

GDIFFUSE: DIFFUSION-BASED SPEECH ENHANCEMENT WITH NOISE MODEL GUIDANCE

Efrayim Yanir, David Burshtein

Tel-Aviv University
efrayimyanir@mail.tau.ac.il
burstyn@tauex.tau.ac.il

Sharon Gannot

Bar-Ilan University
Sharon.Gannot@biu.ac.il

ABSTRACT

This paper introduces a novel speech enhancement (SE) approach based on a denoising diffusion probabilistic model (DDPM), termed Guided diffusion for speech enhancement (GDiffuSE). In contrast to conventional methods that directly map noisy speech to clean speech, our method employs a lightweight helper model to estimate the noise distribution, which is then incorporated into the diffusion denoising process via a guidance mechanism. This design improves robustness by enabling seamless adaptation to unseen noise types and by leveraging large-scale DDPMs originally trained for speech generation in the context of SE. We evaluate our approach on noisy signals obtained by adding noise samples from the BBC sound effects database to LibriSpeech utterances, showing consistent improvements over state-of-the-art baselines under mismatched noise conditions. Examples are available at: <https://ephiephi.github.io/GDiffuSE-examples.github.io>

Index Terms— Generative models, Diffusion processes, DDPM Guidance

1. INTRODUCTION

Dominant approaches for SE utilize discriminative models that map noisy inputs to clean targets [1]. These models perform well under matched conditions but generalize poorly to unseen noise or acoustic environments, often introducing artifacts. Generative models that learn an explicit prior over clean speech have gained popularity in recent years, particularly in the context of SE.

Diffusion-based generative models [2, 3] gradually add Gaussian noise in a forward process and learn a network to reverse it by iterative denoising. Unlike variational autoencoders (VAEs), they have no separate encoder—the “latent” at step t is the noisy sample itself—and the network learns the score (gradient of log-density) across noise levels [4]. They have exhibited promising results in audio generation. For instance, DiffWave achieves high-fidelity audio generation with a small number of parameters [5]. Recent works adapt diffusion models to SE [6, 7, 8]. Two main designs have emerged. (i) A *conditioner vocoder* pipeline, where a diffusion vocoder resynthesizes speech utilizing features predicted from the noisy input, with auxiliary losses pushing those features toward clean targets [7, 9]. These methods require an auxiliary loss and use two separate models for generation and denoising. (ii) *Corruption-aware diffusion* that integrates the corruption model into the forward chain so its reversal directly yields the enhanced signal via linear interpolation between clean and noisy waveforms, e.g., CDiffuSE [10], or by embedding noise statistics in an stochastic differential equation (SDE) drift [6]. The latter design better reflects

real-world, non-white noise [11]. A recent contribution to the field is the Score-based Generative Modeling for Speech Enhancement (SGMSE) family of algorithms [6, 12, 13], which learns a score function that enables sampling from the posterior distribution of clean speech given the noisy observation in the complex short-time Fourier transform (STFT) domain. All of these methods demonstrate that a conditioned diffusion generator can achieve state-of-the-art performance across diverse noise conditions. However, they all require specialized training of the heavy diffusion model for each type of expected noise.

In this paper, we introduce GDiffuSE, a diffusion probabilistic approach to SE. GDiffuSE uses the guidance mechanism [14] by a lightweight noise model, which guides the signal generated by the DiffWave [7] model towards the estimated clean speech. A key benefit of GDiffuSE is that, given a new unknown noise, only the compact noise model has to be trained, which is substantially easier than learning the full distribution of noisy speech. As a result, the system rapidly adapts to unseen acoustic conditions with few noise samples, provided that the noise statistics has not significantly changed between train and inference time..

Our main contributions are threefold: (1) We derive a novel approach for using DDPM guidance for SE by applying guidance directly into a noise-distribution model for SE. (2) We propose a novel reverse process that leverages a foundation diffusion model for SE, offering robust adaptability to unseen noise types—assuming the noise statistics remain consistent between the available noise-only utterance and the noise encountered at inference. (3) The experimental results confirm the effectiveness of GDiffuSE, achieving improved robustness to mismatched noise conditions compared to related generative SE methods.

