INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Diffiner: A Versatile Diffusion-based Generative Refiner for Speech Enhancement

Ryosuke Sawata[†] *Naoki Murata*[‡] *Yuhta Takida*[‡] *Toshimitsu Uesaka*[‡]
Takashi Shibuya[‡] *Shusuke Takahashi*[†] *Yuki Mitsufuji*[‡]

[†]Sony Group Corporation, Tokyo, Japan
[‡]Sony Research, Tokyo, Japan

{Ryosuke.Sawata, Naoki.Murata, Yuta.Takida, Toshimitsu.Uesaka, Takashi.Tak.Shibuya, Shusuke.Takahashi, Yuhki.Mitsufuji}@sony.com

Abstract

Although deep neural network (DNN)-based speech enhancement (SE) methods outperform the previous non-DNN-based ones, they often degrade the perceptual quality of generated outputs. To tackle this problem, we introduce a DNN-based generative refiner, Diffiner, aiming to improve perceptual speech quality pre-processed by an SE method. We train a diffusion-based generative model by utilizing a dataset consisting of clean speech only. Then, our refiner effectively mixes clean parts newly generated via denoising diffusion restoration into the degraded and distorted parts caused by a preceding SE method, resulting in refined speech. Once our refiner is trained on a set of clean speech, it can be applied to various SE methods without additional training specialized for each SE module. Therefore, our refiner can be a versatile post-processing module w.r.t. SE methods and has high potential in terms of modularity. Experimental results show that our method improved perceptual speech quality regardless of the preceding SE methods used. Our code is available at https://github.com/sony/diffiner.

Index Terms: speech enhancement (SE), deep generative model, diffusion-based generative model

1. Introduction

In the field of speech enhancement (SE), DNN-based methods have drastically improved the performance of conventional ones in terms of signal-to-noise ratio (SNR) [1–3]. However, in some cases they tend to degrade qualities of speech such as naturalness and perceptual quality for human listening [4]. Because the inputs for downstream applications (e.g., ASR and telecommunication system) should ideally be clean and high-quality speech, it has been reported that the speech processed by the aforementioned DNN-based methods often degrade their performances [5–7].

There are two main approaches to solving this problem: a) a DNN learning strategy that aims to improve both SNR and perceptual speech quality, and b) optimizing a preceding SE model in terms of the downstream application's criterion. Regarding approach a), there are some studies that introduce criteria related to perceptual quality of human into loss function in order for the target DNN to deal with it [8–10]. For example, Fu *et al.* proposed a DNN-based SE method that can be optimized by utilizing an arbitrary metric related to the perceptual quality for human listening [10], e.g., perceptual evaluation of speech quality (PESQ) [11] and short-time objective intelligibility (STOI) [12], by using a framework of generative adversarial network (GAN). However, although PESQ and STOI are correlated with perceptual speech quality, optimizing the target SE model on the basis of these metrics does not always improve the actual quality perceived by humans because the mechanisms of PESQ and STOI are not perfectly equal to human listening [13]. Shi *et al.* and Liu *et al.* hypothesized that synthesizing conditioned speech would improve perceptual quality and proposed using a vocoder for the SE task to generate clean speech [14,15]. However, training the vocoder with noisy speech tends to be more laborious than in the case of the SE model only, and it often degrades the final perceptual quality. Meanwhile, in terms of approach b), some studies have attempted to optimize SE models so that a criterion of the following application is maximized. For instance, the joint training connects the DNNs of the SE and ASR models and trains the connected model as one DNN in terms of the ASR's criteria [16, 17]. However, because this approach requires training the SE model for each following model, the learning requires much effort. Therefore, a new scheme is desired that can improve the perceptual speech quality without the laborious data collection and training.

To remove the distortions from SE outputs, we focus on the utilization of deep generative models built on only clean speech. Although there are some SE methods using deep generative model built on both noisy and clean speeches [18–20], which they are same as the traditional DNN-based SE methods, a generative model should be built on the target domain, i.e., clean speeches in our case, and thus using noisy speeches may degrade its performance. To be more specific, we consider the task of SE as a generative task, where generative models are expected to detect degraded parts and refine them by mixing generated clean parts effectively. Recently, many types of generative models such as the GAN [21], variational auto-encoder (VAE) [22], flow-based models [23], and denoising diffusion-based models [24] have been proposed. More recently, denoising diffusion-based generative models in particular have been extensively studied [24–26]. Kawar *et al.* devised Denoising Diffusion Restoration Models (DDRM) as an effective way to use the pre-trained denoising diffusion-based generative model for general linear inverse problems [27]. DDRM can be applied to various tasks (e.g., image super-resolution, inpainting, and colorization) without any specialized retraining for each task.

