

A Two-Stage Deep Representation Learning-Based Speech Enhancement Method Using Variational Autoencoder and Adversarial Training

Yang Xiang, *Student Member, IEEE*, Jesper Lisby Højvang, Morten Højfeldt Rasmussen, and Mads Græsbøll Christensen, *Senior Member, IEEE*

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.

Index Terms—Deep representation learning, adversarial training, variational autoencoder, speech enhancement, Bayesian permutation training.

I. 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]. These DNN-based SE methods usually apply different structures (e.g. feedforward multilayer perceptron (MLP) [15], [24], convolutional neural network (FCN) [25], and deep recurrent neural networks (DRNN) [26]–[29]) to predict various targets [17]. Unlike classic algorithms [7], [10]–[14], DNNs can learn the disentangled representations of the data [30], and can use the learned representations to generate related data. Thus, we hypothesize that one of the reasons of why DNN can perform SE is that DNN can extract useful speech representation [31] 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 [32]–[35]. 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' principles is that DNNs are based on data representation learning [30], [36], [37], so it can avoid the speech-phase estimation problem (only DNN's input contains the all signal information) [38]–[40] in traditional T–F processes (STFT analysis). More specifically, recent research [40] has indicated that DNN can directly leverage the speech waveform to achieve excellent SE performance [41]. 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

Y. Xiang is an industrial Ph.D. student, associated with the Audio Analysis Lab, CREATE, Aalborg University, Aalborg, Denmark, and Capturi A/S, Aarhus, Denmark; email: yaxi@create.aau.dk

J. L. Højvang and M. H. Rasmussen are with Capturi A/S, Aarhus, Denmark; email: {jlh,mhr}@capturi.com

M. G. Christensen is with the Audio Analysis Lab, CREATE, Aalborg University, Aalborg, Denmark; email: mgc@create.aau.dk

This work was supported by Innovation Fund Denmark (Grant No. 9065-00046).

years [42], [43].

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]–[24] focus on the learning of the training target and apply DNNs only to predict pre-defined targets (e.g., various masks [16], speech spectrum [24], and speech present probability [44]). However, these algorithms ignore the importance of reliable representations for DNN-based methods [36] and do not consider using DNN to obtain better signal representations. Although direct prediction of pre-defined targets can prevent inaccurate signal assumptions [24], 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 [35], [36] but can also lead to a better prediction ability of the DNN [36]. 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 [31], [36]. Furthermore, DRL can disentangle different latent representations of the speech signal (e.g., content and acoustic representation) [32]–[34], 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. DRL plays a crucial role in finding, disentangling, and analyzing intricate speech information during SE, thereby endowing DRL-based SE algorithms with the potential to reduce WER in ASR systems while improving the human listening experience. This potential stems from the ability of DRL-based methods to analyze various speech-related information and mitigate information loss caused by speech distortion, a capability that previous SE algorithms did not possess [17]. As a result, the DRL-based SE strategy holds promise for applications in hearing aids, robust ASR systems, and online meeting systems where reducing WER and achieving accurate live captioning is crucial when transmitting high-quality speech audio.

Due to the importance of DRL for DNN [36], [37], DRL-based SE algorithms have been investigated in recent research works [43], [45]–[49]. These methods mainly use a variational autoencoder (VAE) [50] 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 [50]. Currently, VAE has been widely applied in various tasks related to representation learning [51], [52]. 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 [43], [45]–[49]. This directly results in inaccurate obtained speech representations and possibly unsatisfactory SE performance [46].

To obtain a more accurate speech representation, a novel VAE-based SE method [53], named Bayesian permutation training variational autoencoder (PVAE), was proposed 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 [43], [45]–[49], this model first extracts a more accurate speech representation from the observed signal, because different latent representations are disentangled [53] 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 [53] can be directly adopted by many current SE DNN structures [17] and also directly used to optimize DNN-based SE algorithms [17]. Conducted experiments [53] indicate that this DRL strategy can help the traditional DNN-based SE method [54] achieve a better SE performance.

To further help PVAE to achieve better SE performance, we propose to leverage β -VAE [55], [56] to improve PVAE's representation learning ability. More specifically, the proposed β -PVAE [57] 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 [53]. An experimental analysis in [57] 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 [57]. 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 [53] rely on a conditional assumption [57]. 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 [57], we extend our DRL-based SE framework [53], [57] and propose in this paper a two-stage DRL-based SE method consisting of a representation learning stage [36] and an adversarial training stage [58]. In the first representation learning stage, we leverage the β -PVAE [57] 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 [58], [59]. 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 reconstruction ability and help the VAE obtain higher quality signals [51], [52], [60]–[62]. Therefore, we introduce adversarial training to improve β -PVAE decoders' generative ability.

Recently, a combination of VAE and GAN (VAE-GAN) [60]–[62] has been widely applied in various speech synthesis tasks [51], [52]. VAE-GAN can achieve better performance than independent GAN or VAE-based methods [60], 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 [51], [52], [60] 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 competitive GAN-based SE methods [63], [64], 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.

II. FUNDAMENTALS

A. Signal Model

In this work, we assume that the noisy speech is additive, so the signal model can be written as follows:

$$y(t) = x(t) + d(t), \quad (1)$$

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

$$y_{f,n} = x_{f,n} + d_{f,n}, \quad (2)$$

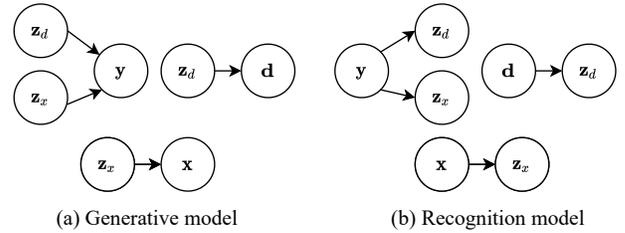


Fig. 1: Graphic illustration of the proposed signal model [53], [57].

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], [65], [66]. At present, LPS, as the input feature, has been widely applied in the DNN-based SE algorithms [15], [17], [65], [66]. 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 (1) and (2) are not used. The purpose of (1) and (2) is used to generate noisy signal. Furthermore, (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.

Following our preliminary work [53], [57], 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 [36], [50]. 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. 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 $p(\mathbf{z}_d|\mathbf{y})$ [53], based on the VAE's property [50]. Fig. 1(b) shows the recognition process [50]. 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 [53].

