DIFFUSION MODEL-BASED MIMO SPEECH DENOISING AND DEREVERBERATION

Rino Kimura[1,2*], *Tomohiro Nakatani*[1], *Naoyuki Kamo*[1], *Delcroix Marc*[1], *Shoko Araki*[1], *Tetsuya Ueda*[2], *Shoji Makino*[2]

[1] NTT Corporation, Japan [2] Waseda University, Japan

ABSTRACT

This paper presents an extension of a diffusion model-based single microphone Speech Enhancement (SE) method, known as the Score-Based Generative Model for SE (SGMSE), to a Multi-Input Multi-Output (MIMO) SE. The extended method is called a multi-stream SGMSE (mSGMSE). MIMO SE's goal is to estimate multi-microphone clean speech signals with spatial cues from noisy, reverberant speech signals captured by a distant microphone array. mSGMSE models the conditional distribution of clean speech signals given the captured signals for multi-microphone signals using a diffusion model and generates clean speech estimates by using the reverse diffusion process. We also propose techniques to make mSGMSE computationally efficient and adaptable to various or unknown array geometries for general MIMO SE scenarios. Experiments show that mSGMSE outperforms SGMSE (when separately applied to each microphone signal) in terms of signal quality, spatial cues, and computation times. mSGMSE also significantly improves the automatic speech recognition performance when applied to the REVERB challenge real dataset, which has substantial mismatches including array geometries from the training dataset. Finally, we confirm that Weighted Prediction Error dereverberation (WPE) preprocessing can further enhance mSGMSE more than SGMSE.

***Index Terms*—** Diffusion model, denoising, dereverberation, multi-input multi-output, microphone array

1. INTRODUCTION

This paper proposes a new Multi-Input Multi-Output (MIMO) Speech Enhancement (SE) method based on the diffusion model [1–3]. Speech signals captured by distant microphones are often corrupted by noise and reverberation, degrading the perceptual quality and localization of speech and reducing the accuracy of such speech-related applications as Automatic Speech Recognition (ASR) and speaker location estimation. MIMO SE overcomes this problem by removing noise and reverberation from multi-microphone inputs. For MIMO SE, it is important not only to estimate the clean speech signal at each microphone but also to recover the inter-microphone relationships of the clean speech, known as spatial cues. These spatial cues include Inter-microphone Time and Level Differences (ITD and ILD), which are typical features of localization and speaker location estimation.

We can achieve MIMO SE based on signal processing and neural network (NN) approaches. Signal processing approaches extend conventional beamformers (BFs) for MIMO BF, such as the extension of the Minimum Variance Distortionless Response BF [4] or the multichannel Wiener filter [5]. Weighted Prediction Error dereverberation (WPE) can also perform MIMO SE [6]. However, these methods have limited SE performance when the number of microphones is small. The NN approaches propose MIMO SE techniques based on discriminative NN [7–9]. Although these techniques are powerful, they may introduce unpleasant artifacts and be sensitive to the mismatched conditions between training and testing.

A diffusion model-based approach was recently introduced for a Single-Input Single-Output (SISO) SE, which enhances a speech signal from a single microphone signal [10–13]. Diffusion models are promising because they can perform SE with high perceptual quality [12]. They also seem more robust to the mismatched conditions between training and testing than discriminative NNs [3]. The Score-based Generative Model for Speech Enhancement (SGMSE), which is a diffusion model approach [3], models the conditional distribution of the clean speech given the captured signal using a diffusion model and estimates the clean speech by solving the reverse diffusion process. SGMSE uses a neural network, called a score model, to approximate the scores that are needed to perform the reverse process. SGMSE was shown to be effective for denoising and dereverberation by simulation experiments [3].

However, SGMSE is designed only for SISO SE, not for MIMO SE. One could perform MIMO SE by individually applying SGMSE to each microphone signal. But this method does not exploit spatial cues, and cannot estimate them accurately. Also, the computational cost would increase as the number of microphones increases.