2. PROBLEM FORMULATION

Let $y_i = x_{0,i} + w_i$ denote the noisy signal received by a single microphone, where $x_{0,i}$ is the clean speech component and w_i is the noise component, for $i \in \{0, \dots, N-1\}$, and N the number of samples in the utterance. Stacking the N samples into column vectors yields $\mathbf{x}_0 \triangleq (x_{0,i})_{i=0}^{N-1}$, $\mathbf{w} \triangleq (w_i)_{i=0}^{N-1}$, $\mathbf{y} \triangleq (y_i)_{i=0}^{N-1}$, leading to the following vector form:

$$\mathbf{y} = \mathbf{x}_0 + \mathbf{w}. \quad (1)$$

Given \mathbf{y} , the goal of the SE algorithm is to estimate $\hat{\mathbf{x}} \triangleq (\hat{x}_i)_{i=0}^{N-1}$ that is perceptually and/or objectively close to \mathbf{x}_0 .

3. PROPOSED METHOD

In this section, we derive the proposed SE algorithm. Sec. 3.1 presents the use of DDPM guidance for SE, and Sec. 3.2 describes the training of the noise model that guides the DDPM. The complete process is illustrated in Fig. 1.

3.1. DDPM Guidance for Speech Enhancement

DDPM [3] uses a diffusion process [2] for generative sampling. DDPM guidance [14] modifies the standard generative sampling procedure of DDPM to a conditional one as summarized in [14, Algorithm 1]. We suggest adopting this approach for SE in a new way, using guidance from the noise model distribution, as summarized in Algorithm 2.

We follow the notations in [3, 14]. The data distribution of the clean speech is given by $\mathbf{x}_0 \sim q(\mathbf{x}_0)$. In the forward diffusion process, a Markov chain progressively adds noise to \mathbf{x}_0 to produce $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T$ as follows:

$$\mathbf{x}_t = \sqrt{1 - \beta_t} \mathbf{x}_{t-1} + \beta_t \mathbf{e}_t, \mathbf{e}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \mathbf{e}_t \perp \mathbf{x}_{t-1}, \quad (2)$$

where \mathbf{e}_t (Gaussian distributed with zero mean, and identity covariance matrix) is statistically independent of \mathbf{x}_{t-1} , and $\beta_t \in [\beta_{\text{start}}, \beta_{\text{end}}]$ is a schedule parameter. Other schedule parameters, α_t and $\bar{\alpha}_t$ are defined in [3, 14] in the following way:

$$\alpha_t = 1 - \beta_t, \quad \bar{\alpha}_t = \prod_{s=1}^t \alpha_s = \prod_{s=1}^t (1 - \beta_s). \quad (3)$$

Consequently, the t -step marginal is [3]:

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \hat{\mathbf{e}}_t, \hat{\mathbf{e}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \hat{\mathbf{e}}_t \perp \mathbf{x}_0. \quad (4)$$

Denoising is performed by recursively applying the following reverse process, for $t = T, T-1, \dots, 1$:

$$p_{\theta}(\mathbf{x}_{t-1} | \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}(\mathbf{x}_t, t), \sigma_t^2 \mathbf{I}). \quad (5)$$

Since the distribution of the reverse process is intractable, it is modeled by a Deep Neural Network (DNN), where θ represents the set of trainable parameters of the denoising network. Therefore, sampling can be expressed with:

$$\mathbf{x}_{t-1} = \boldsymbol{\mu}(\mathbf{x}_t, t) + \sigma_t \mathbf{z}_t, \mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \mathbf{z}_t \perp \mathbf{x}_t, \quad (6)$$

where the mean $\boldsymbol{\mu}(\mathbf{x}_t, t)$ can be expressed using the standard noise-prediction form

$$\boldsymbol{\mu}(\mathbf{x}_t, t) = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \boldsymbol{\epsilon}_{\theta}(\mathbf{x}_t, t) \right). \quad (7)$$

The function $\boldsymbol{\epsilon}_{\theta}(\mathbf{x}_t, t)$ is the network's estimate of the injected noise [3, Algorithm 1]. As shown in [3] and [5], for accelerating the computation it is useful to use in (6):

$$\sigma_t^2 = \tilde{\beta}_t = \begin{cases} \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t} \beta_t & \text{for } t > 1 \\ \beta_1 & \text{for } t = 1 \end{cases}. \quad (8)$$

For an SE problem, we want to add a guidance component to guide the diffusion process towards the clean speech \mathbf{y} . For that, we train the diffusion model in the standard way, but then we wish to sample \mathbf{x}_0 from the conditional probability density function (p.d.f.)

