the observed signal. This process is accomplished by the proposed β-PVAE [52].

In β-PVAE, C-VAE and N-VAE are optimized by (F.5) and (F.6), respectively, and NS-VAE is optimized by (F.8). To calculate (F.5), (F.6), and (F.8), it is necessary to determine the related posterior and prior distributions and predefine $q(\mathbf{z}_x)$ and $q(\mathbf{z}_d)$. For the simplicity of the calculation, we assume that all posterior and prior distributions are multivariate normal distributions with diagonal covariance [45], which is similar to the previous VAE-based SE methods [40–44]. For NS-VAE, we have



(a) Adversarial training for speech



(b) Adversarial training for noise

Fig. F.3: Graphic illustration of adversarial training.

$$\begin{aligned} p(\mathbf{z}_x|\mathbf{y}) &= \mathcal{N}\left(\mathbf{z}_x; \mu_{\theta_{yx}}(\mathbf{y}), \sigma^2_{\theta_{yx}}(\mathbf{y})\mathbf{I}\right) \\ p(\mathbf{z}_d|\mathbf{y}) &= \mathcal{N}\left(\mathbf{z}_d; \mu_{\theta_{yd}}(\mathbf{y}), \sigma^2_{\theta_{yd}}(\mathbf{y})\mathbf{I}\right), \end{aligned} \tag{F.10}$$

where $\mathbf{I}$ is the identity matrix; $\mu_{\theta_{yx}}(\mathbf{y}), \sigma^2_{\theta_{yx}}(\mathbf{y}), \mu_{\theta_{yd}}(\mathbf{y})$, and $\sigma^2_{\theta_{yd}}(\mathbf{y})$ can be estimated by NS-VAE's encoder $G_{\theta_y}(\mathbf{y})$ with parameter $\theta_y = \{\theta_{yx}, \theta_{yd}\}$. μ and σ^2 represent the mean and variance in the related Gaussian distributions, respectively. Moreover, the prior and posterior estimation for C-VAE is

$$\begin{aligned} p(\mathbf{z}_x|\mathbf{x}) &= \mathcal{N}\left(\mathbf{z}_x; \mu_{\theta_x}(\mathbf{x}), \sigma^2_{\theta_x}(\mathbf{x})\mathbf{I}\right) \\ q(\mathbf{x}|\mathbf{z}_x) &= \mathcal{N}\left(\mathbf{x}; \mu_{\varphi_x}(\mathbf{z}_x), \sigma^2_{\varphi_x}(\mathbf{z}_x)\mathbf{I}\right), \end{aligned} \tag{F.11}$$

where $\mu_{\theta_x}(\mathbf{x})$ and $\sigma^2_{\theta_x}(\mathbf{x})$ are obtained by C-VAE's encoder $G_{\theta_x}(\mathbf{x})$ with parameter θ_x, and $\mu_{\varphi_x}(\mathbf{z}_x)$ and $\sigma^2_{\varphi_x}(\mathbf{z}_x)$ can be estimated by C-VAE's decoder $G_{\varphi_x}(\mathbf{z}_x)$ with parameter φ_x. Similarly, for N-VAE, we have

$$\begin{aligned} p(\mathbf{z}_d|\mathbf{d}) &= \mathcal{N}\left(\mathbf{z}_d; \mu_{\theta_d}(\mathbf{d}), \sigma^2_{\theta_d}(\mathbf{d})\mathbf{I}\right) \\ q(\mathbf{d}|\mathbf{z}_d) &= \mathcal{N}\left(\mathbf{d}; \mu_{\varphi_d}(\mathbf{z}_d), \sigma^2_{\varphi_d}(\mathbf{z}_d)\mathbf{I}\right), \end{aligned} \tag{F.12}$$

where $\mu_{\theta_d}(\mathbf{d})$ and $\sigma^2_{\theta_d}(\mathbf{d})$ are obtained by C-VAE's encoder $G_{\theta_d}(\mathbf{d})$ with parameter θ_x, and $\mu_{\varphi_d}(\mathbf{z}_d)$ and $\sigma^2_{\varphi_d}(\mathbf{z}_d)$ can be estimated by C-VAE's decoder $G_{\varphi_d}(\mathbf{z}_d)$ with parameter φ_d. Furthermore, $q(\mathbf{z}_d)$ and $q(\mathbf{z}_x)$ are pre-defined as a centered isotropic multivariate Gaussian, which can be represented as

$$\begin{aligned} q(\mathbf{z}_x) &= \mathcal{N}(\mathbf{z}_x; \mathbf{0}, \mathbf{I}) \\ q(\mathbf{z}_d) &= \mathcal{N}(\mathbf{z}_d; \mathbf{0}, \mathbf{I}). \end{aligned} \tag{F.13}$$

The entire representation learning process is summarized in Algorithm 1.

3.2 Stage 2: Adversarial Training

The second training stage aims to improve the decoders' robustness and signal restoration ability in β-PVAE for the better SE performance. It is difficult to ensure that dis-

Algorithm 2 Adversarial Training.

Repeat:
1. Choose random M samples from the speech, noise, and observed signal dataset, respectively, and build a corresponding mini-batch;
2. Use the observed signal samples as the input of NS-VAE;
3. Estimate the related posterior probability $p(\mathbf{z}_x|\mathbf{y})$ and $p(\mathbf{z}_d|\mathbf{y})$ using the following equation: $\mu_{\theta_{yx}}(\mathbf{y}), \sigma^2_{\theta_{yx}}(\mathbf{y}), \mu_{\theta_{yd}}(\mathbf{y}), \sigma^2_{\theta_{yd}}(\mathbf{y}) = G_{\theta_y}(\mathbf{y})$;
4. Apply the reparameterization trick to obtain sample $\mathbf{z}_x \sim p(\mathbf{z}_x|\mathbf{y})$ and $\mathbf{z}_d \sim p(\mathbf{z}_d|\mathbf{y})$ [53];
5. Use $\mathbf{z}_x$ and $\mathbf{z}_d$ as the C-VAE decoder's (G_{φ_x}) input and N-VAE decoder's (G_{φ_d}) input, respectively;
6. Calculate the loss function (F.14), (F.15), (F.16), (F.17);
5. Freeze all encoders and apply SGD to update parameters φ_x, φ_d, θ_{dx}, and θ_{dd} for $G_{\varphi_x}, G_{\varphi_d}$, $D_{\theta_{dx}}$, and $D_{\theta_{dd}}$ respectively;

until the convergence of the loss function

Return: The trained decoders and discriminators: $G_{\varphi_x}, G_{\varphi_d}$, $D_{\theta_{dx}}$, and $D_{\theta_{dd}}$.

entangled speech and noise latent representations are consistently accurate in complex noisy environments. Considering that decoders' SE performance relies on the accurate representations, we propose to leverage adversarial training to mitigate this contradiction. In general, a GAN can generate the data, given the input is a random noise variable [53, 58]. Moreover, adversarial training can usually improve decoder's signal restoration ability [55–57]. As a result, we can use GANs to reduce decoders' dependence on accurate representation, which means that even with inaccurate representation estimations, decoders can achieve a satisfactory SE performance.