B. VAE and β -VAE

The original VAE is a probabilistic generative model [50] 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 [50] using the

stochastic gradient descent (SGD) or Adagrad [67] algorithm. Maximized, the ELBO can be written as follows:

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

where $D_{KL}(\cdot||\cdot)$ 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 [55], [56] is a modification of the original VAE framework, which introduces an adjustable hyperparameter β in the KL divergence term:

$$\begin{aligned} \mathcal{L}_\beta &= \beta \mathbb{E}_{\mathbf{y} \sim p(\mathbf{y})} [D_{KL}(p(\mathbf{z}_y|\mathbf{y})||q(\mathbf{z}_y))] \\ &\quad - \mathbb{E}_{\mathbf{y} \sim p(\mathbf{y})} [\mathbb{E}_{\mathbf{z}_y \sim p(\mathbf{z}_y|\mathbf{y})} [\log q(\mathbf{y}|\mathbf{z}_y)]]. \end{aligned} \quad (4)$$

β -VAE aims to help the original VAE [50] to obtain a better signal representation. In general, $\beta > 1$ results in more disentangled latent representations [55]. A higher value of β can encourage learning a more disentangled representation.

C. PVAE

Our preliminary work proposed a PVAE-based SE algorithm [53] and indicated that PVAE can help the current DNN-based SE method [54] obtain better signal representations (because different latent representations are disentangled [53] and these representations are expressed in a low-dimension space [36]) 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. 2(a) shows the PVAE framework [53]. 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 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 [50]:

$$\begin{aligned} \mathcal{L}_c(\theta_x, \varphi_x; \mathbf{x}) &= \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \{D_{KL}(p(\mathbf{z}_x|\mathbf{x})||q(\mathbf{z}_x)) \\ &\quad - \mathbb{E}_{\mathbf{z}_x \sim p(\mathbf{z}_x|\mathbf{x})} [\log q(\mathbf{x}|\mathbf{z}_x)]\}, \end{aligned} \quad (5)$$

$$\begin{aligned} \mathcal{L}_d(\theta_d, \varphi_d; \mathbf{d}) &= \mathbb{E}_{\mathbf{d} \sim p(\mathbf{d})} \{D_{KL}(p(\mathbf{z}_d|\mathbf{d})||q(\mathbf{z}_d)) \\ &\quad - \mathbb{E}_{\mathbf{z}_d \sim p(\mathbf{z}_d|\mathbf{d})} [\log q(\mathbf{d}|\mathbf{z}_d)]\}, \end{aligned} \quad (6)$$

where $\theta_x, \varphi_x, \theta_d, \varphi_d$ are the DNN parameters for the related probability estimation [53]: 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. In this paper, we assign the symbol θ to represent the encoder-related parameters, while the symbol φ is used to represent the decoder-related parameters. 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 [53], the NS-VAE's training loss function is expressed as follows:

$$\begin{aligned} \mathcal{L}_p(\theta_y, \varphi_y; \mathbf{y}) &= \mathbb{E}_{\mathbf{y} \sim p(\mathbf{y}), \mathbf{x} \sim p(\mathbf{x})} \{D_{KL}(p(\mathbf{z}_x|\mathbf{y})||p(\mathbf{z}_x|\mathbf{x})) \\ &\quad + \mathbb{E}_{\mathbf{z}_x \sim p(\mathbf{z}_x|\mathbf{y})} [\log \frac{p(\mathbf{z}_x|\mathbf{x})}{q(\mathbf{z}_x)}]\} \\ &\quad + \mathbb{E}_{\mathbf{y} \sim p(\mathbf{y}), \mathbf{d} \sim p(\mathbf{d})} \{D_{KL}(p(\mathbf{z}_d|\mathbf{y})||p(\mathbf{z}_d|\mathbf{d})) \\ &\quad + \mathbb{E}_{\mathbf{z}_d \sim p(\mathbf{z}_d|\mathbf{y})} [\log \frac{p(\mathbf{z}_d|\mathbf{d})}{q(\mathbf{z}_d)}]\} \\ &\quad - \mathbb{E}_{\mathbf{y} \sim p(\mathbf{y})} [\mathbb{E}_{\mathbf{z}_d, \mathbf{z}_x \sim p(\mathbf{z}_d, \mathbf{z}_x|\mathbf{y})} [\log q(\mathbf{y}|\mathbf{z}_x, \mathbf{z}_d)]], \end{aligned} \quad (7)$$

where θ_y and φ_y are the NS-VAE's encoder and decoder parameters, respectively. In (7), KL divergence constraints for speech and noise latent variables are present. These constraints enable us to estimate the desired posterior distributions ($p(\mathbf{z}_d|\mathbf{y})$ and $p(\mathbf{z}_x|\mathbf{y})$) from the noisy signal in a supervised manner. Furthermore, the inclusion of KL divergence terms ensures that the speech and noise signals can be effectively separated in the low-dimensional representation space.

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 (5) and (6). After that, the C-VAE and N-VAE are frozen, and NS-VAE is trained by (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.