To overcome the above limitations of SGMSE, this paper extends it and proposes a multi-stream SGMSE (mSGMSE) that can perform MIMO SE in a computationally efficient way. mSGMSE models the conditional distribution of multi-microphone signals and can naturally capture the spatial cues of the signals. Moreover, we propose a score model for mSGMSE that can compute scores in a computationally efficient way. Regardless of the number of microphones, mSGMSE's computational cost for handling all the microphones is almost the same as that of SGMSE to handle a single microphone. Furthermore, we introduce Multiple Array-Geometry (MAG) training of the score model to make mSGMSE can cope with various and unknown array geometries at inference.

We experimentally compare the performances of mSGMSE and SGMSE for MIMO SE tasks with noisy reverberant speech signals. SGMSE is individually applied to each microphone signal, whereas mSGMSE is collectively applied to the entire microphone array. We use both simulated data with various array geometries and real data with unknown array geometries and demonstrate that mSGMSE can effectively perform MIMO SE for both data types. In particular, it improves the estimation accuracy of the spatial cues for the simulated data and the ASR performance for the real data significantly more than SGMSE. Moreover, when we combine both methods with MIMO WPE preprocessing [6], mSGMSE substantially outperforms SGMSE in every metric, including the signal distortion metrics, while the performance of both methods is greatly enhanced.

2. MIMO SE FORMURATION

We consider a scenario where a microphone array captures a noisy, reverberant speech signal in an unknown environment. Under a general recording condition, we assume that the array geometry is not

*This work was done partly during an internship at NTT Corporation.

given. We formulate MIMO SE as a problem of estimating clean speech $\boldsymbol{x}_0 \in \mathbb{C}^{F\times N\times M}$ from the complex spectrum of captured signal $\boldsymbol{y} \in \mathbb{C}^{F\times N\times M}$, where F, N, and M denote the numbers of frequencies, time frames, and microphones. In this paper, we define clean speech $\boldsymbol{x}_0$ as the direct signal component included in captured signal $\boldsymbol{y}$.

3. CONVENTIONAL DIFFUSION MODEL-BASED SISO SE

This paper adopts SGMSE [3] as a baseline SISO SE. We overview it in this section (see [3] for more details) and extend it to MIMO SE in the next section. We tentatively set $M = 1$ for the number of microphones here for explaining SGMSE.

3.1. SGMSE for SISO SE

SGMSE uses a diffusion model [2] conditioned on captured signal $\boldsymbol{y}$ to perform a SISO SE. The model is characterized by a forward process that transforms clean speech $\boldsymbol{x}_0$ to a mixture of captured speech $\boldsymbol{y}$ and complex White Gaussian Noise (cWGN). We assume here that $\boldsymbol{x}_0$ follows a certain initial distribution conditioned by $\boldsymbol{y}$, i.e., $p(\boldsymbol{x}_0|\boldsymbol{y})$. Then speech enhancement is achieved based on the reverse process of the forward process that oppositely transforms captured speech plus cWGN back to clean speech that follows initial distribution $p(\boldsymbol{x}_0|\boldsymbol{y})$. SGMSE respectively uses the forward and reverse processes for training and inference.

The forward process is defined using a stochastic differential equation (SDE) based on the Ornstein-Uhlenbeck Process [14]:

$$\mathrm{d}\boldsymbol{x}_t = \gamma \underbrace{(\boldsymbol{y} - \boldsymbol{x}_t)}_{\mathbf{f}(\boldsymbol{x}_t,\boldsymbol{y})}\mathrm{d}t + \underbrace{\sqrt{c}k^t}_{g(t)}\mathrm{d}\boldsymbol{w}. \tag{1}$$

Here $\boldsymbol{x}_t$ is the state of the process indexed by $t \in [0, T]$, $\mathbf{f}$ and g are the drift and diffusion coefficient functions, and $\boldsymbol{w}$ is a standard Wiener process. γ (> 0) is a stiffness parameter, and c and k (> 0) are noise scheduling parameters. Based on the above SDE, $\boldsymbol{x}_t$ moves from a clean speech $\boldsymbol{x}_0$ towards $\boldsymbol{y}$ plus cWGN with an increased variance governed by the noise scheduling parameters [2].