To adopt adversarial training in the β-PVAE system, we add two discriminators, $D_{\theta_{dx}}(\cdot)$ and $D_{\theta_{dd}}(\cdot)$, with parameters θ_{dx} and θ_{dd}, respectively. $D_{\theta_{dx}}(\cdot)$ is used to distinguish between the speech generated by the C-VAE decoder $G_{\varphi_x}(\mathbf{z}_x)$ and the ground truth speech $\mathbf{x}$. Similarly, we apply $D_{\theta_{dd}}(\cdot)$ to distinguish between the noise generated by the N-VAE decoder $G_{\varphi_d}(\mathbf{z}_d)$ and the ground truth noise $\mathbf{d}$. Fig. F.3 shows the related adversarial training process. In this work, we use the least squares GAN [65] loss function for adversarial training, which has been widely used in various GAN applications [46, 47] as it can achieve a more stable training process and avoid the problem of vanishing gradients, compared to the original GAN [53] loss function. Moreover, although GAN can generate high-quality signals, GAN may diverge too much from the target signals [55–57]. So, to ensure that the generated signals do not diverge too much from the ground truth signals, we reserve the original reconstruction term in the representation learning stage when conducting adversarial training. This is a GAN training trick for our proposed VAE-GAN, which is similar to the feature matching loss in previous applications of GANs [46, 47, 58, 59, 66, 67]. Therefore, the adversarial loss

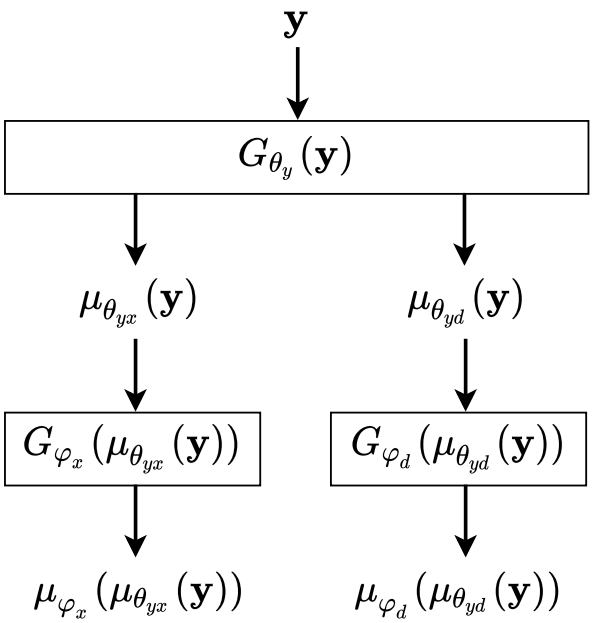


Fig. F.4: VAE-GAN for online SE.

function for C-VAE-decoder can be expressed as follows:

$$\begin{aligned}\mathcal{L}_{gan_c}(G_{\varphi_x}) &= \mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{y})}[(D_{\theta_{dx}}(G_{\varphi_x}(\mathbf{z}_x))-1)^2] \\ &\quad - \mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{y})}[\log q(\mathbf{x}|\mathbf{z}_x)],\end{aligned} \tag{F.14}$$

$$\begin{aligned}\mathcal{L}_{gan_c}(D_{\theta_{dx}}) &= \mathbb{E}_{\mathbf{z}_x\sim p(\mathbf{z}_x|\mathbf{y})}[(D_{\theta_{dx}}(G_{\varphi_x}(\mathbf{z}_x)))^2] \\ &\quad + \mathbb{E}_{\mathbf{x}\sim q_{data}(\mathbf{x})}[(D_{\theta_{dx}}(\mathbf{x})-1)^2].\end{aligned} \tag{F.15}$$

Similarly, the adversarial loss function for noise can be represented as

$$\begin{aligned}\mathcal{L}_{gan_d}(G_{\varphi_d}) &= \mathbb{E}_{\mathbf{z}_d\sim p(\mathbf{z}_d|\mathbf{y})}[(D_{\theta_{dd}}(G_{\varphi_d}(\mathbf{z}_d))-1)^2] \\ &\quad - \mathbb{E}_{\mathbf{z}_d\sim p(\mathbf{z}_d|\mathbf{y})}[\log q(\mathbf{d}|\mathbf{z}_d)],\end{aligned} \tag{F.16}$$

$$\begin{aligned}\mathcal{L}_{gan_d}(D_{\theta_{dd}}) &= \mathbb{E}_{\mathbf{z}_d\sim p(\mathbf{z}_d|\mathbf{y})}[(D_{\theta_{dd}}(G_{\varphi_d}(\mathbf{z}_d)))^2] \\ &\quad + \mathbb{E}_{\mathbf{d}\sim q_{data}(\mathbf{d})}[(D_{\theta_{dd}}(\mathbf{d})-1)^2].\end{aligned} \tag{F.17}$$

The complete adversarial training process is summarized in Algorithm 2.

3.3 VAE-GAN for Online Speech Enhancement

The online SE stage requires only the NS-VAE encoder G_θ, C-VAE decoder G_{φ_x}, and N-VAE decoder G_{φ_d} to conduct SE, which is similar to PVAE [48] and β-PVAE [52]. To obtain an enhanced signal, first, the observed signal is directly used as the input of G_θ. Then, the posterior means $\mu_{\theta_{yx}}(\mathbf{y})$ and $\mu_{\theta_{yd}}(\mathbf{y})$ are obtained. After that, $\mu_{\theta_{yx}}(\mathbf{y})$ and $\mu_{\theta_{yd}}(\mathbf{y})$ are used separately as the input for G_{φ_x} and G_{φ_d} to estimate the speech mean

Algorithm 3 Online SE.