Inspired by the successful adaptation of diffusion-based restoration models, we propose a diffusion-based generative refiner, Diffiner, for pre-processed speech. Specifically, we first train a diffusion-based generative model by utilizing a dataset consisting of clean speech only. Then, we effectively mix clean parts newly generated via the above DDRM-based framework into degraded and distorted ones caused by a preceding SE method, resulting in refinement. More specifically, our method calculates weights based on the output of a preceding SE model and does the above mixing by utilizing the weights. Consequently, Diffiner can boost the perceptual speech quality without no new additional noise datasets since Diffiner can be

10.21437/Interspeech.2023-1547

trained by using only clean speech. It is worth noting that even the same clean speech used to train the preceding SE model can be used for training Diffiner, i.e., without any collection of new speech and noise datasets. Furthermore, once our model is trained only on a set of clean speech, it can be applied to various SE methods without additional training specialized for each SE module. In summary, even though our model is built on only clean speech and does not require retraining the preceding SE model, the perceptual speech quality can be refined. Therefore, our method is versatile w.r.t. SE methods and has high potential in terms of modularity. In our experiments, we show that our method effectively improves speech quality regardless of the SE method used for pre-processing.

2. Preliminaries

In this section, we revisit diffusion-based generative models and a related diffusion-based method for general linear inverse problems.

2.1. Diffusion-based generative models

Denoising diffusion probabilistic models [24,28], which we refer to as diffusion-based generative models in this paper, are latent variable models with the latents $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(T)}$, where T is the number of time steps. The latents have the same dimensionality as the data $\boldsymbol{x}^{(0)} \sim p_{\text{data}}(\boldsymbol{x}^{(0)})$. Their joint distribution is defined as a Markov chain, and the transitions starting at $p(\boldsymbol{x}^{(T)})$, which is a Gaussian distribution, follow the Gaussian transition, as

$$p_\theta(\boldsymbol{x}^{(0:T)}) = p(\boldsymbol{x}^{(T)}) \prod_{t=1}^{T} p_\theta(\boldsymbol{x}^{(t-1)}|\boldsymbol{x}^{(t)}), \tag{1}$$

$$p_\theta(\boldsymbol{x}^{(t-1)}|\boldsymbol{x}^{(t)}) = \mathcal{N}(\boldsymbol{x}^{(t-1)}; \boldsymbol{\mu}_\theta^{(t)}(\boldsymbol{x}^{(t)}), \boldsymbol{\Sigma}_\theta^{(t)}(\boldsymbol{x}^{(t)})). \tag{2}$$

The parameter of the model θ is learned to generate the data distribution following such a Markov chain so that it is consistent with the following inference distribution q:

$$q(\boldsymbol{x}^{(1:T)}|\boldsymbol{x}^{(0)}) = \prod_{t=1}^{T} q(\boldsymbol{x}^{(t)}|\boldsymbol{x}^{(t-1)}),$$
$$q(\boldsymbol{x}^{(t)}|\boldsymbol{x}^{(t-1)}) = \mathcal{N}(\boldsymbol{x}^{(t)}; \sqrt{1-\beta_t}\boldsymbol{x}^{(t-1)}, \beta_t \boldsymbol{I}). \tag{3}$$

The former process is called the *reverse process*, whereas the latter is called the *forward process*. The frequently used parameterization of the mean $\boldsymbol{\mu}_\theta$ is

$$\boldsymbol{\mu}_\theta^{(t)}(\boldsymbol{x}^{(t)}) = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}^{(t)} - \frac{\beta_t}{\sqrt{1-\overline{\alpha}_t}}\boldsymbol{\epsilon}_\theta^{(t)}(\boldsymbol{x}^{(t)})\right), \tag{4}$$

where $\alpha_t := 1 - \beta_t$ and $\overline{\alpha}_t := \prod_{s=1}^{t} \alpha_s$. $\boldsymbol{\epsilon}_\theta^{(t)}$ is a function whose input and output sizes are the same as that of $\boldsymbol{\mu}_\theta$. $\boldsymbol{\Sigma}_\theta^{(t)}(\boldsymbol{x}^{(t)})$ is often set to be $\beta_t \boldsymbol{I}$ or $\tilde{\beta}_t \boldsymbol{I} = \frac{1-\overline{\alpha}_{t-1}}{1-\overline{\alpha}_t}\beta_t \boldsymbol{I}$. With this parameterization, the training objective is reduced to

$$\mathbb{E}_{\boldsymbol{x}^{(0)}, \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), t}\left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta^{(t)}(\sqrt{\overline{\alpha}_t}\boldsymbol{x}^{(0)} + \sqrt{1-\overline{\alpha}_t}\boldsymbol{\epsilon})\|_2^2\right]. \tag{5}$$

Note that with the parameterization of (4), the function $f_\theta^{(t)}(\boldsymbol{x}^{(t)}) := (\boldsymbol{x}^{(t)} - \sqrt{1-\overline{\alpha}_t}\boldsymbol{\epsilon}_\theta^{(t)}(\boldsymbol{x}^{(t)}))/\sqrt{\overline{\alpha}_t}$ can be viewed as a denoiser at each time step t, which can predict the clean signals $\boldsymbol{x}^{(0)}$ given $\boldsymbol{x}^{(t)}$ [24].