SGMSE achieves SISO SE by solving the reverse SDE that performs the reverse process of Eq. (1). It is derived as [15]:

$$\mathrm{d}\boldsymbol{x}_t = [-\gamma\mathbf{f}(\boldsymbol{x}_t, \boldsymbol{y}) + g(t)^2\nabla_{\boldsymbol{x}_t} p_t(\boldsymbol{x}_t|\boldsymbol{y})]\mathrm{d}t + g(t)\mathrm{d}\bar{\boldsymbol{w}}. \tag{2}$$

Here $\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t|\boldsymbol{y})$ is the gradient of $\log p_t(\boldsymbol{x}_t|\boldsymbol{y})$ with respect to $\boldsymbol{x}_t$, called a score, and $\bar{\boldsymbol{w}}$ is a standard Wiener process in reverse time. To perform SISO SE for a given captured signal $\boldsymbol{y}$, the reverse process first obtains $\boldsymbol{x}_T$ by adding a sampled cWGN to $\boldsymbol{y}$. Then it obtains clean speech estimate $\hat{\boldsymbol{x}}_0$ by iteratively solving the reverse SDE from $t = T$ to 0.

Because score $\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t|\boldsymbol{y})$ in the reverse SDE is not readily available, SGMSE approximates it by a pre-trained neural network $\boldsymbol{s}_\theta(\boldsymbol{x}_t, \boldsymbol{y}, t)$, called a score model. The training objective to determine model parameters θ is derived based on the Mean Square Error (MSE) criterion and the perturbation kernel of the forward process [2, 16, 17]:

$$\mathcal{L}(\theta) = E_{t,(\boldsymbol{x}_0,\boldsymbol{y}),\boldsymbol{z}}\left\|\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, \boldsymbol{y}, t) + \frac{\boldsymbol{z}}{\sigma(t)^2}\right\|_2^2, \tag{3}$$

where $\sigma(t)^2 = c(k^{2t} - e^{-2\gamma t})/2(\gamma + \log(k))$ is the variance of the perturbation kernel at t and $\boldsymbol{z} \in \mathbb{C}^{F\times N\times M}$ is a cWGN sampled with a mean zero and an identity covariance matrix. $E_{t,(\boldsymbol{x}_0,\boldsymbol{y}),\boldsymbol{z}}$ denotes the expectation over t, $(\boldsymbol{x}_0, \boldsymbol{y}) \sim p(\boldsymbol{x}_0, \boldsymbol{y})$, and $\boldsymbol{z}$.

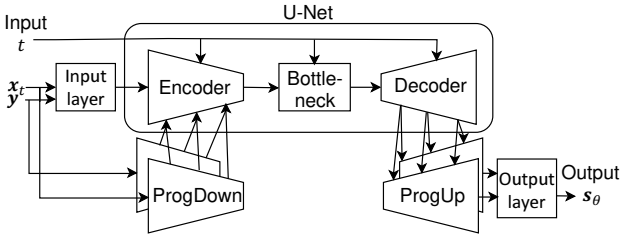

Fig. 1: NCSN++: score model for SISO SE with SGMSE.

3.2. Structure of score model for SGMSE

This subsection briefly describes structure of score model $\boldsymbol{s}_\theta(\boldsymbol{x}_t, \boldsymbol{y}, t)$ used for SGMSE, which will be later extended for MIMO SE. The structure, shown in Fig. 1, is called a Noise Conditional Score Network (NCSN++) [2]. It receives $\boldsymbol{x}_t$, $\boldsymbol{y} \in \mathbb{C}^{F\times N}$, and t as inputs and estimates a score as output. To deal with complex signals, SGMSE handles their real and imaginary parts as separate real signals.

In NCSN++, a U-Net block (U-Net) [18] plays the primary role in estimating the score. It consists of CNN layers, which are grouped into encoder, bottleneck, and decoder blocks. The encoder receives a number of feature maps extracted from input signals $\boldsymbol{x}_t$ and $\boldsymbol{y}$ by the input layer. Each map has a height and width that match the frequency and frame sizes of the input signals. The encoder then gradually transforms the maps by reducing the map size. At each intermediate layer, the encoder also receives down-sampled signals from a Progressive Down-Sampling block (ProgDown) and adds them to the maps. The decoder then gradually transforms the maps by recovering the map size. At each intermediate layer, the decoder also sends the maps to a Progressive Up-Sampling block (ProgUp).