1: Apply the observed signal $\mathbf{y}$ as the NS-VAE's encoder (G_{θ_x}) input;
2. Estimate the posterior probability $p(\mathbf{z}_x|\mathbf{y})$ and $p(\mathbf{z}_d|\mathbf{y})$ by: $\mu_{\theta_{yx}}(\mathbf{y}), \sigma^2_{\theta_{yx}}(\mathbf{y}), \mu_{\theta_{yd}}(\mathbf{y}), \sigma^2_{\theta_{yd}}(\mathbf{y}) = G_{\theta_y}(\mathbf{y})$;
3. Use $\mu_{\theta_{yx}}(\mathbf{y})$ and $\mu_{\theta_{yd}}(\mathbf{y})$ as the inputs of C-VAE decoder G_{φ_x} and N-VAE decoder G_{φ_d}, respectively;
4. Apply decoders to estimate the speech and noise signal:
(1) $\mu_{\varphi_x}(\mu_{\theta_{yx}}(\mathbf{y}))$, $\sigma_{\varphi_x}(\mu_{\theta_{yx}}(\mathbf{y})) = G_{\varphi_x}(\mu_{\theta_{yx}}(\mathbf{y}))$
(2) $\mu_{\varphi_d}(\mu_{\theta_{yd}}(\mathbf{y}))$, $\sigma_{\varphi_d}(\mu_{\theta_{yd}}(\mathbf{y})) = G_{\varphi_d}(\mu_{\theta_{yd}}(\mathbf{y}))$;
5. Use $\mu_{\varphi_x}(\mu_{\theta_{yx}}(\mathbf{y}))$ and $\mu_{\varphi_d}(\mu_{\theta_{yd}}(\mathbf{y}))$ as the estimated speech and noise signal;
6. Apply waveform reconstruction [15] or mask the estimation [16] to obtain the enhanced speech signal $\hat{\mathbf{x}}$.
Return: The enhanced speech $\hat{\mathbf{x}}$.

$\mu_{\varphi_x}(\mu_{\theta_{yx}}(\mathbf{y}))$ and noise mean $\mu_{\varphi_d}(\mu_{\theta_{yd}}(\mathbf{y}))$, respectively. Finally, the estimated means are utilized as the enhanced speech and noise signal. The enhancement process is shown in Fig. F.4 and Algorithm 3. In the online SE stage, the means are used directly to estimate the signals, without the reparameterization trick [45], which is different from the training process [45]. Moreover, the proposed VAE-GAN can simultaneously estimate the speech and noise in the observed signal, so the final enhanced signal can be obtained by direct waveform reconstruction [15] or mask estimation [16].

4 Experimental Settings and Results

In this section, the proposed VAE-GAN-based SE algorithm is evaluated. To explore VAE-GAN's SE potential, we use related state-of-the-art (SOTA) algorithms as the reference methods to investigate VAE-GAN's SE performance.

4.1 Datasets

In this work, we created a training and test dataset using the speech and noise from the DNS challenge 2021 corpus [68]. To build a clean speech dataset, we selected English speakers and randomly split 70% of the speakers for training, 20% for validation, and 10% for evaluation. For the noise, all the noise from the DNS noise corpus was randomly divided into training, validation, and test noise in a proportion similar to that used for speech utterances. The noise dataset comprised approximately 150 audio classes and 60,000 clips (the noise details can be found in [68]). After that, the corresponding training, validation, and test corpus for speech and noise were randomly mixed using the DNS script [68] with random signal-to-noise ratio (SNR) levels (between –10dB and 15dB). The other parameters of the signal mixing were the default values in the DNS script [68]. Finally, we randomly chose 20 hours of mixed training utterances, 5 hours of mixed validation utterances, and 1 hour of mixed test utterances to build the experimental dataset. All signals were down-sampled to 16 kHz [68].

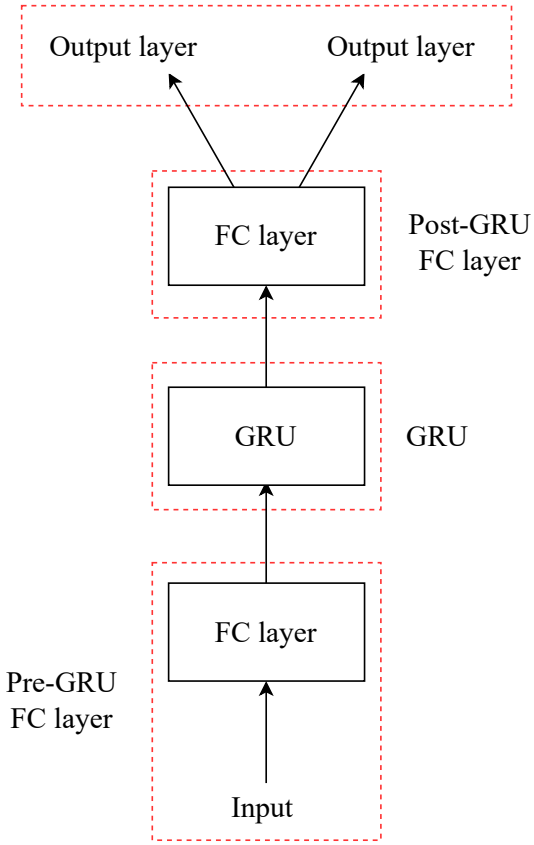


Fig. F.5: Network structure in VAE-GAN.

We also used the LibriSpeech [69], 100 environmental noises [70], and NOISEX-92 database [71] to evaluate the SE performance of various algorithms. The purpose was to see the SE performance of various algorithms in the unseen dataset. Random one-hour speech data from LibriSpeech database were chosen and then mixed randomly with all noises from 100 environmental noises [70] and the NOISEX-92 database [71]. The mixed SNRs were randomly chosen from the –10dB to 15dB. Finally, we obtained a one-hour noisy speech test data.

4.2 Experimental Setup

In our experiment, the signal frame length was 512 samples (32 ms) with a frame shift of 256 samples. The size of STFT was 256 points, so the 257-dimension LPS feature vectors were used to train the networks. Moreover, there were a total of 7 DNNs to be trained in VAE-GAN: C-VAE encoder G_{θ_x}, C-VAE decoder G_{φ_x}, N-VAE encoder G_{θ_d}, N-VAE decoder G_{φ_d}, NS-VAE encoder G_{θ_y}, speech discriminators $D_{\theta_{dx}}$, and noise discriminator $D_{\theta_{dd}}$. All the DNNs in our experiment were based on the gated recurrent unit

Table F.1: Network Details of VAE-GAN

Networks	Pre-GRU FC layer			GRU layer	
	Number	Nodes	AF	Number	Nodes
G_{θ_x} and G_{θ_d}	3	257-512-512	ReLU	1	512
G_{φ_x} and G_{φ_d}	1	128	ReLU	1	512
G_{θ_y}	3	257-512-512	ReLU	1	512
$D_{\theta_{dx}}$ and $D_{\theta_{dd}}$	2	257-512	ReLU	1	256

Networks	Post-GRU FC layer		
	Number	Nodes	AF
G_{θ_x} and G_{θ_d}	0	N/A	N/A
G_{φ_x} and G_{φ_d}	2	512-512	ReLU
G_{θ_y}	1	512	ReLU
$D_{\theta_{dx}}$ and $D_{\theta_{dd}}$	1	512	ReLU