$p_{\phi}(\mathbf{x}_0 | \mathbf{y})$, modeled by a DNN ϕ . This can be done as described in [2, 14]. Rather than using (6)-(7) we use:

$$\mathbf{x}_{t-1} = \boldsymbol{\mu}_t^{\text{guid}} + \sigma_t \tilde{\mathbf{e}}_t, \quad \tilde{\mathbf{e}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \quad (9)$$

where

$$\boldsymbol{\mu}_t^{\text{guid}} = \boldsymbol{\mu}(\mathbf{x}_t, t) + s_t \frac{\beta_t}{\sqrt{\alpha_t}} \nabla_{\mathbf{x}} \log p_{\phi}(\mathbf{y} | \mathbf{x})|_{\mathbf{x}=\boldsymbol{\mu}(\mathbf{x}_t, t)}. \quad (10)$$

We set the gradient scale, s_t , according to the schedule:

$$s_t = \lambda_{\max} \left(\frac{\sqrt{1 - \bar{\alpha}_t}}{\sqrt{1 - \bar{\alpha}_1}} \right)^{\gamma}, \quad \gamma > 0, \lambda_{\max} > 0. \quad (11)$$

Intuitively, this schedule yields weak guidance when the state is very noisy and stronger guidance when the effective signal-to-noise ratio (SNR) rises and the guidance is more reliable. This choice is consistent with standard SNR-dependent scheduling for diffusion models [15], and aligns with recent evidence that guidance strength should vary with the noise level rather than remain constant [16, 17].

Now, given observation \mathbf{y} we can use (9)-(10) to estimate the clean speech. We just need to know $\nabla_{\mathbf{x}_t} \log p_{\phi}(\mathbf{y} | \mathbf{x}_t)$.

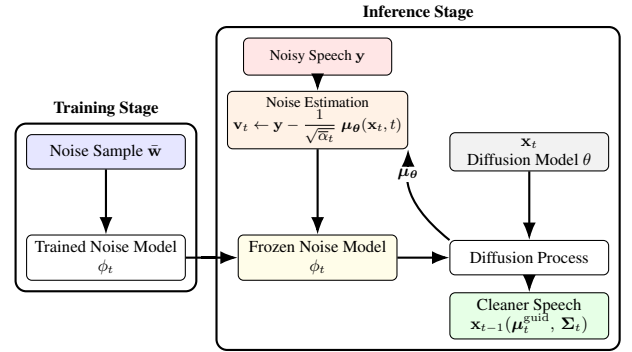


Fig. 1: GDiffuSE: The trained noise model guides the diffusion model for SE. **Training stage:** Noise sample $\bar{\mathbf{w}} \in \mathbb{R}^N$ trains the noise models ϕ_t for each t . **Inference stage:** Starting from \mathbf{x}_t (white noise for $t = T$), the diffusion process, guided by the loss from ϕ_t (19), generates \mathbf{x}_{t-1} ; the clean estimate is \mathbf{x}_0 . The input to ϕ_t is the noise estimate (which uses \mathbf{y}). This is repeated T times (See Algorithms 1, 2).

3.2. Noise Model Training

In this section, we specify how to train the noise model, ϕ . The conditional density $p_{\phi}(\mathbf{y} | \mathbf{x}_t)$ is inferred using the noise at the t -th guided diffusion step and the additive (acoustic) noise, as follows. Combining (4) with (1) yields,

$$\mathbf{y} = \mathbf{x}_0 + \mathbf{w} = \frac{1}{\sqrt{\bar{\alpha}_t}} \mathbf{x}_t - \sqrt{\frac{1 - \bar{\alpha}_t}{\bar{\alpha}_t}} \hat{\mathbf{e}}_t + \mathbf{w}. \quad (12)$$