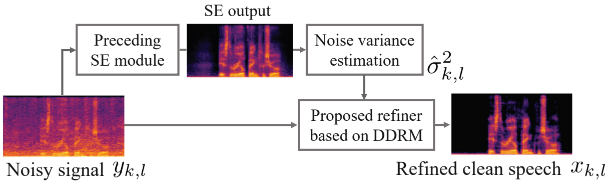

SE output
Preceding SE module
Noise variance estimation
$\hat{\sigma}^2_{k,l}$
Proposed refiner based on DDRM
Noisy signal $y_{k,l}$
Refined clean speech $x_{k,l}$

Figure 1: *Overview of proposed diffusion-based speech refiner*

2.2. Denoising diffusion restoration models

Kawar *et al.* proposed DDRM [27], which is an unsupervised method for solving general linear inverse problems. The goal of a linear inverse problem is to restore the signal $\boldsymbol{x} \in \mathbb{R}^n$ from the observation $\boldsymbol{y} \in \mathbb{R}^m$ obtained by the following linear equation:

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{z}, \tag{6}$$

where $\boldsymbol{z} \sim \mathcal{N}(0, \sigma_{\boldsymbol{y}}^2 \boldsymbol{I})$ is an *i.i.d.* additive Gaussian noise with known variance. Here, $\boldsymbol{H} \in \mathbb{R}^{m \times n}$ is a degradation linear operator and is assumed to be known.

For any linear inverse problem, the DDRM is defined as

$$p_\theta(\boldsymbol{x}^{(0:T)}|\boldsymbol{y}) = p_\theta^{(T)}(\boldsymbol{x}^{(T)}|\boldsymbol{y}) \prod_{t=0}^{T-1} p_\theta^{(t)}(\boldsymbol{x}^{(t)}|\boldsymbol{x}^{(t+1)}, \boldsymbol{y}), \tag{7}$$

where $\boldsymbol{x}^{(0)}$ is the estimate of $\boldsymbol{x}$ in this model. Given the singular value decomposition (SVD) of $\boldsymbol{H}$, DDRM can take how much information from $\boldsymbol{y}$ is available into account in the domain induced by the SVD (for denoising). For instance, components corresponding to larger singular values are less-noisy in the spectral space, and thus much information from the observation is used to restore the signal. Meanwhile, components corresponding to smaller singular values are hard to observe because of lower SNR, and thus much information from the generative model is used.

The point of DDRM is that once an unconditional diffusion-based generative model is trained on clean data, it can be utilized for various types of linear inverse problems because the knowledge of the linear operator is required only at an inference time, as discussed in the original paper [27].

3. Diffusion-based Speech Refiner

In this section, we propose our diffusion-based refiner for SE outputs. First, we train a diffusion-based generative model on clean speech data. After obtaining results from an arbitrary preceding SE module, the variance of the noise included in noisy input at each time-frequency bin is estimated. With the estimate, the proposed refiner generates clean speech on the basis of the DDRM framework, which utilizes the pre-trained diffusion-based model.

3.1. Diffusion-based generative model on clean speech data

We assume that the noisy time-domain signal y is decomposed as $y = x + n$, where x and n are the target signals (to be enhanced) and noise signals, respectively. Then we can obtain their short-time Fourier transformation (STFT) coefficients, denoted as $y_{k,l}, x_{k,l}$, and $n_{k,l} \in \mathbb{C}$, respectively. k and l denote the frequency bin and time frame indexes. Note that the existing diffusion-based models can be extended to complex-valued data, such as STFT coefficients [19,20].

First, a diffusion-based generative model for the STFT coefficients is trained on clean speech data, which gives the pre-trained diffusion-based generative model $x_{k,l} \sim p_\theta(x_{k,l})$.