Networks	Output layer		
	Number	Nodes	AF
G_{θ_x} and G_{θ_d}	2	128	Linear
G_{φ_x} and G_{φ_d}	2	257	Linear
G_{θ_y}	4	128	Linear
$D_{\theta_{dx}}$ and $D_{\theta_{dd}}$	1	1	Linear

(GRU) [72] due to their computational efficiency and superior performance in SE [73]. In this work, we stacked GRU layers after the fully-connected (FC) layers, followed by hidden FC layers and FC output layers (Figure F.5). This network design was similar to the baseline algorithm [74] in DNS challenge 2022 [75]. The detailed model design of each neural network is shown in Table F.1, where AF represents the activation function in each output layer; Pre-GRU FC layer and Post-GRU FC layer represent the FC layer before the GRU layer and after the GRU layer, respectively; and the Nodes is the node number in each layer (all output layers have the same number of nodes in the same network). Additionally, we set the dimension of latent variables $L = 128$, so for all encoders, the node number of the output layer is 128. All networks were trained by the Adam algorithm [76] with a 128 mini-batch size. The learning rate is 0.001.

Table F.2: SI-SDR Comparison in DNS dataset with a 95% confidence interval

SNR (dB)	Noise	GAN-SE [59]	NSNet2 [74]	β-PVAE-L [52]
-5	-4.40 ($\pm$ 0.80)	2.15 ($\pm$ 0.79)	5.07 ($\pm$ 0.74)	2.63 ($\pm$ 0.80)
0	2.63 ($\pm$ 1.04)	6.79 ($\pm$ 0.61)	9.77 ($\pm$ 0.81)	5.69 ($\pm$ 0.59)
5	7.63 ($\pm$ 1.08)	9.30 ($\pm$ 0.50)	13.09 ($\pm$ 0.82)	8.10 ($\pm$ 0.46)
10	13.58 ($\pm$ 1.05)	11.75 ($\pm$ 0.42)	16.76 ($\pm$ 0.72)	10.46 ($\pm$ 0.35)
Average	4.86 ($\pm$ 0.99)	7.49 ($\pm$ 0.58)	11.17 ($\pm$ 0.77)	6.72 ($\pm$ 0.55)

SNR (dB)	Noise	VAE-GAN-L	β-PVAE-M [52]	VAE-GAN-M
-5	-4.40 ($\pm$ 0.80)	4.52 ($\pm$ 0.72)	3.52 ($\pm$ 0.93)	**5.37** ($\pm$ **0.89**)
0	2.63 ($\pm$ 1.04)	8.48 ($\pm$ 0.52)	8.92 ($\pm$ 0.92)	**10.17** ($\pm$ **0.86**)
5	7.63 ($\pm$ 1.08)	10.96 ($\pm$ 0.39)	12.96 ($\pm$ 0.93)	**14.11** ($\pm$ **0.85**)
10	13.58 ($\pm$ 1.05)	13.07 ($\pm$ 0.30)	17.75 ($\pm$ 0.88)	**18.58** ($\pm$ **0.84**)
Average	4.86 ($\pm$ 0.99)	9.26 ($\pm$ 0.48)	10.78 ($\pm$ 0.91)	**12.06** ($\pm$ **0.86**)

4.3 Evaluation Metrics and Reference Methods

In this work, we will use the scale-invariant signal-to-distortion ratio (SI-SDR) in decibel (dB) [77], short-time objective intelligibility (STOI) [78], and perceptual evaluation of speech quality (PESQ) [79] as evaluation metrics to evaluate the proposed VAE-GAN's SE performance. SI-SDR is used to measure the signal distortion of the enhanced speech, so it can directly show the difference between the ground truth signal and the enhanced signal. PESQ and STOI are used to evaluate the quality and intelligibility for the enhanced speech, respectively.

To better evaluate the proposed VAE-GAN's SE performance, we choose three related SOTA SE algorithms as reference methods. The first reference method is GAN-SE [59], which is a SOTA GAN-based SE algorithm that can help us verify whether the better sig-

Table F.3: STOI (%) Comparison in DNS dataset with a 95% confidence interval

SNR (dB)	Noise	GAN-SE [59]	NSNet2 [74]	β-PVAE-L [52]
-5	73.80 ($\pm$ 1.70)	72.26 ($\pm$ 1.91)	78.15 ($\pm$ 1.61)	72.94 ($\pm$ 1.77)
0	82.46 ($\pm$ 1.40)	81.47 ($\pm$ 1.42)	87.03 ($\pm$ 1.12)	82.23 ($\pm$ 1.32)
5	88.01 ($\pm$ 1.11)	87.02 ($\pm$ 1.02)	91.63 ($\pm$ 0.81)	87.57 ($\pm$ 0.99)
10	93.54 ($\pm$ 0.72)	92.13 ($\pm$ 0.61)	95.59 ($\pm$ 0.47)	92.54 ($\pm$ 0.59)
Average	84.45 ($\pm$ 1.23)	83.22 ($\pm$ 1.24)	88.10 ($\pm$ 1.09)	83.82 ($\pm$ 1.00)

SNR (dB)	Noise	VAE-GAN-L	β-PVAE-M [52]	VAE-GAN-M
-5	73.80 ($\pm$ 1.70)	76.83 ($\pm$ 1.81)	77.27 ($\pm$ 1.71)	**79.29** ($\pm$ **1.80**)
0	82.46 ($\pm$ 1.40)	85.62 ($\pm$ 1.18)	86.02 ($\pm$ 1.25)	**87.06** ($\pm$ **1.19**)
5	88.01 ($\pm$ 1.11)	90.71 ($\pm$ 0.80)	91.08 ($\pm$ 0.91)	**92.01** ($\pm$ **0.82**)
10	93.54 ($\pm$ 0.72)	94.68 ($\pm$ 0.46)	95.58 ($\pm$ 0.51)	**96.02** ($\pm$ **0.47**)
Average	84.45 ($\pm$ 1.23)	86.96 ($\pm$ 1.06)	87.48 ($\pm$ 1.09)	**88.60** ($\pm$ **1.07**)

nal representations (disentangled and low-dimension representations) in the observed signal can improve GAN's SE performance. In addition, we can see the effectiveness of a disentangled signal representation for the GAN-based SE method. This also shows the DRL's importance for DNN-based SE algorithm. The second reference method is β-PVAE [52]. By comparing VAE-GAN's SE performance with β-PVAE, we can validate our hypothesis that adversarial training can improve β-PVAE's SE performance (the β-PVAE's encoder and decoders have the same structure as the VAE-GAN). Finally, we compare the proposed VAE-GAN with the DNS 2021 challenge baseline NSNet2 [74, 80] to see whether the VAE-GAN's SE performance is competitive with the current popular SOTA SE algorithms [74]. The main purpose of our experiment is not to outperform all SOTA performance, but to authentically verify the validity of the proposed VAE-GAN framework and its further potential.