In contrast, ProgDown guides and stabilizes the training of the U-Net encoder. It down-samples input signals $\boldsymbol{x}_t$ and $\boldsymbol{y}$ individually and step-by-step on both the frequency and frame axes to fit the feature map size of each U-Net encoder intermediate layer. The down-sampled signals are then fed to each layer of the U-Net encoder after being converted to fit the number of maps of the layer. ProgUp receives feature maps from each U-Net decoder intermediate layer after converting them to have the same number of maps as that of input signals. Then, ProgUp up-samples the maps in each channel individually and mixes them step-by-step with other maps received from other layers of the U-Net decoder. Finally, the output layer extracts the score from the maps received from ProgUp. Note that the down-sampling and up-sampling are pre-fixed functions and are not updated during training.

U-Net is by far the largest block in NCSN++, and the majority of the computational resources required for training and inference are concentrated on the U-Net part.

4. PROPOSED DIFFUSION MODEL-BASED MIMO SE

This section extends SGMSE so that it can perform MIMO SE in a computationally efficient way. We call the extended model multi-stream SGMSE (mSGMSE). Hereafter, we assume the number of microphones to be $M > 1$.

mSGMSE's diffusion processes are basically the same as those of SGMSE except that mSGMSE models multi-microphone signals. Specifically, we set $M > 1$ for the number of microphones of $\boldsymbol{x}_t$, $\boldsymbol{y}$, $\boldsymbol{z}$, and $\boldsymbol{s}_\theta$ in Eqs. (1) to (3) for the forward and reverse SDEs and the training objective of the score model. With this extension, multi-microphone clean speech $\boldsymbol{x}_0$ is estimated by the reverse process following joint distribution $p(\boldsymbol{x}_0|\boldsymbol{y})$ defined across the M microphones. Now $p(\boldsymbol{x}_0|\boldsymbol{y})$ models the relationships across all the microphones. Thus, mSGMSE can perform MIMO SE with accurate

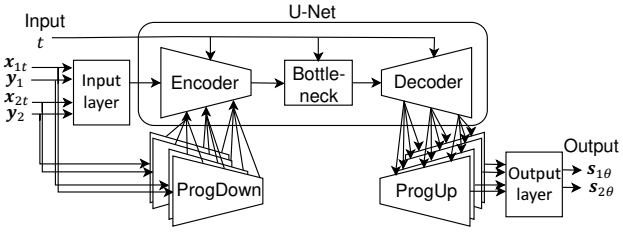

Fig. 2: mNCSN++: score model for MIMO SE with mSGMSE. $\boldsymbol{x}_{mt} \in \mathbb{C}^{F\times N}$ is complex spectrum corresponding to the mth microphone. The same applies to $\boldsymbol{s}_{m\theta}$ and $\boldsymbol{y}_m$.

estimation of the signal's spatial cues, if the score model can reliably approximate the score for multi-microphone signals.

4.1. Structure of score model for mSGMSE

There are several alternatives to extend the score model to estimate the scores of multi-microphone signals. For example, following the idea behind MIMO TasNet [8], we can use M NCSN++ blocks to process M microphone signals and let the blocks interact to capture the signal's spatial cues. However, such an approach significantly increases the computational cost as the number of microphones increases.

In this paper, we propose a multi-microphone extension of NCSN++, i.e., mNCSN++, shown in Fig. 2. In order to achieve minimal increase in the computational cost for handling the multi-microphone signals, mNCSN++ adopts the same U-Net as that of NCSN++, assuming that extension is not necessary for this part. Instead, we modify the other parts in NCSN++ to handle multi-microphone signals and scores:

1. The input layer extracts feature maps that match the size of the U-Net input from the multi-microphone input signals.
2. ProgDown down-samples the multi-microphone input signals individually and step-by-step and sends them to each intermediate layer of the U-Net encoder after converting them to fit the number of feature maps of the layer.
3. ProgUp receives feature maps from the U-Net decoder after converting them to have the number of maps as that of multi-microphone input signals and up-samples and mixes each channel of the maps individually and step-by-step.
4. The output layer extracts scores for the multi-microphone signals from the maps received from ProgUp.