Here, the forward process for the Markov chain is defined as the process that injects circular-symmetric complex Gaussian noise as follows,

$$q^{(t)}(x_{k,l}^{(t)} \mid x_{k,l}^{(0)}) = \mathcal{N}_{\mathbb{C}}(x_{k,l}^{(0)}, \sigma_t^2), \tag{8}$$

with different noise levels $0 = \sigma_0 < \sigma_1 < \cdots < \sigma_T$. As in general diffusion-based generative models, although this forward process that adds noise is performed independently for each time-frequency bin, the reverse process that removes noise is trained by considering correlations between the bins. Again, note that, with the trained model $p_\theta(x_{k,l})$, the denoiser $f_\theta^{(t)}(\cdot)$ that is defined in the same manner as Sec. 2.1 can predict the denoised signals, which is denoted here as $\overline{x}_{k,l}^{(t)}$, given noisy signals $x_{k,l}^{(t)}$.

3.2. Diffusion-based refiner for SE outputs

In this subsection, we introduce our DDRM-based refiner. After obtaining the clean speech estimation with the existing SE module, which we denote as $\hat{x}_{k,l}$, we can also compute the estimated noise signal $\hat{n}_{k,l}$ by subtracting $\hat{x}_{k,l}$ from $y_{k,l}$. Then we model the estimated variance of STFT coefficients $\hat{n}_{k,l}$, as

$$\hat{\sigma}_{k,l}^2 = \min(\max(\lambda|\hat{n}_{k,l}|^2, \delta), R) \tag{9}$$

where $\delta > 0$ and $R > 0$ are the minimum and maximum thresholds of the estimated variance for avoiding numerical instability, respectively. We leave $\lambda > 0$ as a tunable hyperparameter. This variance estimate leads to the following approximated linear inverse problem:

$$y_{k,l} = x_{k,l} + n_{k,l},\ n_{k,l} \sim \mathcal{N}_{\mathbb{C}}(0, \hat{\sigma}_{k,l}^2), \tag{10}$$

and it can be written in the form of a general linear inverse problem as in (6):

$$\tilde{y}_{k,l} = \frac{1}{\hat{\sigma}_{k,l}} x_{k,l} + z_{k,l}, z_{k,l} \sim \mathcal{N}_{\mathbb{C}}(0,1), \tag{11}$$

where $\tilde{y}_{k,l} = y_{k,l}/\hat{\sigma}_{k,l}$. Here, we assume $z_{k,l}$ follows an *i.i.d.* complex Gaussian distribution for tractability. Based on the inverse problem in (11), the proposed method is able to generate refined signals with DDRM, which is summarized in Algorithm 1. η_a, η_b, and $\eta_c \in [0,1]$ are tunable hyperparameters that control diversity of generated samples. For notational simplicity, $\tilde{y}_{k,l}$ is replaced by $y_{k,l}/\hat{\sigma}_{k,l}$ in the algorithm. We can interpret this to mean that the algorithm generates speech by appropriately combining the estimated clean speech $\overline{x}_{k,l}^{(t)}$ with the observed noisy speech $y_{k,l}$ in accordance with the estimated noise variance $\hat{\sigma}_{k,l}^2$ at each time-frequency bin.

In practical cases of SE, the estimated variance $\hat{\sigma}_{k,l}^2$ may include estimation errors and additive noise $z_{k,l}$ may have correlations among different time-frequency bins. Thus, the term $(y_{k,l} - \overline{x}_{k,l}^{(t)})$ in the update procedure, which uses the information from the observation, does not necessarily follow the independent complex Gaussian with the estimated variance, which is assumed in DDRM. To address this issue, we add a modification to the original DDRM framework as shown in Algorithm 1, which is referred to as **"Diffiner"**. We use the term $(x_{k,l}^{(t+1)} - \overline{x}_{k,l}^{(t)})$ instead of $(y_{k,l} - \overline{x}_{k,l}^{(t)})$ if $\sigma_t < \hat{\sigma}_{k,l}$ because the observed signal is no longer helpful in this condition. This change in the term means that the algorithm ignores the observational information and relies on the generative model during less noisy steps ($\sigma_t < \hat{\sigma}_{k,l}$).