D. β -PVAE

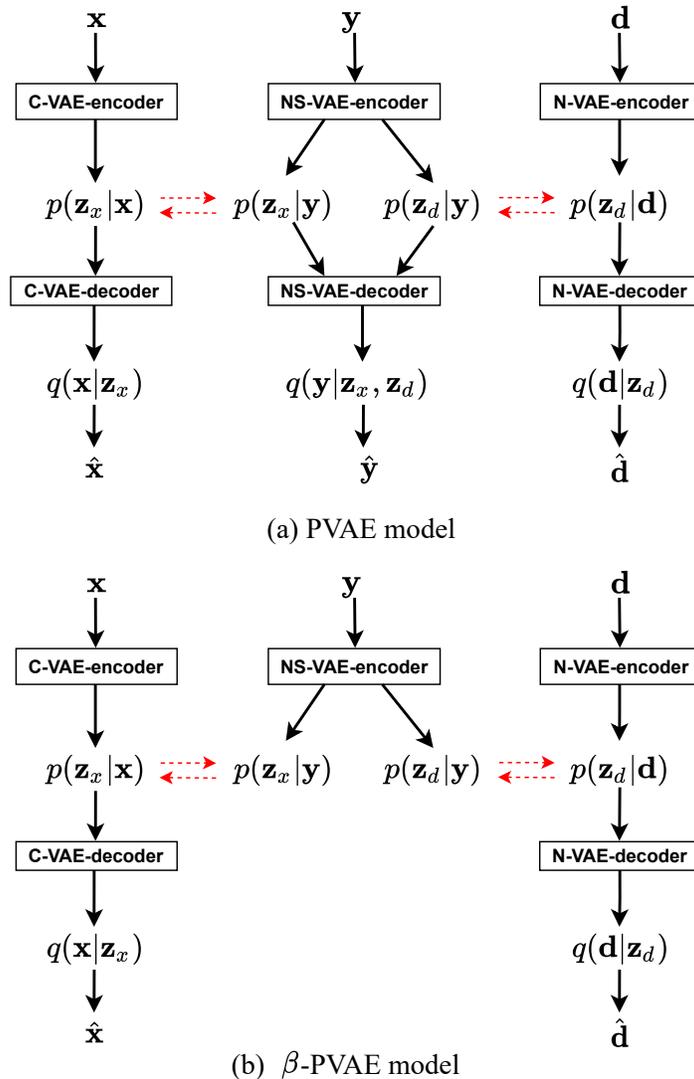
To further improve PVAE's SE performance, we propose to leverage β -VAE to improve PVAE's disentangling ability [57] in our another preliminary work. Furthermore, the proposed β -PVAE makes the best use of the β -VAE's trade-off property to simplify the PVAE's network structure and training parameters by setting β to infinity and discarding the noisy speech restoration term [57], which means that β -PVAE can achieve a better disentangling and enhancement performance than PVAE with a simpler structure. Based on our derivations [53], [57], the β -PVAE's optimization target for $\beta \rightarrow +\infty$ is [57]

$$\begin{aligned} \mathcal{L}_{\beta p}(\theta_y; \mathbf{y}) &= \mathbb{E}_{\mathbf{y} \sim p(\mathbf{y}), \mathbf{x} \sim p(\mathbf{x})} \{D_{KL}(p(\mathbf{z}_x|\mathbf{y})||p(\mathbf{z}_x|\mathbf{x})) \\ &\quad + \mathbb{E}_{\mathbf{z}_x \sim p(\mathbf{z}_x|\mathbf{y})} [\log \frac{p(\mathbf{z}_x|\mathbf{x})}{q(\mathbf{z}_x)}]\} \\ &\quad + \mathbb{E}_{\mathbf{y} \sim p(\mathbf{y}), \mathbf{d} \sim p(\mathbf{d})} \{D_{KL}(p(\mathbf{z}_d|\mathbf{y})||p(\mathbf{z}_d|\mathbf{d})) \\ &\quad + \mathbb{E}_{\mathbf{z}_d \sim p(\mathbf{z}_d|\mathbf{y})} [\log \frac{p(\mathbf{z}_d|\mathbf{d})}{q(\mathbf{z}_d)}]\}. \end{aligned} \quad (8)$$

Comparing (8) with (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. 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 (8) rather than (7).

E. Generative Adversarial Network (GAN)

A GAN [58] consists of two networks: a generator network and a discriminator network. The generator network $G(\mathbf{z})$


 Fig. 2: Model illustration of PVAE and β -PVAE [53], [57].

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 $q(\mathbf{z})$ [59]. 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 [59]. In Jensen–Shannon (JS) divergence, GANs is optimized by the minimax of the loss function [58]:

$$\min_G \max_D \mathcal{L}_{gan}(G, D) = \mathbb{E}_{\mathbf{y} \sim q_{data}(\mathbf{y})} [\log(D(\mathbf{y}))] + \mathbb{E}_{\mathbf{z} \sim q(\mathbf{z})} [\log(1 - D(G(\mathbf{z})))] \quad (9)$$

GANs have been applied in SE [63], [64], [68], [69], 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 [63], [64]. 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 [51], [52].

III. SPEECH ENHANCEMENT WITH VAE AND GAN

To obtain a higher quality enhanced speech, in this paper, we extend DRL-based SE framework [57]. 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, representation learning, 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 accurate knowledge of the input sample distribution [58], [59] (it only needs samples) and it can also improve VAE decoder’s generative performance [60]–[62], 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

Algorithm 1 Representation Learning.

Pre-train 1: Using the speech dataset and loss function (5) to train a general speech VAE (C-VAE) [50].

Pre-train 2: Using the noise dataset and loss function (6) to train a general noise VAE (N-VAE) [50].

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) $\mu_{\theta_{yx}}(\mathbf{y}), \sigma_{\theta_{yx}}^2(\mathbf{y}), \mu_{\theta_{yd}}(\mathbf{y}), \sigma_{\theta_{yd}}^2(\mathbf{y}) = G_{\theta_y}(\mathbf{y})$,
 - (2) $\mu_{\theta_x}(\mathbf{x}), \sigma_{\theta_x}^2(\mathbf{x}) = G_{\theta_x}(\mathbf{x})$,
 - (3) $\mu_{\theta_d}(\mathbf{d}), \sigma_{\theta_d}^2(\mathbf{d}) = G_{\theta_d}(\mathbf{d})$;
4. Calculate loss function (8);
5. Freeze C-VAE and N-VAE and apply the SGD algorithm to update the NS-VAE's parameters θ_y [50];

until the convergence of the loss function.

Return: The trained NS-VAE (G_{θ_x}).

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 SE.

A. Stage 1: Representation Learning

In the first stage, we aim to disentangle speech and noise latent variables from the observed signal. This process is accomplished by the proposed β -PVAE [57]. The purpose of the representation learning stage is to separate speech and noise signals in the low-dimensional representation space.