In the above extension, the U-Net part, which requires the majority of computations, is unchanged from NCSN++ even when the number of microphones is increased. Thus, the increase in the computational cost can be minimal. Specifically, the computational cost of mSGSME for processing M microphone signals is comparable with that of SGMSE for processing a single microphone signal.

4.2. Multiple Array-Geometry (MAG) training

One important concern of mSGMSE is whether we can make it applicable to signals captured by various microphone arrays with different array geometries. If mSGMSE must use fixed array geometry for training and testing, its applicability will be severely limited. Note that SGMSE does not have this concern because it performs SISO SE individually to each microphone signal.

As a premise for mSGMSE to handle various array geometries, we assume that distribution $p(\boldsymbol{x}_0|\boldsymbol{y})$ of the multi-microphone signals can more reliably determine $\boldsymbol{x}_0$ given $\boldsymbol{y}$ than separately modeling the distribution for each microphone, even when the array geometry is not fixed. Under this premise, we prepare a training dataset that contains signals from various array geometries and use it to train the score model. We call this teaching method Multiple Array-Geometry (MAG) training of the score model.

5. EXPERIMENTS

We experimentally compared mSGMSE and SGMSE for the MIMO SE tasks that jointly performed denoising and dereverberation. For the MIMO tasks, SGMSE was applied individually to each microphone signal. We used two types of data recorded under A) matched conditions and B) mismatched conditions to evaluate their robustness against the mismatches between training and testing. We also investigated the reduction of the computational cost for the MIMO SE by mSGMSE.

5.1. Datasets under matched and mismatched conditions

We created a dataset, named WSJ0-CHiME3, for training the SE models and performing the matched condition evaluation. We simulated each captured signal by mixing a speech signal taken from the Wall Street Journal (WSJ0) dataset [19] and 10 noise signals from the CHiME3 dataset [20] after convolving each signal with a Room Impulse Response (RIR). We also simulated a clean speech target using the same RIR truncated at 2 ms. We used a linear array with $M = 2$ microphones. For the MAG training, the microphone spacing was randomly selected from 2, 4, 6, 8, 10, 12, and 14 cm. We generated the RIRs using the image method. The room size was set to $5 \times 5 \times 2$ m. The speaker, the array, and the noise sources were randomly positioned for each utterance; the speaker-array distance was constrained between 0.5 and 1.5 m. The reverberated speech to the noise ratio and the reverberation time (T60) varied from 10 to 14 dB and from 0.2 to 1.0 s. The training, validation, and evaluation sets contained 7138, 5000, and 333 utterances. Note that the evaluation set also included various microphone spacings (i.e., array geometries).

For the mismatched condition evaluation, we used the simulated and real evaluation data (SIMU and REAL) from the REVERB challenge dataset (REVERB) [21]. SIMU was generated using the measured RIRs, and REAL consisted of actual recordings. Both SIMU and REAL included noisy and reverberant speech signals captured by circular arrays with eight microphones under six and two different conditions, respectively. We used the first two microphones for our evaluation. The array geometries of SIMU and REAL were unknown to mSGMSE trained on WSJ0-CHiME3. We used clean speech signals in SIMU as the clean SIMU targets.

5.2. Evaluation metrics for signal quality, spatial cue, and computation time

We evaluated the two SE models in terms of the a) signal quality, b) spatial cues, and c) Real-Time Factors (RTFs).

For the signal quality metrics, we used the Signal-to-Distortion Ratio (SDR) [22], the Scale-Invariant SDR (SI-SDR) [23], the frequency weighted segmental Signal-to-Noise Ratio (fwsSNR) [24], the Perceptual Evaluation of Speech Quality (PESQ) [24], the Extended Short-Time Objective Intelligibility (ESTOI) [25], and the Word Error Rate (WER). For SDR, we set the length of the time-invariant filters to 512 taps. For WER, we used a REVERB ASR recipe developed for Kaldi [26].