Denote the combined noise:

$$\mathbf{v}_t \triangleq -\frac{\sqrt{1 - \bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}} \hat{\mathbf{e}}_t + \mathbf{w} = \mathbf{w} - g(t) \hat{\mathbf{e}}_t \quad (13)$$

where

$$g(t) = \sqrt{\frac{1 - \bar{\alpha}_t}{\bar{\alpha}_t}}. \quad (14)$$

The first component is the diffusion noise, and the second is the acoustic noise that should be suppressed. Consequently, the conditional probability of the measurements given the desired speech estimate at the t -th step is given by

$$p_\phi(\mathbf{y} | \mathbf{x}_t) = p_\phi^{\mathbf{V}_t | \mathbf{x}_t}(\mathbf{y} - \frac{1}{\sqrt{\alpha_t}} \mathbf{x}_t | \mathbf{x}_t). \quad (15)$$

Hence, the required conditional probability density simplifies to the conditional density of the random variable \mathbf{V}_t given the variable \mathbf{x}_t , $p_\phi^{\mathbf{V}_t | \mathbf{x}_t}(\mathbf{v}_t | \mathbf{x}_t)$. Obviously, the additive noise, \mathbf{w} , is statistically independent of \mathbf{x}_t . To further simplify the derivation, we also make the assumption that $\hat{\mathbf{e}}_t$ is independent of \mathbf{x}_t . Consequently, the density of \mathbf{V}_t given \mathbf{x}_t becomes the density of $\mathbf{w} - g(t) \cdot \hat{\mathbf{e}}_t$, where $\hat{\mathbf{e}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is independent of \mathbf{w} . We also assume the availability of a noise sample $\bar{\mathbf{w}}$ from the same distribution as \mathbf{w} , which can be used to train a model for \mathbf{v}_t . In practice, a voice activity detector (VAD) can be used to allocate such segments from the given noisy utterance. Given a segment $\bar{\mathbf{w}}$, for each diffusion step t we can compute $g(t)$, the noise level for a specific step (14), and generate noise \mathbf{v}_t with the required density:

$$v_{t,i} = \bar{w}_i - \hat{e}_t \cdot g(t), \quad \hat{e}_t \sim \mathcal{N}(0, 1). \quad (16)$$

For inferring $p_\phi^{\mathbf{V}_t}(\mathbf{v}_t)$ we apply maximum likelihood (ML). The log likelihood is given by:

$$\log P(v_0, \dots, v_{N-1} | \theta) = \sum_{i=0}^{N-1} \log p(v_i | v_0, \dots, v_{i-1}, \theta), \quad (17)$$

and therefore, we need the conditional distribution of $v_{t,i} | (v_{t,0}, \dots, v_{t,i-1})$. We model it by a Gaussian: $v_{t,i} | (v_{t,0}, \dots, v_{t,i-1}) \sim \mathcal{N}(\cdot, \mu_{t,i}, \sigma_{t,i}^2)$. The noise is modeled separately for each t with shifted causal convolutional neural networks (CNNs) [18] to predict the mean and the variance:

$$\mu_{t,i}(v_{t,0}, \dots, v_{t,i-1}), \sigma_{t,i}^2(v_{t,0}, \dots, v_{t,i-1}) = \phi_t(v_{t,0}, \dots, v_{t,i-1}) \quad (18)$$

and ϕ_t is trained using the ML loss $(-\log L)_t$:

$$\text{loss}_t(\mathbf{v}_t) = - \sum_{i=0}^{N-1} \left[-\log \left(\sqrt{2\pi} \cdot \sigma_{t,i} \right) - \frac{v_{t,i} - \mu_{t,i}}{2 \cdot \sigma_{t,i}^2} \right]. \quad (19)$$

The training of the noise model a given noise sample $\bar{\mathbf{w}}$, is summarized in Algorithm 1. The guided reverse diffusion is summarized in Algorithm 2. The training and inference procedures are schematically depicted in Fig. 1. It is important to note that the backbone diffusion model is trained solely on clean speech, so large amounts of noisy data are not required. In practice, we employ a *pre-trained* diffusion model for clean speech (see Sec. 4.1), and only the lightweight noise model needs to be trained in the proposed scheme.