Algorithm 1 Proposed diffusion-based refiner for SE outputs

```
Input: noisy input $y_{k,l}$, estimated noise variance $\hat{\sigma}_{k,l}^2$
Output: refined signal $x_{k,l}$
  initialize $x_{k,l}^{(T)} \sim \mathcal{N}_{\mathbb{C}}(0, \sigma_T^2 - \hat{\sigma}_{k,l}^2)$
  for $t = T - 1$ to 0 do
      Predict denoised signal $\overline{x}_{k,l}^{(t)}$ using $f_\theta^{(t)}(\cdot)$
      Sample $z_{k,l} \sim \mathcal{N}_{\mathbb{C}}(0,1)$
      if $\sigma_t < \hat{\sigma}_{k,l}$ then
          Pattern 1: Original DDRM update rule (Diffiner)
          $x_{k,l}^{(t)} = \overline{x}_{k,l}^{(t)} + \eta_a \sigma_t \frac{y_{k,l} - \overline{x}_{k,l}^{(t)}}{\hat{\sigma}_{k,l}} + \sqrt{1-\eta_a^2}\sigma_t z_{k,l}$
          Pattern 2: Modified update rule (Diffiner+)
          $x_{k,l}^{(t)} = \overline{x}_{k,l}^{(t)} + \eta_c \sigma_t \frac{x_{k,l}^{(t+1)} - \overline{x}_{k,l}^{(t)}}{\sigma_{t+1}} + \sqrt{1-\eta_c^2}\sigma_t z_{k,l}$
      else // $\sigma_t \geq \hat{\sigma}_{k,l}$
          $x_{k,l}^{(t)} = (1-\eta_b)\overline{x}_{k,l}^{(t)} + \eta_b y_{k,l} + \sqrt{\sigma_t^2 - \eta_b^2 \hat{\sigma}_{k,l}^2} z_{k,l}$
      end if
  end for
  Refined signals $x_{k,l} = x_{k,l}^{(0)}$
```

4. Experiments

4.1. Setup

4.1.1. Dataset

To train and evaluate our proposed model, we utilized an openly available dataset, Voice Bank Corpus (VBC) [29], consisting of only clean speech since our diffusion-based refiner does not require noisy or noise signals as we discussed in Sec. 1.

Meanwhile, to train and evaluate DNN-based SE models that are used before our refiner, we utilized VoiceBank-DEMAND (VBD) [30], which is also openly available and frequently used in DNN-based speech enhancement [18, 31]. The train and test sets consist of 28 and two speakers (11572 and 824 utterances), respectively.

4.1.2. Preceding SE methods

We used four SE methods as preceding modules to evaluate our refiner: Wiener filter [32], Deep complex U-net (DCUnet) [31], Wave-U-net [33], and speech enhancement GAN (SEGAN) [18]. Note that all of them except the Wiener filter were trained on the VBD dataset.

4.1.3. Proposed diffusion-based generative refiner

For the proposed refiner, we trained a diffusion-based generative model on STFT spectrograms of the clean speech obtained from the VoiceBank corpus. As parameters for STFT, the window size, hop size, and number of time frames were set to 512, 256, and 256, respectively. A Hann window was used as the analysis window. By truncating the direct current (DC) component, we treated the spectrograms as 256×256 tensors with two channels, which correspond to the real part and the imaginary part of a complex value. We modified the diffusion-based model with U-Net-based architecture[1] so that the number of input/output channels is two.

We trained the model on a single NVIDIA A100 GPU (40 GB memory) for 7.5×10^5 steps, which took about three days. We used the Adam optimizer [34] with a learning rate

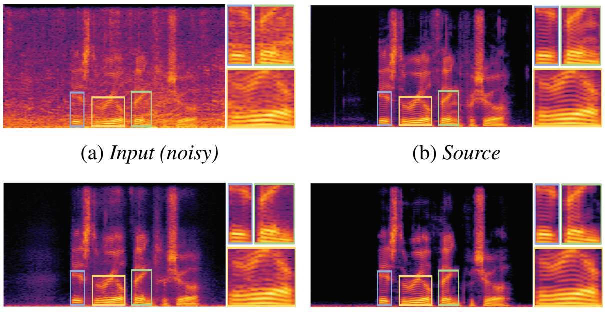
[1] https://github.com/openai/guided-diffusion

Table 1: *Experimental results. Please note that the details of our proposed Diffiner and Diffiner+ are written in Sec. 3.2 and Algorithm 1.*

Method	**Reference-free Metrics**		**Reference-based Metrics**		
	NISQA	**OVRL**	**SI-SDR**	**WB-PESQ**	**ESTOI**
Source	4.546	3.220	-	4.644	1.000
Input (noisy)	3.040	2.697	8.448	1.971	0.787
Wiener filter [32]	3.544	2.846	15.65	2.414	0.793
w/ Diffiner	4.472	3.064	**18.09**	**2.476**	**0.847**
w/ Diffiner+	**4.621**	**3.079**	16.52	2.387	0.820
DCUnet [31]	4.287	3.149	**20.16**	**2.981**	**0.886**
w/ Diffiner	4.752	3.183	19.80	2.970	0.885
w/ Diffiner+	**4.827**	**3.187**	19.27	2.810	0.861
Wave-U-net [33]	3.968	3.091	18.15	2.657	0.842
w/ Diffiner	4.663	3.159	**19.62**	**2.684**	**0.871**
w/ Diffiner+	**4.773**	**3.161**	18.33	2.608	0.848
SEGAN [18]	3.527	3.019	15.94	2.166	0.823
w/ Diffiner	4.372	3.154	**19.36**	**2.546**	**0.867**
w/ Diffiner+	**4.609**	**3.160**	17.79	2.513	0.851
UNIVERSE†[36]	4.606	3.109	10.10	2.901	0.838
SGMSE+‡[20]	4.565	3.178	17.42	2.903	0.864

† Implemented by ourselves because the code is not publicly available. The network was trained on VBD, and we only considered Mel bands for feature NLLs.