In β -PVAE, C-VAE and N-VAE are optimized by (5) and (6), respectively, and NS-VAE is optimized by (8). To calculate (5), (6), and (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 [50], which is similar to the previous VAE-based SE methods [45]–[49]. For NS-VAE, we have

$$\begin{aligned} p(\mathbf{z}_x|\mathbf{y}) &= \mathcal{N}(\mathbf{z}_x; \mu_{\theta_{yx}}(\mathbf{y}), \sigma_{\theta_{yx}}^2(\mathbf{y})\mathbf{I}) \\ p(\mathbf{z}_d|\mathbf{y}) &= \mathcal{N}(\mathbf{z}_d; \mu_{\theta_{yd}}(\mathbf{y}), \sigma_{\theta_{yd}}^2(\mathbf{y})\mathbf{I}), \end{aligned} \quad (10)$$

where \mathbf{I} is the identity matrix; $\mu_{\theta_{yx}}(\mathbf{y}), \sigma_{\theta_{yx}}^2(\mathbf{y}), \mu_{\theta_{yd}}(\mathbf{y}),$ and $\sigma_{\theta_{yd}}^2(\mathbf{y})$ 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

$$\begin{aligned} p(\mathbf{z}_x|\mathbf{x}) &= \mathcal{N}(\mathbf{z}_x; \mu_{\theta_x}(\mathbf{x}), \sigma_{\theta_x}^2(\mathbf{x})\mathbf{I}) \\ q(\mathbf{x}|\mathbf{z}_x) &= \mathcal{N}(\mathbf{x}; \mu_{\varphi_x}(\mathbf{z}_x), \sigma_{\varphi_x}^2(\mathbf{z}_x)\mathbf{I}), \end{aligned} \quad (11)$$

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:

$$\mu_{\theta_{yx}}(\mathbf{y}), \sigma_{\theta_{yx}}^2(\mathbf{y}), \mu_{\theta_{yd}}(\mathbf{y}), \sigma_{\theta_{yd}}^2(\mathbf{y}) = G_{\theta_y}(\mathbf{y});$$
4. Apply the reparameterization trick to obtain sample $\mathbf{z}_x \sim p(\mathbf{z}_x|\mathbf{y})$ and $\mathbf{z}_d \sim p(\mathbf{z}_d|\mathbf{y})$ [58];
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 (14), (15), (16), (17);
5. Freeze all encoders and apply SGD to update parameters $\varphi_x, \varphi_d, \theta_{dx},$ 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 $D_{\theta_{dd}}.$

where $\mu_{\theta_x}(\mathbf{x})$ and $\sigma_{\theta_x}^2(\mathbf{x})$ are obtained by C-VAE's encoder $G_{\theta_x}(\mathbf{x})$ with parameter θ_x , and $\mu_{\varphi_x}(\mathbf{z}_x)$ and $\sigma_{\varphi_x}^2(\mathbf{z}_x)$ can be estimated by C-VAE's decoder $G_{\varphi_x}(\mathbf{z}_x)$ with parameter φ_x . Similarly, for N-VAE, we have

$$\begin{aligned} p(\mathbf{z}_d|\mathbf{d}) &= \mathcal{N}(\mathbf{z}_d; \mu_{\theta_d}(\mathbf{d}), \sigma_{\theta_d}^2(\mathbf{d})\mathbf{I}) \\ q(\mathbf{d}|\mathbf{z}_d) &= \mathcal{N}(\mathbf{d}; \mu_{\varphi_d}(\mathbf{z}_d), \sigma_{\varphi_d}^2(\mathbf{z}_d)\mathbf{I}), \end{aligned} \quad (12)$$

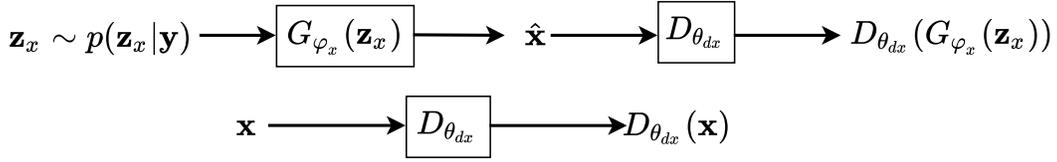
where $\mu_{\theta_d}(\mathbf{d})$ and $\sigma_{\theta_d}^2(\mathbf{d})$ are obtained by C-VAE's encoder $G_{\theta_d}(\mathbf{d})$ with parameter θ_x , and $\mu_{\varphi_d}(\mathbf{z}_d)$ and $\sigma_{\varphi_d}^2(\mathbf{z}_d)$ can be estimated by C-VAE's decoder $G_{\varphi_d}(\mathbf{z}_d)$ with 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

$$\begin{aligned} q(\mathbf{z}_x) &= \mathcal{N}(\mathbf{z}_x; \mathbf{0}, \mathbf{I}) \\ q(\mathbf{z}_d) &= \mathcal{N}(\mathbf{z}_d; \mathbf{0}, \mathbf{I}). \end{aligned} \quad (13)$$

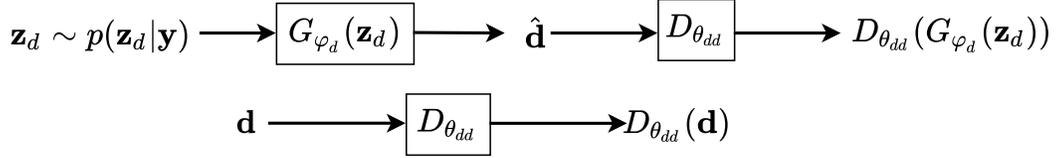
The entire representation learning process is summarized in Algorithm 1.

B. Stage 2: Adversarial Training

The second training stage aims to improve the decoders' robustness and signal restoration ability in β -PVAE for better SE performance. It is difficult to ensure that disentangled speech and noise latent representations are consistently accurate in complex noisy environments. Considering that decoders' SE performance relies on 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 [58], [63]. Moreover, adversarial training can usually improve decoder's signal restoration ability [60]–[62]. 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. Note that the signal separation process mainly occurs during the representation learning stage.



(a) Adversarial training for speech



(b) Adversarial training for noise

Fig. 3: Graphic illustration of adversarial training.

In the adversarial training stage, the main role of the decoders is to convert low-dimensional representations back to high-dimensional signals, focusing on signal reconstruction rather than signal separation.