Table F.4: PESQ Comparison in DNS dataset with a 95% confidence interval

SNR (dB)	Noise	GAN-SE [59]	NSNet2 [74]	β-PVAE-L [52]
-5	1.81 ($\pm$ 0.02)	2.00 ($\pm$ 0.03)	2.28 ($\pm$ 0.02)	2.08 ($\pm$ 0.03)
0	2.04 ($\pm$ 0.02)	2.33 ($\pm$ 0.02)	2.60 ($\pm$ 0.02)	2.46 ($\pm$ 0.03)
5	2.28 ($\pm$ 0.02)	2.62 ($\pm$ 0.02)	2.87 ($\pm$ 0.02)	2.77 ($\pm$ 0.02)
10	2.70 ($\pm$ 0.01)	3.00 ($\pm$ 0.01)	3.24 ($\pm$ 0.01)	3.14 ($\pm$ 0.01)
Average	2.21 ($\pm$ 0.02)	2.49 ($\pm$ 0.02)	2.75 ($\pm$ 0.02)	2.61 ($\pm$ 0.03)

SNR (dB)	Noise	VAE-GAN-L	β-PVAE-M [52]	VAE-GAN-M
-5	1.81 ($\pm$ 0.02)	**2.31** ($\pm$ **0.02**)	2.19 ($\pm$ 0.03)	2.30 ($\pm$ 0.02)
0	2.04 ($\pm$ 0.02)	**2.64** ($\pm$ **0.02**)	2.55 ($\pm$ 0.02)	2.62 ($\pm$ 0.01)
5	2.28 ($\pm$ 0.02)	**2.94** ($\pm$ **0.01**)	2.85 ($\pm$ 0.02)	2.93 ($\pm$ 0.01)
10	2.70 ($\pm$ 0.01)	**3.29** ($\pm$ **0.01**)	3.21 ($\pm$ 0.01)	**3.29** ($\pm$ **0.01**)
Average	2.21 ($\pm$ 0.02)	**2.80** ($\pm$ **0.02**)	2.70 ($\pm$ 0.02)	2.79 ($\pm$ 0.01)

For the VAE-GAN and β-PVAE, enhanced speech can be obtained by waveform reconstruction [15] or mask estimation [16]. The direct waveform reconstruction is based solely on the speech estimate, while the mask is based on both speech and noise estimate. So, we use β-PVAE-M and β-PVAE-L that represent that the enhanced speech is acquired by mask estimation and direct waveform reconstruction using β-PVAE [52], respectively; VAE-GAN-L and VAE-GAN-M denote that the enhanced speech is obtained by the proposed VAE-GAN using direct waveform reconstruction and mask estimation, respectively. We use the ideal ratio mask [16] that is widely applied in various SE tasks [16, 18] to conduct mask estimation.

Table F.5: Experimental result comparisons in LibriSpeech dataset with a 95% confidence interval

Evaluation Metrics	Noise	GAN-SE [59]	NSNet2 [74]	β-PVAE-L [52]
SI-SDR	1.81 ($\pm$ 0.23)	6.16 ($\pm$ 0.36)	9.20 ($\pm$ 0.70)	6.40 ($\pm$ 0.45)
STOI (%)	82.75 ($\pm$ 1.63)	80.86 ($\pm$ 1.69)	86.03 ($\pm$ 1.51)	81.56 ($\pm$ 1.53)
PESQ	2.31 ($\pm$ 0.03)	2.52 ($\pm$ 0.02)	2.69 ($\pm$ 0.01)	2.54 ($\pm$ 0.03)

Evaluation Metrics	Noise	VAE-GAN-L	β-PVAE-M [52]	VAE-GAN-M
SI-SDR	1.81 ($\pm$ 0.23)	8.24 ($\pm$ 0.50)	7.04 ($\pm$ 0.46)	**10.18** ($\pm$ **0.56**)
STOI (%)	82.75 ($\pm$ 1.63)	84.50 ($\pm$ 1.47)	85.32 ($\pm$ 1.53)	**86.05** ($\pm$ **1.50**)
PESQ	2.31 ($\pm$ 0.03)	2.71 ($\pm$ 0.02)	2.67 ($\pm$ 0.02)	**2.72** ($\pm$ **0.01**)

4.4 Experimental Results and Analysis

In this work, STOI, PESQ, and SI-SDR are used to evaluate the SE performance of SE algorithms. We show the experimental results at four representative SNR levels (-5dB, 0dB, 5dB, and 10dB): at each SNR level, we randomly select one hour of speech signal to conduct the evaluation.

Table F.2 shows the SI-SDR comparison with a 95% confidence interval in the DNS dataset [68]. Comparing VAE-GAN-L and β-PVAE-L, it is evident that there is a SI-SDR score improvement, which illustrates that adversarial training can effectively improve the decoder's signal estimation performance and generate benefits for the signal reconstruction. Additionally, the performance of mask estimation depends on the accuracy of the signal estimation, so VAE-GAN-M also obtain higher SI-SDR score than β-PVAE-M. Comparing the VAE-GAN-based methods (VAE-GAN-L and VAE-GAN-M) with GAN-SE, we find that all VAE-GAN-based methods can achieve a higher SI-SDR score than GAN-SE, which indicates the importance of representation learning for the GAN-based SE method. A disentangled signal representation can help GANs generate a higher quality target. This verifies our previous hypothesis. Finally, considering that VAE-GAN-M also shows a higher SI-SDR score than NSNet2, the proposed algorithm is quite competitive with the current practical SOTA SE algorithms. In this paper, we choose only a basic DNN structure to conduct the related experiments. Based on the experimental results, we believe that our algorithm has a strong potential to achieve better SE performance if VAE-GAN is applied to a more advanced DNN structure [21].

The STOI comparisons in the DNS dataset [68] are shown in Table F.3, showing

that VAE-GAN-based methods can continuously improve speech intelligibility from –5dB to 10dB. This finding is different from the β-PVAE-based method, in which it is difficult to improve the STOI score in high SNR environments. The comparison between VAE-GAN and β-PVAE indicates that adversarial training can effectively improve speech intelligibility. Meanwhile, comparing VAE-GAN and GAN-SE, we find that VAE-GAN significantly outperforms GAN-SE, which demonstrates the importance of a good disentangled signal representation for improving speech intelligibility. Additionally, Table F.3 indicates that VAE-GAN-M can also obtain higher STOI score than NSNet2.