‡ We used the authors' code and a provided checkpoint, but some results were slightly different from their paper's evaluation scores. This is because the predictor-corrector samplers used in SGMSE+ are stochastic.



(a) *Input (noisy)*

(b) *Source*

(c) *Pre-processed by DCUnet*

(d) *Refined by Diffiner+*

Figure 2: *Spectrograms of noisy input, clean target, DCUnet, and refined by diffusion-based generative refiner.*

of 0.001 and batch size of 8. Following [35], an exponential moving average of the model weight was taken with a decay of 0.9999 to be used for inference.

For the inference, we first conducted a grid search for the parameters required in both of the deep generative "Diffiner" and "Diffiner+": η_a, η_b, and η_c in Algorithm 1. Specifically, we searched the parameter space $[0.0, 0.2, 0.4, 0.6, 0.8, 1.0]$ for η_a, η_b, and η_c, respectively. Furthermore, λ, δ, and R were set to $\lambda = 1.0$, $\delta = 1.0^{-5}$, and $R = \sigma^2_{T-1} \simeq 97$, respectively. Then, we ran our refiners with $T = 200$.

4.2. Results

To evaluate the performance, we used 5 metrics[2]: SI-SDR, WB-PESQ, ESTOI, non-intrusive speech quality assessment (NISQA), and the overall (OVRL) metric of the deep noise challenge mean opinion score (DNSMOS) P.835 [11, 37–40]. The first 3 metrics, which require the pair of a target signal and the corresponding reference source, have been recently used for performance evaluation on VBD [20, 41]. However, we focus on improving the last 2 metrics, i.e., NISQA and OVRL, in our experiments. This is because some methods for speech enhancement, including our refiner, are based on generative models, whose outputs contain generated parts that do not completely match the corresponding references. In contrast with the aforementioned traditional reference-based metrics, NISQA and OVRL can predict the mean opinion score (MOS) of a target signal without the corresponding reference source. Thus, we considered that they are suitable for the evaluation of the generative model-based results.

The evaluation results are summarized in Table 1. Note that we added some diffusion-based state-of-the-art methods, universal speech enhancement with score-based diffusion (UNIVERSE) [36] and the improved version of score-based generative model for SE (SGMSE+) [20], as references. As shown in the table, all reference-free scores, i.e., NISQA and OVRL, were always improved by applying our refiners, effectively. Therefore, in terms of human-like MOS scores, both Diffiner and Diffiner+ succeeded to improve the SE results regardless of what kind of preceding SE method was used. In particular, the results obtained by applying Diffiner+ are comparable to the source in NISQA and OVRL. Namely, in terms of reference-free metrics related to MOS, Diffiner+ could improve all preceding SE methods, achieving as natural speech as the source. Furthermore, all NISQA and OVRL scores by Diffiner+ are comparable to or higher than those of UNIVERSE and SGMSE+, which are state-of-the-art diffusion-based SE methods. In contrast, our refiners reduced the reference-based metrics marginally. As we discussed in the previous paragraph, this is because the refined results contain generated parts that do not match the reference, and thus it could spoil the reference-based scores. However, as shown in the table, Diffiner+ could keep its scores comparable with that of the source while improving NISQA and OVRL. Thus, although it is a generative model, Diffiner+ can refine the target signals and keep them close to the corresponding references.

An example of the refined results is shown in Fig. 2. Even though the details of the refined parts were not always consistent with the corresponding parts of the source spectrogram (see Figs. 2(b) and (d)), our refiner could generate natural-looking parts and succeeded to mix them into the distorted parts resulting in a high-quality spectrogram (see Figs. 2(c) and (d)).

5. Conclusion

We presented a DNN-based generative refiner to improve speech that has already been pre-processed by a preceding SE method. We devised "Diffiner", an extension of DDRM, that is more suitable for the task of speech enhancement. After our model is trained on a set of clean speech, it can be used as a versatile speech refiner for results processed by preceding SE methods without additional training specialized for each method. Experimental results showed that our method effectively improved the speech quality in terms of NISQA and OVRL regardless of the SE method used as pre-processing. In future work, we will explore the feasibility of our deep generative refiner for a scaled dataset larger than VBC and its application to other types of sound such as music and environmental sounds. Moreover, our refiner can be integrated with various DNN-based SE methods, resulting in a joint generative refiner. Especially, there may be an effective way to integrate our deep generative refiner with other diffusion-based SE methods [20, 36, 42–44] into a single unified model.