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. 3 shows the related adversarial training process. In this work, we use the least squares GAN [70] loss function for adversarial training, which has been widely used in various GAN applications [51], [52] as it can achieve a more stable training process and avoid the problem of vanishing gradients, compared to the original GAN [58] loss function. Moreover, although GAN can generate high-quality signals, GAN may diverge too much from the target signals [60]–[62]. 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 [51], [52], [63], [64], [71], [72]. Therefore, the adversarial loss function for C-VAE-decoder can be expressed as follows:

$$\mathcal{L}_{gan_c}(G_{\varphi_x}) = \mathbb{E}_{\mathbf{z}_x \sim p(\mathbf{z}_x|\mathbf{y})} [(D_{\theta_{dx}}(G_{\varphi_x}(\mathbf{z}_x)) - 1)^2] - \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x}|\mathbf{y})} [\log q(\mathbf{x}|\mathbf{z}_x)], \quad (14)$$

$$\mathcal{L}_{gan_c}(D_{\theta_{dx}}) = \mathbb{E}_{\mathbf{z}_x \sim p(\mathbf{z}_x|\mathbf{y})} [(D_{\theta_{dx}}(G_{\varphi_x}(\mathbf{z}_x)))^2] + \mathbb{E}_{\mathbf{x} \sim q_{data}(\mathbf{x})} [(D_{\theta_{dx}}(\mathbf{x}) - 1)^2]. \quad (15)$$

Similarly, the adversarial loss function for noise can be represented as

$$\mathcal{L}_{gan_d}(G_{\varphi_d}) = \mathbb{E}_{\mathbf{z}_d \sim p(\mathbf{z}_d|\mathbf{y})} [(D_{\theta_{dd}}(G_{\varphi_d}(\mathbf{z}_d)) - 1)^2] - \mathbb{E}_{\mathbf{d} \sim p(\mathbf{d}|\mathbf{y})} [\log q(\mathbf{d}|\mathbf{z}_d)], \quad (16)$$

$$\mathcal{L}_{gan_d}(D_{\theta_{dd}}) = \mathbb{E}_{\mathbf{z}_d \sim p(\mathbf{z}_d|\mathbf{y})} [(D_{\theta_{dd}}(G_{\varphi_d}(\mathbf{z}_d)))^2] + \mathbb{E}_{\mathbf{d} \sim q_{data}(\mathbf{d})} [(D_{\theta_{dd}}(\mathbf{d}) - 1)^2]. \quad (17)$$

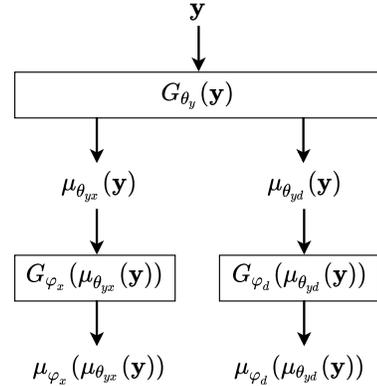


Fig. 4: VAE-GAN for online SE.

Algorithm 3 VAE-GAN-based SE.

- 1: Apply the observed signal \mathbf{y} as the NS-VAE's encoder (G_{θ_y}) input;
2. Estimate the posterior probability $p(\mathbf{z}_x|\mathbf{y})$ and $p(\mathbf{z}_d|\mathbf{y})$ by: $\mu_{\theta_{yx}}(\mathbf{y}), \sigma_{\theta_{yx}}^2(\mathbf{y}), \mu_{\theta_{yd}}(\mathbf{y}), \sigma_{\theta_{yd}}^2(\mathbf{y}) = G_{\theta_y}(\mathbf{y})$;
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 $\hat{\mathbf{x}}$.

The complete adversarial training process is summarized in Algorithm 2.

C. VAE-GAN for Speech Enhancement

The SE stage requires only the NS-VAE encoder G_{θ_y} , C-VAE decoder G_{φ_x} , and N-VAE decoder G_{φ_d} to conduct SE,

which is similar to PVAE [53] and β -PVAE [57]. 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 $\mu_{\varphi_x}(\mu_{\theta_{yx}}(\mathbf{y}))$ 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. 4 and Algorithm 3. In the SE stage, the means are used directly to estimate the signals, without the reparameterization trick [50], which is different from the training process [50]. 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].

IV. 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 competitive algorithms as the reference methods to investigate VAE-GAN’s SE performance.

A. Datasets

In this work, we created a training and test dataset using the speech and noise from the DNS challenge 2021 corpus [73]. 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 [73]). After that, the corresponding training, validation, and test corpus for speech and noise were randomly mixed using the DNS script [73] 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 [73]. 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 [73].

We also used the LibriSpeech [74], 100 environmental noises [75], and NOISEX-92 database [76] 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 [75] and the NOISEX-92 database [76]. The mixed SNRs were randomly chosen from the -10dB to 15dB . Finally, we obtained a one-hour noisy speech test data.

B. Experimental Setup

In our experiment, the signal frame length was 512 samples (32 ms) with a frame shift of 256 samples. A STFT analysis

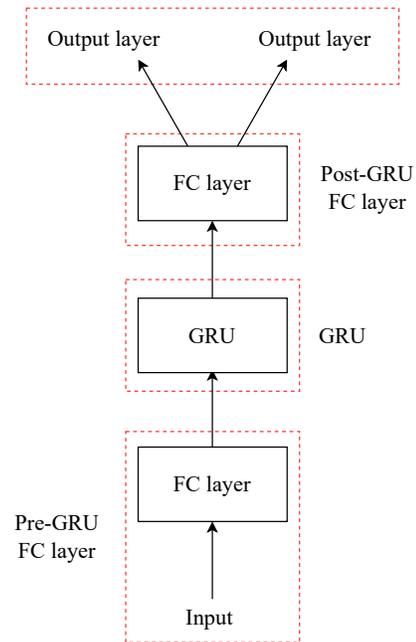


Fig. 5: Network structure in VAE-GAN.

was used to compute the DFT of each overlapping windowed frame. 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 (GRU) [77] due to their computational efficiency and superior performance in SE [78]. In this work, we stacked GRU layers after the fully-connected (FC) layers, followed by hidden FC layers and FC output layers (Figure 5). This network design was similar to the baseline algorithm [79] in DNS challenge 2022 [80]. The detailed model design of each neural network is shown in Table I, 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 [81] with a 128 mini-batch size. The learning rate is 0.001. We conducted the experiments using the Python programming language and the PyTorch toolkit [82].