4. EXPERIMENTAL STUDY

In this section, we provide the implementation details of the proposed method, describe the competing method, the datasets used for training and testing, and evaluate the method's performance.

4.1. Implementation details

The noise model architecture is a CNN with 4 causal convolutional layers and linear heads for $\mu_{t,i}$ and $\sigma_{t,i}$, featuring residual connections and weight normalization. We use a WaveNet-style

Algorithm 1 Noise Model Training

Require: noise sample $\bar{\mathbf{w}} \in \mathbb{R}^N$, diffusion steps T , # epochs E , step size η , schedule $g(t)$.

- 1: **for** $t \leftarrow T$ **down to** 1 **do**
- 2: Compute $g(t)$, $\hat{\mathbf{e}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
- 3: $\mathbf{v}_t \leftarrow \bar{\mathbf{w}} - \hat{\mathbf{e}}_t g(t)$ ▷ elementwise: $v_{t,i} = \bar{w}_i - \hat{e}_t g(t)$
- 4: **for** $k \leftarrow 1$ **to** E **do** ▷ NumEpochs
- 5: $(\mu_{t,i}, \sigma_{t,i}^2)_{i=0}^{N-1} \leftarrow \phi_t(v_{t,0}, \dots, v_{t,i-1})$
- 6: $\text{loss}_t(\mathbf{v}_t) \leftarrow \text{See (19)}$
- 7: $\phi_t \leftarrow \text{ADAMSTEP}(\phi_t, \nabla_{\phi_t} \text{loss}_t, \eta)$
- 8: **end for**
- 9: **end for**
- 10: **return** $\{\phi_t\}_{t=1}^T$

Algorithm 2 Guided reverse diffusion (sampling)

Require: schedules $\{\alpha_t, \bar{\alpha}_t, \tilde{\beta}_t\}$; denoiser ϵ_θ ; noise models $\{\phi_t\}$; scheduled scales $\{s_t\}$; observation \mathbf{y}

- 1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
- 2: **for** $t \leftarrow T$ **down to** 1 **do**
- 3: $\mu_\theta(\mathbf{x}_t, t) \leftarrow \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(\mathbf{x}_t, t) \right)$
- 4: $\sigma_\theta^2(\mathbf{x}_t, t) \leftarrow \tilde{\beta}_t$
- 5: $\mathbf{v}_t \leftarrow \mathbf{y} - \frac{1}{\sqrt{\alpha_t}} \mu_\theta(\mathbf{x}_t, t)$
- 6: $\{\mu_{t,i}, \sigma_{t,i}^2\}_{i=0}^{N-1} \leftarrow \phi_t(v_{t,0}, \dots, v_{t,i-1})$
- 7: $\text{loss}_t(\mathbf{v}_t) \leftarrow \text{See (19)}$
- 8: $\mu_t^{\text{guid}} \leftarrow \mu_\theta(\mathbf{x}_t, t) + s_t \left(\frac{\beta_t}{\sqrt{\alpha_t}} \right) \left(-\frac{1}{\sqrt{\bar{\alpha}_t}} \frac{\partial \text{loss}_t(\mathbf{v}_t)}{\partial \mathbf{v}_t} \right)$
- 9: $\Sigma_t \leftarrow \text{diag}(\sigma_\theta^2(\mathbf{x}_t, t))$
- 10: $\mathbf{x}_{t-1} \sim \mathcal{N}(\mu_t^{\text{guid}}, \Sigma_t)$
- 11: **end for**
- 12: **return** \mathbf{x}_0

tanh-sigmoid gate, $\text{Gate}(h, g) = \tanh(h) \odot \text{sigm}(g)$, with $h = \text{Conv}_{\text{causal}}(x)$ and $g = \text{Conv}_{1 \times 1}(h)$. The network's parameters are a kernel size of 9, 2 channels, and dilations of [1, 2, 4, 8]. The parameters λ_{max} and γ in (11) exhibit a wide range of values with good results, spanning between [0.5, 1] for both. We calibrated them on one clip per SNR level to $\gamma = 0.7$ and $\lambda_{\text{max}} = [0.8, 0.72, 0.6, 0.55]$ for SNR levels [10, 5, 0, -5] dB, respectively. For the generator, we used the unconditional DDPM model, trained by UnDiff [19] with 200 diffusion steps, on the Datasets VCTK [20] and LJ-Speech [21].