[2] To compute these metrics, we utilized the latest versions of the following public codes as of March 2023:
·`https://github.com/sigsep/sigsep-mus-eval`
·`https://github.com/gabrielmittag/NISQA`
·`https://github.com/microsoft/DNS-Challenge`

6. References

[1] Y. Luo and N. Mesgarani, “Conv-TasNet: Surpassing ideal time-frequency magnitude masking for speech separation,” *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 27, no. 8, pp. 1256–1266, 2019.

[2] X. Hao, X. Su, R. Horaud, and X. Li, “Fullsubnet: A full-band and sub-band fusion model for real-time single-channel speech enhancement,” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6633–6637.

[3] H.-S. Choi, S. Park, J. H. Lee, H. Heo, D. Jeon, and K. Lee, “Real-time denoising and dereverberation with tiny recurrent U-net,” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 5789–5793.

[4] A. R. Avila, M. J. Alam, D. O’Shaughnessy, and T. Falk, “Investigating Speech Enhancement and Perceptual Quality for Speech Emotion Recognition,” in *Proc. of Interspeech*, 2018, pp. 3663–3667.

[5] Y.-H. Tu, J. Du, L. Sun, F. Ma, and C.-H. Lee, “On Design of Robust Deep Models for CHiME-4 Multi-Channel Speech Recognition with Multiple Configurations of Array Microphones,” in *Proc. of Interspeech*, 2017, pp. 394–398.

[6] S.-J. Chen, A. S. Subramanian, H. Xu, and S. Watanabe, “Building State-of-the-art Distant Speech Recognition Using the CHiME-4 Challenge with a Setup of Speech Enhancement Baseline,” in *Proc. of Interspeech*, 2018, pp. 1571–1575.

[7] M. Fujimoto and H. Kawai, “One-Pass Single-Channel Noisy Speech Recognition Using a Combination of Noisy and Enhanced Features,” in *Proc. of Interspeech*, 2019, pp. 486–490.

[8] S. Kataria, J. Villalba, and N. Dehak, “Perceptual loss based speech denoising with an ensemble of audio pattern recognition and self-supervised models,” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 7118–7122.

[9] T.-A. Hsieh, C. Yu, S.-W. Fu, X. Lu, and Y. Tsao, “Improving perceptual quality by phone-fortified perceptual loss using wasserstein distance for speech enhancement,” in *Proc. of Interspeech 2021*, 2021, pp. 196–200.

[10] S.-W. Fu, C.-F. Liao, Y. Tsao, and S.-D. Lin, “MetricGAN: Generative adversarial networks based black-box metric scores optimization for speech enhancement,” in *Proc. of the 36th International Conference on Machine Learning (ICML)*, vol. 97, 09–15 Jun 2019, pp. 2031–2041.

[11] A. Rix, J. Beerends, M. Hollier, and A. Hekstra, “Perceptual evaluation of speech quality (PESQ) - A new method for speech quality assessment of telephone networks and codecs,” in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, 2001, pp. 749–752.

[12] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, “A short-time objective intelligibility measure for time-frequency weighted noisy speech,” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2010, pp. 4214–4217.

[13] Y. Koizumi, K. Niwa, Y. Hioka, K. Kobayashi, and Y. Haneda, “Dnn-based source enhancement to increase objective sound quality assessment score,” *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1780–1792, 2018.

[14] J. Shi, X. Chang, T. Hayashi, Y.-J. Lu, S. Watanabe, and B. Xu, “Discretization and re-synthesis: an alternative method to solve the cocktail party problem,” *arXiv preprint arXiv:2112.09382*, 2021.

[15] H. Liu, X. Liu, Q. Kong, Q. Tian, Y. Zhao, D. Wang, C. Huang, and Y. Wang, “VoiceFixer: A Unified Framework for High-Fidelity Speech Restoration,” in *Proc. of Interspeech*, 2022, pp. 4232–4236.

[16] Z.-Q. Wang and D. Wang, “Joint training of speech separation, filterbank and acoustic model for robust automatic speech recognition,” in *Proc. of Interspeech*, 2015, pp. 2839–2843.

[17] T. Menne, R. Schlüter, and H. Ney, “Investigation into joint optimization of single channel speech enhancement and acoustic modeling for robust ASR,” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6660–6664.

[18] S. Pascual, A. Bonafonte, and J. Serrà, “SEGAN: Speech Enhancement Generative Adversarial Network,” in *Proc. of Interspeech*, 2017, pp. 3642–3646.