C. Evaluation Metrics and Reference Methods

In this work, we will use the scale-invariant signal-to-distortion ratio (SI-SDR) in decibel (dB) [83], short-time objective intelligibility (STOI) [84], and perceptual evaluation of speech quality (PESQ) [85] as evaluation metrics to evaluate the proposed VAE-GAN’s SE performance. SI-SDR is used

TABLE I: Network Details of VAE-GAN

Networks	Pre-GRU FC layer			GRU layer		Post-GRU FC layer			Output layer		
	Number	Nodes	AF	Number	Nodes	Number	Nodes	AF	Number	Nodes	AF
G_{θ_x} and G_{θ_d}	3	257-512-512	ReLU	1	512	0	N/A	N/A	2	128	Linear
G_{φ_x} and G_{φ_d}	1	128	ReLU	1	512	2	512-512	ReLU	2	257	Linear
G_{θ_y}	3	257-512-512	ReLU	1	512	1	512	ReLU	4	128	Linear
$D_{\theta_{dx}}$ and $D_{\theta_{dd}}$	2	257-512	ReLU	1	256	1	512	ReLU	1	1	Linear

TABLE II: SI-SDR Comparison in DNS dataset with a 95% confidence interval

SNR (dB)	Noise	GAN-SE [64]	Y-SE-L [54]	Y-SE-M [54]	NSNet2 [79]	β -PVAE-L [57]	VAE-GAN-L	β -PVAE-M [57]	VAE-GAN-M
-5	-4.40 (± 0.80)	2.15 (± 0.79)	1.88 (± 0.78)	2.61 (± 0.84)	5.07 (± 0.74)	2.63 (± 0.80)	4.52 (± 0.72)	3.52 (± 0.93)	5.37 (± 0.89)
0	2.63 (± 1.04)	6.79 (± 0.61)	5.24 (± 0.60)	5.66 (± 0.89)	9.77 (± 0.81)	5.69 (± 0.59)	8.48 (± 0.52)	8.92 (± 0.92)	10.17 (± 0.86)
5	7.63 (± 1.08)	9.30 (± 0.50)	7.02 (± 0.54)	7.99 (± 0.88)	13.09 (± 0.82)	8.10 (± 0.46)	10.96 (± 0.39)	12.96 (± 0.93)	14.11 (± 0.85)
10	13.58 (± 1.05)	11.75 (± 0.42)	9.02 (± 0.44)	10.16 (± 0.81)	16.76 (± 0.72)	10.46 (± 0.35)	13.07 (± 0.30)	17.75 (± 0.88)	18.58 (± 0.84)
Average	4.86 (± 0.99)	7.49 (± 0.58)	5.79 (± 0.59)	6.61 (± 0.86)	11.17 (± 0.77)	6.72 (± 0.55)	9.26 (± 0.48)	10.78 (± 0.91)	12.06 (± 0.86)

TABLE III: STOI (%) Comparison in DNS dataset with a 95% confidence interval

SNR (dB)	Noise	GAN-SE [64]	Y-SE-L [54]	Y-SE-M [54]	NSNet2 [79]	β -PVAE-L [57]	VAE-GAN-L	β -PVAE-M [57]	VAE-GAN-M
-5	73.80 (± 1.70)	72.26 (± 1.91)	71.13 (± 1.89)	72.44 (± 1.92)	78.15 (± 1.61)	72.94 (± 1.77)	76.83 (± 1.81)	77.27 (± 1.71)	79.29 (± 1.80)
0	82.46 (± 1.40)	81.47 (± 1.42)	81.02 (± 1.44)	82.01 (± 1.51)	87.03 (± 1.12)	82.23 (± 1.32)	85.62 (± 1.18)	86.02 (± 1.25)	87.06 (± 1.19)
5	88.01 (± 1.11)	87.02 (± 1.02)	86.99 (± 1.01)	87.26 (± 0.93)	91.63 (± 0.81)	87.57 (± 0.99)	90.71 (± 0.80)	91.08 (± 0.91)	92.01 (± 0.82)
10	93.54 (± 0.72)	92.13 (± 0.61)	92.01 (± 0.71)	92.92 (± 0.73)	95.59 (± 0.47)	92.54 (± 0.59)	94.68 (± 0.46)	95.58 (± 0.51)	96.02 (± 0.47)
Average	84.45 (± 1.23)	83.22 (± 1.24)	82.79 (± 1.26)	83.66 (± 1.27)	88.10 (± 1.09)	83.82 (± 1.00)	86.96 (± 1.06)	87.48 (± 1.09)	88.60 (± 1.07)

TABLE IV: PESQ Comparison in DNS dataset with a 95% confidence interval

SNR (dB)	Noise	GAN-SE [64]	Y-SE-L [54]	Y-SE-M [54]	NSNet2 [79]	β -PVAE-L [57]	VAE-GAN-L	β -PVAE-M [57]	VAE-GAN-M
-5	1.81 (± 0.02)	2.00 (± 0.03)	1.92 (± 0.02)	2.04 (± 0.03)	2.28 (± 0.02)	2.08 (± 0.03)	2.31 (± 0.02)	2.19 (± 0.03)	2.30 (± 0.02)
0	2.04 (± 0.02)	2.33 (± 0.02)	2.31 (± 0.03)	2.40 (± 0.02)	2.60 (± 0.02)	2.46 (± 0.03)	2.64 (± 0.02)	2.55 (± 0.02)	2.62 (± 0.01)
5	2.28 (± 0.02)	2.62 (± 0.02)	2.61 (± 0.02)	2.70 (± 0.02)	2.87 (± 0.02)	2.77 (± 0.02)	2.94 (± 0.01)	2.85 (± 0.02)	2.93 (± 0.01)
10	2.70 (± 0.01)	3.00 (± 0.01)	3.01 (± 0.02)	3.12 (± 0.01)	3.24 (± 0.01)	3.14 (± 0.01)	3.29 (± 0.01)	3.21 (± 0.01)	3.29 (± 0.01)
Average	2.21 (± 0.02)	2.49 (± 0.02)	2.46 (± 0.02)	2.57 (± 0.02)	2.75 (± 0.02)	2.61 (± 0.03)	2.80 (± 0.02)	2.70 (± 0.02)	2.79 (± 0.01)

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 enhance the evaluation of speech

enhancement (SE) performance on unseen datasets, we also employ DNSMOS P.835 [86]–[88]. This metric allows us to assess the speech quality (SIG), background noise quality (BAK), and overall quality (P808 MOS) of the audio samples. DNSMOS P.835 has been shown to highly align with human

TABLE V: Experimental result comparisons in LibriSpeech dataset with a 95% confidence interval