Table F.4 indicates the PESQ comparison with a 95% confidence interval in the DNS dataset [68]. Moreover, VAE-GAN-L can consistently obtain the highest PESQ score under all four SNR environments. Comparing VAE-GAN-L and β-PVAE-L, we find that VAE-GAN-L obtains a very significant PESQ score improvement (a 0.19 advantage for the average PESQ score.) by introducing adversarial training, which shows the importance of adversarial training in direct signal reconstruction that can mitigate the effects of inaccurate posterior estimation for signal estimation. In addition, it is of interest that VAE-GAN-L is competitive with VAE-GAN-M, a finding that is different from the previous SI-SDR and STOI comparisons. This may indicate that adversarial training is more suitable for improving speech quality [59]. Table F.4 also shows that VAE-GAN-L achieves a higher average PESQ score than NSNet2 [74] (a 0.05 advantage), which indicates the VAE-GAN's benefits for improving speech quality. Finally, it is evident that representation learning is also very important for the GAN-based SE algorithms [59], improving speech quality (VAE-GAN-L outperforms GAN-SE with a 0.31 average PESQ score). Here, we want to indicate that the PESQ results are very noteworthy because VAE-GAN-L-based method that is without noise and mask estimation can outperform the mask-based method VAE-GAN-M. In general, the mask or filter-based methods [7, 8] need to estimate the speech and noise signal for SE. However, based on the experimental results, maybe we need to consider whether we still need to apply mask or filter for SE if we can use DRL or other methods to estimate high-quality speech signals because the filter or mask may also damage the speech signal [7]. This problem will be considered in our following research.

Table F.5 presents the experimental comparisons in the LibriSpeech dataset [69] featuring the average scores of different SNR levels. The results in the LibriSpeech dataset tend to be similar to the results in the DNS dataset [68], which indicates that the proposed algorithm can still achieve satisfactory SE performance for unseen signals. Comparing β-PVAE-L and VAE-GAN-L, it is evident that VAE-GAN-L returns higher SI-SDR, STOI, and PESQ scores than β-PVAE-L, supporting the importance of adversarial training for improving the accuracy of signal estimation. Furthermore, as previously, VAE-GAN-M can produce the best SE performance.

To sum up, we find that the proposed VAE-GAN can achieve the best SE performance compared with the reference methods under the STOI, PESQ, and SI-SDR evaluation metrics. The experimental results demonstrate that: 1) representation learning can help the GAN-based SE method to obtain better SE performance; 2) adversarial training can significantly improve decoders' signal estimation in β-PVAE. Moreover, the comparison of VAE-GAN and NSNet2 [74] shows that VAE-GAN is very competitive with the current SOTA SE algorithms [74, 80]. In this experiment, we only use a basic neural network structure [74]; however, based on the experimental results, we believe that VAE-GAN has a significant potential to achieve better SE performance provided VAE-GAN is applied in more advanced neural networks [81–83].

5 Conclusion and Future Work

In this paper, we propose a two-stage DRL-based (VAE-GAN) SE algorithm. VAE-GAN leverages adversarial training to mitigate the problem of inaccurate posterior estimation in β-PVAE and can reduce the effect of inaccurate posterior estimation towards signal reconstruction, resulting in a more accurate speech estimation from the observed signal. We also compare the proposed VAE-GAN with other related SOTA SE algorithms, and the experimental results show that VAE-GAN can obtain higher STOI, PESQ, and SI-SDR scores and achieve the best SE performance among the competing algorithms. Therefore, the results verify that DRL can significantly improve SE performance for the GAN-based SE method [59], which validates DRL's importance for SE. On the other hand, the results also support that adversarial training is crucial for improving β-PVAE's SE performance. According to the experiments, VAE-GAN can have a significant potential in achieving better SE performance if applied in other advanced neural network structures.

For future work, we propose two ways which may further improve VAE-GAN's SE performance. First, as mentioned before, it is possible to apply the proposed VAE-GAN in more advanced neural network structures. For example, we can consider using complex neural networks [21, 81–83] to perform related prior and posterior estimations in VAE-GAN with complex Gaussian distributions. Second, the proposed VAE-GAN can disentangle different types of latent variables, so it can possible to disentangle phoneme or text latent variables from the observed signal, which means it can be possible to analyze context information when conducting SE, a probability that has not been considered in previous SE methods [1, 17]. Finally, additional SE-related information can be considered to achieve better SE performance.

References

[1] Philipos C Loizou, *Speech enhancement: theory and practice*, CRC press, 2013.

[2] Jinyu Li, Li Deng, Yifan Gong, and Reinhold Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 22, no. 4, pp. 745–777, 2014.

[3] Neena Pandey, Abhipsa Pal, et al., "Impact of digital surge during covid-19 pandemic: A viewpoint on research and practice," *International journal of information management*, vol. 55, pp. 102171, 2020.

[4] Sefik Emre Eskimez, Xiaofei Wang, Min Tang, Hemin Yang, Zirun Zhu, Zhuo Chen, Huaming Wang, and Takuya Yoshioka, "Human Listening and Live Captioning: Multi-Task Training for Speech Enhancement," in *Proc. Interspeech*, 2021, pp. 2686–2690.

[5] Kazuma Iwamoto, Tsubasa Ochiai, Marc Delcroix, Rintaro Ikeshita, Hiroshi Sato, Shoko Araki, and Shigeru Katagiri, "How bad are artifacts?: Analyzing the impact of speech enhancement errors on asr," *arXiv preprint arXiv:2201.06685*, 2022.

[6] Firas Jabloun and Benoît Champagne, "Incorporating the human hearing properties in the signal subspace approach for speech enhancement," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 6, pp. 700–708, 2003.

[7] Jesper Rindom Jensen, Jacob Benesty, and Mads Græsbøll Christensen, "Noise reduction with optimal variable span linear filters," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 24, no. 4, pp. 631–644, 2015.

[8] Knud B Christensen, Mads G Christensen, Jesper B Boldt, and Fredrik Gran, "Experimental study of generalized subspace filters for the cocktail party situation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2016, pp. 420–424.

[9] Sriram Srinivasan, Jonas Samuelsson, and W Bastiaan Kleijn, "Codebook driven short-term predictor parameter estimation for speech enhancement," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 14, no. 1, pp. 163–176, 2005.

[10] Mathew Shaji Kavalekalam, Jesper Kjær Nielsen, Jesper Bünsow Boldt, and Mads Græsbøll Christensen, "Model-based speech enhancement for intelligibility improvement in binaural hearing aids," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 27, no. 1, pp. 99–113, 2018.

[11] Mathew Shaji Kavalekalam, Jesper Kjær Nielsen, Liming Shi, Mads Græsbøll Christensen, and Jesper Boldt, "Online parametric NMF for speech enhancement," in *Proc. European Signal Processing Conf.*, 2018, pp. 2320–2324.

[12] Nasser Mohammadiha, Paris Smaragdis, and Arne Leijon, "Supervised and unsupervised speech enhancement using nonnegative matrix factorization," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 21, no. 10, pp. 2140–2151, 2013.