[19] S. Welker, J. Richter, and T. Gerkmann, “Speech Enhancement with Score-Based Generative Models in the Complex STFT Domain,” in *Proc. of Interspeech*, 2022, pp. 2928–2932.

[20] J. Richter, S. Welker, J.-M. Lemercier, B. Lay, and T. Gerkmann, “Speech enhancement and dereverberation with diffusion-based generative models,” *arXiv preprint arXiv:2208.05830*, 2022.

[21] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in *Proc. of Advances in Neural Information Processing Systems (NeurIPS)*, 2014, pp. 2672–2680.

[22] D. P. Kingma and M. Welling, “Auto-encoding variational Bayes,” in *Proc. of International Conference on Learning Representation (ICLR)*, 2014.

[23] D. Rezende and S. Mohamed, “Variational inference with normalizing flows,” in *Proc. of International Conference on Machine Learning (ICML)*. PMLR, 2015, pp. 1530–1538.

[24] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” in *Proc. of the 34th International Conference on Neural Information Processing Systems*, 2020, pp. 6840–6851.

[25] J. Song, C. Meng, and S. Ermon, “Denoising diffusion implicit models,” in *International Conference on Learning Representations*, 2020.

[26] P. Dhariwal and A. Nichol, “Diffusion models beat GANs on image synthesis,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 8780–8794, 2021.

[27] B. Kawar, M. Elad, S. Ermon, and J. Song, “Denoising diffusion restoration models,” in *Proc. of Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[28] A. Q. Nichol and P. Dhariwal, “Improved denoising diffusion probabilistic models,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 8162–8171.

[29] C. Veaux, J. Yamagishi, and S. King, “The voice bank corpus: Design, collection and data analysis of a large regional accent speech database,” in *Proc. of International Conference Oriental COCOSDA held jointly with Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE)*, 2013, pp. 1–4.

[30] C. Valentini-Botinhao, X. Wang, S. Takaki, and J. Yamagishi, “Investigating RNN-based speech enhancement methods for noise-robust text-to-speech,” in *Proc. of International Speech Communication Association (ISCA) Speech Synthesis Workshop*, 2016, pp. 146–152.

[31] 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 *Proc. of International Conference on Learning Representations (ICLR)*, 2019.

[32] N. Wiener, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series: With Engineering Applications*. The MIT Press, 08 1949. [Online]. Available: https://doi.org/10.7551/mitpress/2946.001.0001

[33] D. Stoller, S. Ewert, and S. Dixon, “Wave-U-Net: A Multi-Scale Neural Network for End- to-End Audio Source Separation,” in *Proc. of the 19th International Society for Music Information Retrieval (ISMIR) Conference* , 2018, pp. 334–340.

[34] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. of 3rd International Conference on Learning Representations (ICLR)*, Y. Bengio and Y. LeCun, Eds., 2015.

[35] Y. Song and S. Ermon, “Improved techniques for training score-based generative models,” in *Proc. of the 34th International Conference on Neural Information Processing Systems*, 2020, pp. 12 438–12 448.

[36] J. Serrà, S. Pascual, J. Pons, R. O. Araz, and D. Scaini, “Universal speech enhancement with score-based diffusion,” *arXiv preprint arXiv:2206.03065*, 2022.

[37] J. L. Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, “SDR – half-baked or well done?” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 626–630.

[38] J. Jensen and C. H. Taal, “An algorithm for predicting the intelligibility of speech masked by modulated noise maskers,” *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 2009–2022, 2016.

[39] A. R. Avila, H. Gamper, C. Reddy, R. Cutler, I. Tashev, and J. Gehrke, “Non-intrusive speech quality assessment using neural networks,” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 631–635.

[40] C. K. A. Reddy, V. Gopal, and R. Cutler, “DNSMOS P.835: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors,” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 886–890.

[41] R. Scheibler, Y. Ji, S.-W. Chung, J. Byun, S. Choe, and M.-S. Choi, “Diffusion-based generative speech source separation,” *arXiv*, vol. abs/2210.17327, 2022.

[42] Y.-J. Lu, Y. Tsao, and S. Watanabe, “A study on speech enhancement based on diffusion probabilistic model,” in *Proc. of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2021, pp. 659–666.

[43] J. Zhang, S. Jayasuriya, and V. Berisha, “Restoring Degraded Speech via a Modified Diffusion Model,” in *Proc. of Interspeech*, 2021, pp. 221–225.

[44] Y.-J. Lu, Z.-Q. Wang, S. Watanabe, A. Richard, C. Yu, and Y. Tsao, “Conditional diffusion probabilistic model for speech enhancement,” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 7402–7406.