Evaluation Metrics	Noise	GAN-SE [64]	Y-SE-L [54]	Y-SE-M [54]	NSNet2 [79]	β -PVAE-L [57]	VAE-GAN-L	β -PVAE-M [57]	VAE-GAN-M
SI-SDR	1.81 (± 0.23)	6.16 (± 0.36)	5.94 (± 0.46)	6.20 (± 0.52)	9.20 (± 0.70)	6.40 (± 0.45)	8.24 (± 0.50)	7.04 (± 0.46)	10.18 (± 0.56)
STOI (%)	82.75 (± 1.63)	80.86 (± 1.69)	80.04 (± 1.71)	80.92 (± 1.60)	86.03 (± 1.51)	81.56 (± 1.53)	84.50 (± 1.47)	85.32 (± 1.53)	86.05 (± 1.50)
PESQ	2.31 (± 0.03)	2.52 (± 0.02)	2.49 (± 0.02)	2.53 (± 0.03)	2.69 (± 0.01)	2.54 (± 0.03)	2.71 (± 0.02)	2.67 (± 0.02)	2.72 (± 0.01)
SIG	2.87 (± 0.05)	2.78 (± 0.04)	3.00 (± 0.04)	3.03 (± 0.05)	3.05 (± 0.03)	3.12 (± 0.04)	3.14 (± 0.05)	3.13 (± 0.04)	3.16 (± 0.04)
BAK	2.30 (± 0.03)	3.36 (± 0.04)	3.30 (± 0.03)	3.45 (± 0.03)	3.68 (± 0.04)	3.70 (± 0.04)	3.77 (± 0.03)	3.44 (± 0.04)	3.83 (± 0.03)
P808 MOS	2.92 (± 0.03)	3.11 (± 0.04)	3.08 (± 0.05)	3.16 (± 0.03)	3.44 (± 0.04)	3.18 (± 0.04)	3.49 (± 0.03)	3.30 (± 0.04)	3.62 (± 0.04)

ratings for speech quality evaluation, making it a effective measure for our purposes.

To better evaluate the proposed VAE-GAN’s SE performance, we choose three related competitive SE algorithms as reference methods. The first reference method is GAN-SE [64], which is a competitive GAN-based SE algorithm that can help us verify whether the better 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 the DNN-based SE algorithm. The second reference method is β -PVAE [57]. 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). In addition to the aforementioned methods, we also conducted a direct comparison between our proposed method and Y-SE [54]. Y-SE utilizes the same DNN-based SE architecture as our approach but is trained without the use of VAE and GAN. The only difference between Y-SE and our method lies in the training strategy. Y-SE is essentially an end-to-end trained model without the inclusion of specific disentanglement or DRL techniques. By comparing our method directly with Y-SE [54], we can clearly observe the impact and benefits that our proposed approach brings to a general DNN-based SE framework. Finally, we compare the proposed VAE-GAN with the DNS 2021 challenge baseline NSNet2 [79], [89] to see whether the VAE-GAN’s SE performance is competitive with the current popular SE algorithms [79]. The main purpose of our experiment is not to outperform all state-of-the-art (SOTA) performance, but to authentically verify the validity of the proposed VAE-GAN framework and its further potential.

For the Y-SE, 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 [57], 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; Y-SE-L and Y-SE-M denote that the enhanced speech is obtained by the Y-SE [54] 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.

D. 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 II shows the SI-SDR comparison with a 95% confidence interval in the DNS dataset [73]. 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 and Y-SE-based methods (Y-SE-L and Y-SE-M), we find that all VAE-GAN-based methods can achieve a higher SI-SDR score than GAN-SE and Y-SE-based methods, which indicates the importance of representation learning for some DNN-based SE frameworks. 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 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 [73] are shown in Table III, showing that VAE-GAN-based methods can con-

tinuously 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, GAN-SE and Y-SE-based methods, we find that VAE-GAN significantly outperforms other two methods, which demonstrates the importance of a good disentangled signal representation for improving speech intelligibility. Additionally, Table III indicates that VAE-GAN-M can also obtain higher STOI score than NSNet2.

Table IV indicates the PESQ comparison with a 95% confidence interval in the DNS dataset [73]. 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 [64]. Table IV also shows that VAE-GAN-L achieves a higher average PESQ score than NSNet2 [79] (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 [64] and Y-SE-based SE algorithms [54], 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, or directly predict masks or complex filters 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.

In conclusion, by comparing our VAE-GAN-based method with the Y-SE-based method, we can clearly observe the significant impact and benefits that our proposed approach brings to the general DNN-based speech enhancement (SE) framework. This comparison effectively demonstrates the added value of incorporating VAE and GAN in the training process. The use of VAE helps in learning meaningful representations and disentangling latent variables, while GAN enhances the robustness and generative capabilities of the model. Together, these components contribute to the superior performance and improved results achieved by our proposed approach.

Table V presents the experimental comparisons in the LibriSpeech dataset [74] 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 [73], 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. Moreover, when comparing VAE-GAN-M with NSNet2 using DNSMOS P.835 evaluation metrics (SIG, BAK, and P808 MOS), we observe that VAE-GAN-M exhibits a notable advantage in enhancing the human listening experience. These results show the benefits of our algorithm in improving the subjective perception of speech quality.

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 [79] shows that VAE-GAN is very competitive with the current SE algorithms [79], [89]. In this experiment, we only use a basic neural network structure [79]; 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 [90]–[92].

V. 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 competitive 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 [64], 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], [90]–[92] to perform related prior and posterior estimations in VAE-GAN with complex Gaussian distributions. In addition, we can also apply real-world recordings to evaluate the SE performance of related SE algorithms. Second, the proposed VAE-GAN can

disentangle different types of latent variables, so it can possible to disentangle phoneme or emotional 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.

ACKNOWLEDGMENTS

This work was supported by Innovation Fund Denmark (Grant No. 9065-00046).