4.2. Baseline method

We used SGMSE [12], a speech denoising model, which is a fully generative SOTA method. This model was trained on clean speech from either the WSJ0 Dataset [22] or the TIMIT dataset [23], and noise signals from the CHiME3 Dataset [24].

4.3. Datasets

As the backbone diffusion model is pre-trained (with clean speech), we only need noise clips for training the noise model and noisy signals (clean speech plus noise) for inference. We used LibriSpeech [25] (out-of-domain) as clean speech. For the noise, we selected real clips from the BBC sound effects dataset [26]. This lesser-known corpus was chosen because it includes noise types that

are rarely found in widely used datasets such as CHIME3, thereby enabling a more rigorous evaluation of robustness.

For the test set, we selected 20 speakers, each contributing one 5-second clean sample resampled to 16 kHz. The noise data consisted of 25-second clips, with 20 clips used for training the noise model and 5 seconds for testing. Noisy utterances were generated by mixing the 5-second clean speech with noise at various SNR levels.

4.4. Evaluation metrics

To assess the performance of the proposed GDiffuSE algorithm and compare it with the baseline method we used the following metrics: STOI [27], PESQ [28], SI-SDR [29] (all intrusive metrics that require a clean reference), and DNSMOS [30] (a non-intrusive, reference-free measure).

4.5. Experimental results

Results for real noise signals from the BBC sound effects dataset are shown in Table 1. Our method consistently outperforms SGMSE in PESQ and SI-SDR across all SNR levels, even if the gains are modest. Although SGMSE achieves higher STOI and DNSMOS scores, informal listening tests confirm that our approach delivers noticeably better perceptual sound quality.

To further assess robustness, we selected 20 noise clips with spectral profiles emphasizing higher frequencies. Since the noise statistics remain relatively stable over time, these clips align well with our model assumptions. As shown in Table 2, the performance gains of GDiffuSE over SGMSE become even more pronounced in this setting.¹

The spectrogram comparison in Fig. 2 highlights this difference: while SGMSE struggles to suppress the unseen noise, GDiffuSE adapts effectively to these challenging conditions. Audio examples² further confirm the superiority of the proposed method, particularly for unfamiliar noise types, where improvements in PESQ and SI-SDR are most evident.

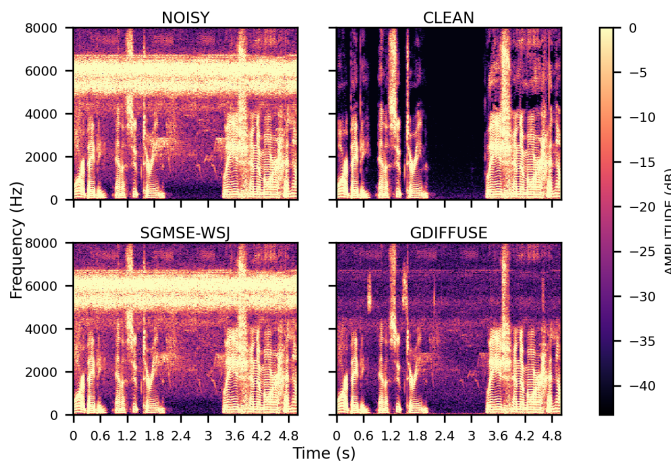


Fig. 2: Spectrograms assessment for sample NHU05093027 (monsoon forest) drawn from the BBC sound effect dataset.

¹In a future study, we will aim to comprehensively characterize the noise types for which GDiffuSE achieves the most significant gains.

²Available at <https://ephiephi.github.io/GDiffuSE-examples.github.io>

Table 1: Objective evaluation of the GDiffuSE algorithm using noise drawn from BBC sound effect dataset (higher is better).