[13] Yang Xiang, Liming Shi, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, and Mads Græsbøll Christensen, "An NMF-HMM speech enhancement method based on kullback-leibler divergence," in *Proc. Interspeech*, 2020, pp. 2667–2671.

[14] Yang Xiang, Liming Shi, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, and Mads Græsbøll Christensen, "A novel NMF-HMM speech enhancement algorithm based on poisson mixture model," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2021, pp. 721–725.

[15] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 23, no. 1, pp. 7–19, 2014.

[16] Yuxuan Wang, Arun Narayanan, and DeLiang Wang, "On training targets for supervised speech separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 22, no. 12, pp. 1849–1858, 2014.

[17] DeLiang Wang and Jitong Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 26, no. 10, pp. 1702–1726, 2018.

[18] Lei Sun, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "Multiple-target deep learning for lstm-rnn based speech enhancement," in *2017 Hands-free Speech Communications and Microphone Arrays (HSCMA)*, 2017, pp. 136–140.

[19] Yi Luo and Nima Mesgarani, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 27, no. 8, pp. 1256–1266, 2019.

[20] Yang Xiang and Changchun Bao, "A parallel-data-free speech enhancement method using multi-objective learning cycle-consistent generative adversarial network," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 28, pp. 1826–1838, 2020.

[21] Yanxin Hu, Yun Liu, Shubo Lv, Mengtao Xing, Shimin Zhang, Yihui Fu, Jian Wu, Bihong Zhang, and Lei Xie, "DCCRN: Deep Complex Convolution Recurrent Network for Phase-Aware Speech Enhancement," in *Proc. Interspeech*, 2020, pp. 2472–2476.

[22] Andong Li, Wenzhe Liu, Xiaoxue Luo, Chengshi Zheng, and Xiaodong Li, "Icassp 2021 deep noise suppression challenge: Decoupling magnitude and phase optimization with a two-stage deep network," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2021, pp. 6628–6632.

[23] Zhong-Qiu Wang, Gordon Wichern, and Jonathan Le Roux, "On the compensation between magnitude and phase in speech separation," *IEEE Signal Processing Letters*, vol. 28, pp. 2018–2022, 2021.

[24] KHR Chan, Y Yu, C You, H Qi, J Wright, and Y ReduNet Ma, "a white-box deep network from the principle of maximizing rate reduction. arxiv. 2021," *arXiv preprint arXiv:2105.10446*, 2021.

[25] John Wright and Yi Ma, *High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications*, Cambridge University Press, 2022.

[26] Wei-Ning Hsu, Yu Zhang, and James Glass, "Learning latent representations for speech generation and transformation," *Proc. Interspeech*, pp. 1273–1277, 2017.

[27] Wei-Ning Hsu, Yu Zhang, and James Glass, "Unsupervised domain adaptation for robust speech recognition via variational autoencoder-based data augmentation," in *Proc. IEEE Workshop Automatic. Speech Recognition. and Understanding.*, 2017, pp. 16–23.

[28] Wei-Ning Hsu, Yu Zhang, and James Glass, "Unsupervised learning of disentangled and interpretable representations from sequential data," in *Proc. Advances in Neural Inform. Process. Syst.*, 2017, pp. 1876–1887.

[29] Yuying Xie, Thomas Arildsen, and Zheng-Hua Tan, "Disentangled speech representation learning based on factorized hierarchical variational autoencoder with self-supervised objective," in *proc. IEEE International Workshop on Machine Learning for Signal Processing*, 2021, pp. 1–6.

[30] Yoshua Bengio, Aaron Courville, and Pascal Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[31] Xili Dai, Shengbang Tong, Mingyang Li, Ziyang Wu, Kwan Ho Ryan Chan, Pengyuan Zhai, Yaodong Yu, Michael Psenka, Xiaojun Yuan, Heung Yeung Shum, et al., "Closed-loop data transcription to an ldr via minimaxing rate reduction," *arXiv preprint arXiv:2111.06636*, 2021.

[32] Donald S Williamson, Yuxuan Wang, and DeLiang Wang, "Complex ratio masking for monaural speech separation," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 24, no. 3, pp. 483–492, 2015.

[33] Timo Gerkmann, Martin Krawczyk-Becker, and Jonathan Le Roux, "Phase processing for single-channel speech enhancement: History and recent advances," *IEEE signal processing Magazine*, vol. 32, no. 2, pp. 55–66, 2015.

[34] Szu-Wei Fu, Yu Tsao, Xugang Lu, and Hisashi Kawai, "Raw waveform-based speech enhancement by fully convolutional networks," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2017, pp. 006–012.

[35] Szu-Wei Fu, Tao-Wei Wang, Yu Tsao, Xugang Lu, and Hisashi Kawai, "End-to-end waveform utterance enhancement for direct evaluation metrics optimization by fully convolutional neural networks," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 26, no. 9, pp. 1570–1584, 2018.

[36] Daniel Michelsanti, Zheng-Hua Tan, Shi-Xiong Zhang, Yong Xu, Meng Yu, Dong Yu, and Jesper Jensen, "An overview of deep-learning-based audio-visual speech enhancement and separation," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 29, pp. 1368–1396, 2021.

[37] G. Carbajal, J. Richter, and T. Gerkmann, "Disentanglement learning for variational autoencoders applied to audio-visual speech enhancement," in *Proc. IEEE Workshop Appl. of Signal Process. to Aud. and Acoust.* IEEE, 2021, pp. 126–130.

[38] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Process. Lett.*, vol. 21, no. 1, pp. 65–68, 2013.

[39] Yan-Hui Tu, Jun Du, and Chin-Hui Lee, "Speech enhancement based on teacher–student deep learning using improved speech presence probability for noise-robust speech recognition," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 27, no. 12, pp. 2080–2091, 2019.

[40] Simon Leglaive, Laurent Girin, and Radu Horaud, "A variance modeling framework based on variational autoencoders for speech enhancement," in *Proc. IEEE Workshop Machine Learning. Signal Process.*, 2018, pp. 1–6.

[41] Yoshiaki Bando, Masato Mimura, Katsutoshi Itoyama, Kazuyoshi Yoshii, and Tatsuya Kawahara, "Statistical speech enhancement based on probabilistic integration of variational autoencoder and non-negative matrix factorization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 716–720.

[42] Simon Leglaive, Laurent Girin, and Radu Horaud, "Semi-supervised multichannel speech enhancement with variational autoencoders and non-negative matrix factorization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 101–105.

[43] Guillaume Carbajal, Julius Richter, and Timo Gerkmann, "Guided variational autoencoder for speech enhancement with a supervised classifier," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 681–685.