REFERENCES

- [1] P. C. Loizou, *Speech enhancement: theory and practice*. CRC press, 2013.
- [2] 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.
- [3] 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.
- [4] 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.
- [5] 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.
- [6] 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.
- [7] 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.
- [8] 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.
- [9] 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.
- [10] 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.
- [11] 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.
- [12] 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.
- [13] 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.
- [14] —, "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] 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.
- [16] 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.
- [17] 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.
- [18] 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.
- [19] 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.
- [20] 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.
- [21] 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.
- [22] 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.
- [23] 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.
- [24] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Process. Lett.*, vol. 21, no. 1, pp. 65–68, 2013.
- [25] S. R. Park and J. Lee, "A fully convolutional neural network for speech enhancement," *arXiv preprint arXiv:1609.07132*, 2016.
- [26] H. Jacobsson, "Rule extraction from recurrent neural networks: Ataxonomy and review," *Neural Comput.*, vol. 17, no. 6, pp. 1223–1263, 2005.
- [27] 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.
- [28] M. Keshavarzi, T. Goehring, J. Zakis, R. E. Turner, and B. C. Moore, "Use of a deep recurrent neural network to reduce wind noise: Effects on judged speech intelligibility and sound quality," *Trends in hearing*, vol. 22, p. 2331216518770964, 2018.
- [29] T. Goehring, M. Keshavarzi, R. P. Carlyon, and B. C. Moore, "Using recurrent neural networks to improve the perception of speech in non-stationary noise by people with cochlear implants," *The Journal of the Acoustical Society of America*, vol. 146, no. 1, pp. 705–718, 2019.
- [30] 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.
- [31] J. Wright and Y. Ma, *High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications*. Cambridge University Press, 2022.
- [32] W.-N. Hsu, Y. Zhang, and J. Glass, "Learning latent representations for speech generation and transformation," *Proc. Interspeech*, pp. 1273–1277, 2017.
- [33] —, "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.
- [34] —, "Unsupervised learning of disentangled and interpretable representations from sequential data," in *Proc. Advances in Neural Inform. Process. Syst.*, 2017, pp. 1876–1887.
- [35] Y. Xie, T. Arildsen, and Z.-H. 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.
- [36] 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.
- [37] 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 minimizing rate reduction," *arXiv preprint arXiv:2111.06636*, 2021.
- [38] 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.
- [39] T. Gerkmann, M. Krawczyk-Becker, and J. Le Roux, "Phase processing for single-channel speech enhancement: History and recent advances," *IEEE signal processing Magazine*, vol. 32, no. 2, pp. 55–66, 2015.
- [40] S.-W. Fu, Y. Tsao, X. Lu, and H. 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.
- [41] S.-W. Fu, T.-W. Wang, Y. Tsao, X. Lu, and H. 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.
- [42] D. Michelsanti, Z.-H. Tan, S.-X. Zhang, Y. Xu, M. Yu, D. Yu, and J. 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.
- [43] 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.
- [44] 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.
- [45] 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.
- [46] 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.
- [47] S. Leglaive, L. Girin, and R. 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.
- [48] 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.
- [49] 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.
- [50] D. P. Kingma and M. Welling, “Auto-encoding variational Bayes,” *arXiv preprint arXiv:1312.6114*, 2013.
- [51] 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.
- [52] 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.
- [53] 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.
- [54] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, “Deep learning for monaural speech separation,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 1562–1566.
- [55] 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.
- [56] 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.
- [57] Y. Xiang, J. L. Højvang, M. H. Rasmussen, and M. G. Christensen, “A deep representation learning speech enhancement method using β -vae,” *Accepted by Eurosispc (arXiv preprint: arXiv:2205.05581)*, 2022.
- [58] 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.
- [59] 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.
- [60] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, “Autoencoding beyond pixels using a learned similarity metric,” in *International conference on machine learning*. PMLR, 2016, pp. 1558–1566.
- [61] G. Parmar, D. Li, K. Lee, and Z. Tu, “Dual contradistinctive generative autoencoder,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 823–832.
- [62] H. Huang, R. He, Z. Sun, T. Tan *et al.*, “Introvae: Introspective variational autoencoders for photographic image synthesis,” *Proc. Advances in Neural Inform. Process. Syst.*, vol. 31, 2018.
- [63] S. Pascual, A. Bonafonte, and J. Serrà, “Segan: Speech enhancement generative adversarial network,” *Proc. Interspeech*, pp. 3642–3646, 2017.
- [64] D. Michelsanti and Z.-H. Tan, “Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification,” *Proc. Interspeech*, pp. 2008–2012, 2017.
- [65] E. A. Wan and A. T. Nelson, “Networks for speech enhancement,” *Handbook of neural networks for speech processing*. Artech House, Boston, USA, vol. 139, no. 1, p. 7, 1999.
- [66] F. Xie and D. Van Compernelle, “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*, vol. 2. IEEE, 1994, pp. II–53.
- [67] 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.
- [68] 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.
- [69] 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.
- [70] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, “Least squares generative adversarial networks,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2794–2802.
- [71] K. Kumar, R. Kumar, T. de Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. de Brébisson, Y. Bengio, and A. C. Courville, “Melgan: Generative adversarial networks for conditional waveform synthesis,” *Proc. Advances in Neural Inform. Process. Syst.*, vol. 32, 2019.
- [72] 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. 17022–17033, 2020.
- [73] 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.
- [74] V. Panayotov, G. Chen, D. Povey, and S. 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.
- [75] G. Hu and D. 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.
- [76] A. Varga and H. J. Steeneken, “Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems,” *Speech Commun.*, vol. 12, no. 3, pp. 247–251, 1993.
- [77] K. Cho, B. van Merriënboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using rnn encoder-decoder for statistical machine translation,” in *EMNLP*, 2014.
- [78] C. K. Reddy, E. Beyrami, J. Pool, R. Cutler, S. Srinivasan, and J. Gehrke, “A scalable noisy speech dataset and online subjective test framework,” *Proc. Interspeech*, pp. 1816–1820, 2019.
- [79] 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.
- [80] H. Dubey, V. Gopal, R. Cutler, S. Matuselych, S. Braun, E. S. Eskimez, M. Thakker, T. Yoshioka, H. Gamper, and R. Aichner, “Icassp 2022 deep noise suppression challenge,” in *ICASSP*, 2022.
- [81] D. P. Kingma and J. Ba, “ADAM: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [82] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, “Pytorch: An imperative style, high-performance deep learning library,” *Advances in neural information processing systems*, vol. 32, 2019.
- [83] 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.
- [84] 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.
- [85] 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.
- [86] C. K. Reddy, V. Gopal, and R. Cutler, “Dnsmos: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors,” in *ICASSP 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6493–6497.

- [87] —, “Dnsmos p.835: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors,” in *ICASSP 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022.
- [88] B. Naderi and R. Cutler, “Subjective evaluation of noise suppression algorithms in crowdsourcing,” in *INTERSPEECH*, 2021.
- [89] Y. Xia, S. Braun, C. K. Reddy, H. Dubey, R. Cutler, and I. 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.
- [90] 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.
- [91] 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.
- [92] 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.