For evaluating the spatial cues, we used the estimation errors of the Inter-microphone Time Difference of arrival (ΔITD) between the clean and estimated signals, the errors of the Inter-microphone Level Difference (ΔILD), and the log-determinant divergence (LDD) [27].

ΔITD and ΔILD evaluate the accuracy of the spatial cue preservation from the clean speech signal, and are defined:

$$\Delta_{ITD} = \left|\mathrm{ITD}\left(\mathbf{s}^1, \mathbf{s}^2\right) - \mathrm{ITD}\left(\hat{\mathbf{s}}^1, \hat{\mathbf{s}}^2\right)\right|, \quad (4)$$

$$\Delta_{ILD} = \left|10\log_{10}\frac{\|\mathbf{s}^1\|_2^2}{\|\mathbf{s}^2\|_2^2} - 10\log_{10}\frac{\|\hat{\mathbf{s}}^1\|_2^2}{\|\hat{\mathbf{s}}^2\|_2^2}\right|. \quad (5)$$

Here $\mathbf{s}^1, \mathbf{s}^2 \in \mathbb{R}^{\mathcal{T}}$ are clean speech signals with the length $\mathcal{T}$ in the time domain at the 1st and 2nd microphones, and $\hat{\mathbf{s}}^1, \hat{\mathbf{s}}^2 \in \mathbb{R}^{\mathcal{T}}$ are their estimated signals. $\|\cdot\|_2$ denotes the L_2-norm of a signal. $\mathrm{ITD}(\cdot,\cdot)$ calculates the Time Difference Of Arrival (TDOA) of the signals estimated using the Generalized Cross-Correlation PHAse Transform (GCC-PHAT) [28].

LDD, a metric for the distance between spatial covariance matrices $\Phi_{\boldsymbol{x}}$ and $\Phi_{\hat{\boldsymbol{x}}}$ of clean and estimated speech signals, is defined:

$$D_{\mathrm{LDD}}(\Phi_{\hat{\boldsymbol{x}}}\|\Phi_{\boldsymbol{x}}) = \mathrm{tr}\left(\Phi_{\hat{\boldsymbol{x}}}\Phi_{\boldsymbol{x}}^{-1}\right) - \log\det\left(\Phi_{\hat{\boldsymbol{x}}}\Phi_{\boldsymbol{x}}^{-1}\right) - M, \quad (6)$$

where $\mathrm{tr}(\cdot)$ denotes a matrix trace. LDD quantifies the difference in the spatial distributions of the two signals.

In the experiments, we calculated ΔITD, ΔILD, and LDD for each short-time segment, excluding non-speech segments, and averaged them over the segments to get the final figures.

5.3. Analysis condition

We implemented mSGMSE by modifying the publicly available code[1] for SGMSE. We trained the score model using the Adam optimizer with a learning rate 1.0×10^{-4} and exponential weight averaging. We set $\gamma = 1.5$, $c = 1.15 \times 10^{-2}$ and $k = 10.0$. We used short-time Fourier transform with transformed amplitudes [3]. We used predictor-corrector sampling [2] to solve the reverse process and set the number of diffusion steps at 30.

5.4. Evaluation results in terms of signal quality

Table 1 shows the evaluation results under the "matched" condition using WSJ0-CHiME3, with and without MIMO WPE preprocessing. All the signal quality metrics (SI-SDR, fwsSNR, PESQ, and ESTOI) were improved by both SGMSE and mSGMSG, and mSGMSE outperformed SGMSE for all metrics except for a few exceptions, i.e., SI-SDR and ESTOI, which were obtained without preprocessing.

Table 2 shows the evaluation results under the "mismatched" condition using REVERB, with and without MIMO WPE preprocessing. The same tendency as the matched condition was obtained for SDR and fwsSNR. In addition, although both mSGMSE and SGMSE effectively reduce WER for SIMU and REAL, mSGMSE largely outperformed SGMSE.