SNR	Method	STOI	PESQ	DNSMOS	SI-SDR
10	GDiffuSE	0.91 ± 0.05	1.60 ± 0.36	2.92 ± 0.24	14.80 ± 3.55
	sgmseW	0.94 ± 0.04	1.59 ± 0.34	3.06 ± 0.27	14.23 ± 3.07
	sgmseT	0.93 ± 0.04	1.46 ± 0.27	3.04 ± 0.25	12.41 ± 1.77
	Input	0.90 ± 0.06	1.20 ± 0.14	2.42 ± 0.41	10.00 ± 0.02
5	GDiffuSE	0.86 ± 0.08	1.40 ± 0.32	2.73 ± 0.32	10.91 ± 4.47
	sgmseW	0.90 ± 0.06	1.34 ± 0.30	2.94 ± 0.27	10.46 ± 4.03
	sgmseT	0.88 ± 0.07	1.20 ± 0.16	2.78 ± 0.27	7.80 ± 2.65
	Input	0.84 ± 0.09	1.11 ± 0.09	2.03 ± 0.46	5.01 ± 0.03
0	GDiffuSE	0.78 ± 0.11	1.25 ± 0.27	2.65 ± 0.33	6.66 ± 5.52
	sgmseW	0.84 ± 0.10	1.18 ± 0.17	2.79 ± 0.34	6.04 ± 4.68
	sgmseT	0.82 ± 0.10	1.11 ± 0.09	2.61 ± 0.31	3.38 ± 3.53
	Input	0.77 ± 0.11	1.07 ± 0.06	2.41 ± 1.05	0.02 ± 0.04
-5	GDiffuSE	0.69 ± 0.15	1.12 ± 0.15	2.26 ± 0.61	1.34 ± 6.42
	sgmseW	0.76 ± 0.14	1.09 ± 0.10	2.51 ± 0.39	0.77 ± 5.52
	sgmseT	0.74 ± 0.14	1.07 ± 0.06	2.35 ± 0.36	-1.46 ± 4.24
	Input	0.69 ± 0.13	1.09 ± 0.17	2.04 ± 1.03	-4.97 ± 0.07

Table 2: Evaluation on 20 samples with spectral profile emphasizing high frequencies at SNR=5 dB.

Method	STOI	PESQ	DNSMOS	SI-SDR
GDiffuSE	0.88 ± 0.07	1.39 ± 0.24	2.87 ± 0.25	11.25 ± 3.21
sgmseWSJ0	0.91 ± 0.07	1.26 ± 0.17	2.82 ± 0.25	9.43 ± 2.64
sgmseTIMIT	0.89 ± 0.07	1.20 ± 0.14	2.84 ± 0.29	8.64 ± 2.85
Input	0.85 ± 0.09	1.07 ± 0.03	1.98 ± 0.47	5.00 ± 0.03

5. CONCLUSIONS

In this work, we introduced GDiffuSE, a lightweight SE method that employs a guidance mechanism to leverage foundation diffusion models without retraining the large backbone. By modeling the noise distribution—an easier task than mapping noisy to clean speech—our approach requires only a short reference noise clip, assuming stable noise statistics between training and inference, thereby improving robustness to unfamiliar noise types. On a dataset unseen during SGMSE training, our method surpasses the state-of-the-art SGMSE, as demonstrated by our experimental study and our project webpage.

6. REFERENCES

- [1] D. Wang and J. Chen, “Supervised speech separation based on deep learning: An overview,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1702–1726, 2018.
- [2] J. Sohl-Dickstein, E. A. Weiss, N. Maheswaranathan, and S. Ganguli, “Deep unsupervised learning using nonequilibrium thermodynamics,” in *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, 2015.
- [3] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.
- [4] Y. Song, J. Sohl-Dickstein, D. P. Kingma, P. Abbeel, P. Dhariwal, and N. B. Chen, “Score-based generative modeling through stochastic differential equations,” *International Conference on Learning Representations (ICLR)*, 2021.
- [5] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, “Dif-fwave: A versatile diffusion model for audio synthesis,” in *In-*

ternational Conference on Learning Representations (ICLR), 2021.