[44] Huajian Fang, Guillaume Carbajal, Stefan Wermter, and Timo Gerkmann, "Variational autoencoder for speech enhancement with a noise-aware encoder," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 676–680.

[45] Diederik P Kingma and Max Welling, "Auto-encoding variational Bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[46] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," *arXiv preprint arXiv:2106.06103*, 2021.

[47] Yongmao Zhang, Jian Cong, Heyang Xue, Lei Xie, Pengcheng Zhu, and Mengxiao Bi, "Visinger: Variational inference with adversarial learning for end-to-end singing voice synthesis," *arXiv preprint arXiv:2110.08813*, 2021.

[48] Yang Xiang, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, and Mads Græsbøll Christensen, "A bayesian permutation training deep representation learning method for speech enhancement with variational autoencoder," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2022, pp. 381–385.

[49] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis, "Deep learning for monaural speech separation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 1562–1566.

[50] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," *International Conference on Learning Representations*, 2017.

[51] Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner, "Understanding disentangling in β-vae," *arXiv preprint arXiv:1804.03599*, 2018.

[52] Yang Xiang, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, and Mads Græsbøll Christensen, "A deep representation learning speech enhancement method using β-vae," *Accepted by Eurosipco (arXiv preprint: arXiv:2205.05581)*, 2022.

[53] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Proc. Advances in Neural Inform. Process. Syst.*, 2014, pp. 2672–2680.

[54] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka, "f-gan: Training generative neural samplers using variational divergence minimization," *Proc. Advances in Neural Inform. Process. Syst.*, vol. 29, 2016.

[55] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther, "Autoencoding beyond pixels using a learned similarity metric," in *International conference on machine learning*. PMLR, 2016, pp. 1558–1566.

[56] Gaurav Parmar, Dacheng Li, Kwonjoon Lee, and Zhuowen Tu, "Dual contradistinctive generative autoencoder," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 823–832.

[57] Huaibo Huang, Ran He, Zhenan Sun, Tieniu Tan, et al., "Introvae: Introspective variational autoencoders for photographic image synthesis," *Proc. Advances in Neural Inform. Process. Syst.*, vol. 31, 2018.

[58] Santiago Pascual, Antonio Bonafonte, and Joan Serrà, "Segan: Speech enhancement generative adversarial network," *Proc. Interspeech*, pp. 3642–3646, 2017.

[59] Daniel Michelsanti and Zheng-Hua Tan, "Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification," *Proc. Interspeech*, pp. 2008–2012, 2017.

[60] Eric A Wan and Alex T Nelson, "Networks for speech enhancement," *Handbook of neural networks for speech processing. Artech House, Boston, USA*, vol. 139, no. 1, pp. 7, 1999.

[61] Fei Xie and Dirk Van Compernolle, "A family of mlp based nonlinear spectral estimators for noise reduction," in *Proceedings of ICASSP'94. IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 1994, vol. 2, pp. II–53.

[62] John Duchi, Elad Hazan, and Yoram Singer, "Adaptive subgradient methods for online learning and stochastic optimization.," *Journal of machine learning research*, vol. 12, no. 7, 2011.

[63] Szu-Wei Fu, Cheng Yu, Kuo-Hsuan Hung, Mirco Ravanelli, and Yu Tsao, "Metricgan-u: Unsupervised speech enhancement/dereverberation based only on noisy/reverberated speech," *arXiv preprint arXiv:2110.05866*, 2021.

[64] Wen-Yuan Ting, Syu-Siang Wang, Hsin-Li Chang, Borching Su, and Yu Tsao, "Speech enhancement based on cyclegan with noise-informed training," *arXiv preprint arXiv:2110.09924*, 2021.

[65] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley, "Least squares generative adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2794–2802.

[66] Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brébisson, Yoshua Bengio, and Aaron C Courville, "Melgan: Generative adversarial networks for conditional waveform synthesis," *Proc. Advances in Neural Inform. Process. Syst.*, vol. 32, 2019.

[67] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *Proc. Advances in Neural Inform. Process. Syst.*, vol. 33, pp. 17022–17033, 2020.

[68] Chandan KA Reddy, Harishchandra Dubey, Kazuhito Koishida, Arun Nair, Vishak Gopal, Ross Cutler, Sebastian Braun, Hannes Gamper, Robert Aichner, and Sriram Srinivasan, "Interspeech 2021 deep noise suppression challenge," in *Proc. Interspeech*, 2021.

[69] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2015, pp. 5206–5210.

[70] Guoning Hu and DeLiang Wang, "A tandem algorithm for pitch estimation and voiced speech segregation," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 18, no. 8, pp. 2067–2079, 2010.

[71] Andrew Varga and Herman JM Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 12, no. 3, pp. 247–251, 1993.

[72] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," in *EMNLP*, 2014.

[73] Chandan KA Reddy, Ebrahim Beyrami, Jamie Pool, Ross Cutler, Sriram Srinivasan, and Johannes Gehrke, "A scalable noisy speech dataset and online subjective test framework," *Proc. Interspeech*, pp. 1816–1820, 2019.

[74] Sebastian Braun and Ivan Tashev, "Data augmentation and loss normalization for deep noise suppression," in *International Conference on Speech and Computer*. Springer, 2020, pp. 79–86.

[75] Harishchandra Dubey, Vishak Gopal, Ross Cutler, Sergiy Matusevych, Sebastian Braun, Emre Sefik Eskimez, Manthan Thakker, Takuya Yoshioka, Hannes Gamper, and Robert Aichner, "Icassp 2022 deep noise suppression challenge," in *ICASSP*, 2022.

[76] Diederik P Kingma and Jimmy Ba, "ADAM: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[77] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R Hershey, "Sdr–half-baked or well done?," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 626–630.

[78] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, 2011.

[79] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2001, vol. 2, pp. 749–752.

[80] Yangyang Xia, Sebastian Braun, Chandan KA Reddy, Harishchandra Dubey, Ross Cutler, and Ivan Tashev, "Weighted speech distortion losses for neural-network-based real-time speech enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2020, pp. 871–875.

[81] C Trabelsi, O Bilaniuk, Y Zhang, D Serdyuk, S Subramanian, JF Santos, S Mehri, N Rostamzadeh, Y Bengio, and CJ Pal, "Deep complex networks," *arXiv preprint arXiv:1705.09792*, 2017.

[82] Hyeong-Seok Choi, Jang-Hyun Kim, Jaesung Huh, Adrian Kim, Jung-Woo Ha, and Kyogu Lee, "Phase-aware speech enhancement with deep complex u-net," in *International Conference on Learning Representations*, 2018.

[83] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

ISSN (online): 2446-1628
ISBN (online): 978-87-7573-783-3

AALBORG UNIVERSITY PRESS