- [6] C. Welker and W. Kellermann, “Score-based generative speech enhancement in the complex spectrogram domain,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 7317–7321.
- [7] J. Serrà, J. Pons, P. d. Benito, S. Pascual, and A. Bonafonte, “Universal speech enhancement with score-based diffusion models,” *arXiv preprint arXiv:2208.05055*, 2022.
- [8] Y.-J. Lu, Y. Tsao, and S. Watanabe, “A study on speech enhancement based on diffusion probabilistic model,” in *APSIPA Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2021.
- [9] Y. Koizumi, K. Yatabe, S. Saito, and M. Delcroix, “SpecGrad: Diffusion-based speech denoising with noisy spectrogram guidance,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 8967–8971.
- [10] X. Lu, S. Zhang, K. J. Sim, S. Narayanan, and Z. Li, “Conditional diffusion probabilistic model for end-to-end speech enhancement,” in *Proceedings of the Annual Conference of the International Speech Communication Association (INTER-SPEECH)*, 2022, pp. 1–5.
- [11] P. Vincent, “A connection between score matching and denoising autoencoders,” *Neural Computation*, vol. 23, no. 7, pp. 1661–1674, 2011.
- [12] J. Richter, S. Welker, J.-M. Lemercier, B. Lay, and T. Gerkmann, “Speech enhancement and dereverberation with diffusion-based generative models,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 2351–2364, 2023.
- [13] J.-M. Lemercier, J. Richter, S. Welker, and T. Gerkmann, “Storm: A diffusion-based stochastic regeneration model for speech enhancement and dereverberation,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 2724–2737, 2023.
- [14] P. Dhariwal and A. Nichol, “Diffusion models beat GANs on image synthesis,” *Advances in neural information processing systems*, vol. 34, pp. 8780–8794, 2021.
- [15] T. Karras, M. Aittala, T. Aila, and S. Laine, “Elucidating the design space of diffusion-based generative models,” *Advances in neural information processing systems*, vol. 35, pp. 26 565–26 577, 2022.
- [16] X. Wang, N. Dufour, N. Andreou, M.-P. Cani, V. F. Abrevaya, D. Picard, and V. Kalogeiton, “Analysis of classifier-free guidance weight schedulers,” *arXiv preprint arXiv:2404.13040*, 2024.
- [17] T. Kynkäänniemi, M. Aittala, T. Karras, S. Laine, T. Aila, and J. Lehtinen, “Applying guidance in a limited interval improves sample and distribution quality in diffusion models,” *Advances in Neural Information Processing Systems*, vol. 37, pp. 122 458–122 483, 2024.
- [18] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, “Wavenet: A generative model for raw audio,” *arXiv preprint arXiv:1609.03499*, 2016.
- [19] A. Iashchenko, P. Andreev, I. Shchekotov, N. Babaev, and D. Vetrov, “Undiff: Unsupervised voice restoration with unconditional diffusion model,” in *Proc. Interspeech*, 2023.
- [20] J. Yamagishi, C. Veaux, and K. MacDonald, “CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit (version 0.92),” <https://doi.org/10.7488/ds/2645>, 2019.
- [21] K. Ito and L. Johnson, “The LJ speech dataset,” <https://keithito.com/LJ-Speech-Dataset/>, 2017.
- [22] J. S. Garofolo, D. Graff, D. Paul, and D. Pallett, “CSR-I (WSJ0) Complete,” <https://catalog.ldc.upenn.edu/LDC93S6A>, Linguistic Data Consortium, Philadelphia, 1993.
- [23] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren, and V. Zue, “TIMIT acoustic-phonetic continuous speech corpus,” <https://catalog.ldc.upenn.edu/LDC93S1>, Linguistic Data Consortium, Philadelphia, 1993.
- [24] J. Barker *et al.*, “The third CHiME speech separation and recognition challenge: Dataset, task and baselines,” in *Proc. IEEE ASRU*, 2015, pp. 504–511.
- [25] V. Panayotov *et al.*, “Librispeech: An ASR corpus based on public domain audio books,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.
- [26] “BBC sound effects archive,” <https://sound-effects.bbcrewind.co.uk/>, 2025.
- [27] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, “An algorithm for intelligibility prediction of time–frequency weighted noisy speech,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.
- [28] 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 International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2001, pp. 749–752.
- [29] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, “SDR – half-baked or well done?” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019.
- [30] C. K. A. Reddy, V. G. Tarunathan, H. Dubey, and *et al.*, “DNSMOS: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2021, pp. 6493–6497.