5.5. Evaluation results in terms of spatial cue preservation

Table 1 also shows the evaluation results in terms of spatial cues (ΔITD, ΔILD, and LDD) under the matched condition. Although SGMSE failed to improve ΔITD and ΔILD, mSGMSE very effectively improved them with and without WPE preprocessing. In contrast, LDD was improved by both mSGMSE and SGMSE, and again mSGMSE largely outperformed SGMSE.

5.6. Evaluation results in terms of RTF

Table 3 shows the RTFs achieved by mSGMSE and SGMSE for WSJ0-CHiME3, measured using a single GPU of Nvidia RTX A6000. RTF, which evaluates the computation time of the SE methods, is defined as the average computation time [s] required for processing a 1-s utterance. In the table, mSGMSE reduced the RTF for MIMO SE almost by a factor of $M(=2)$.

[1] https://github.com/sp-uhh/sgmse

Table 1: Evaluation under matched condition using WSJ0-CHiME3 with various array geometries: Obs denotes observed signal.

	Use WPE	SI-SDR [dB]	fwsSNR [dB]	PESQ	ESTOI	ΔITD [ms]	ΔILD [dB]	LDD ($\times 10^4$)
Obs	-	-4.3	4.60	1.24	0.47	0.029	0.584	6.11
SGMSE	-	**6.5**	11.4	2.40	**0.85**	0.183	0.701	1.06
mSGMSE	-	6.2	**11.6**	**2.42**	**0.85**	**0.002**	**0.232**	**0.19**
Obs	✓	-1.9	4.90	1.33	0.56	0.009	0.539	2.51
SGMSE	✓	7.1	11.4	2.28	0.84	0.289	0.720	0.89
mSGMSE	✓	**7.6**	**12.2**	**2.56**	**0.87**	**0.002**	**0.217**	**0.21**

Table 2: Evaluation under mismatched condition using REVERB challenge dataset with unknown array geometries in simulated and real evaluation data (SIMU and REAL).

	Use	SIMU			REAL
	WPE	SDR [dB]	fwsSNR [dB]	WER [%]	WER [%]
Obs	-	9.54	3.62	7.40	18.61
SGMSE	-	**10.96**	9.74	5.25	15.18
mSGMSE	-	10.41	**10.25**	**5.03**	**13.90**
Obs	✓	11.78	4.35	4.72	13.80
SGMSE	✓	11.92	9.37	5.27	15.18
mSGMSE	✓	**12.65**	**10.17**	**4.58**	**11.34**

Table 3: Real-Time Factor (RTF) for performing MIMO SE ($M = 2$) measured using a single GPU of Nvidia RTX A6000.

	SGMSE	mSGMSE
RTF	1.9912	**0.9995**

6. CONCLUSIONS

We introduced mSGMSE, a MIMO SE method based on a diffusion model. mSGMSE extends SGMSE, a SISO SE method, by modeling the conditional distribution of clean speech signals from multiple microphones. We also presented a computationally efficient score model, mNCSN++, and a MAG training scheme that enables mSGMSE to adapt to various or unknown array geometries. We experimentally demonstrated that mSGMSE enhances signal quality and spatial cues for MIMO SE under both matched and mismatched conditions, compared to individually applying SGMSE to each microphone. Furthermore, mSGMSE reduced the computational time of SGMSE by a factor of $M(=2)$. We showed that mSGMSE significantly improved the ASR performance for the REVERB challenge real evaluation set, even though we trained mSGMSE only using very different simulation data. We also revealed that using MIMO WPE preprocessing further improved mSGMSE.

Future work will evaluate mSGMSE more thoroughly, with more microphones, by comparing it with discriminative NN-based MIMO SE methods and by combining it with advanced signal processing techniques.

7. ACKNOWLEDGEMENTS

This work was supported by JSPS KAKENHI Grant Number 23H03423.

8. REFERENCES

[1] Jonathan Ho, Ajay Jain, and Pieter Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*, 2020.

[2] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole, "Score-based generative modeling through stochastic differential equations," in *Proc. International Conference on Learning Representations (ICLR)*, 2021.

[3] Julius Richter, Simon Welker, Jean-Marie Lemercier, Bunlong Lay, and Timo 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.

[4] Elior Hadad, Daniel Marquardt, Simon Doclo, and Sharon Gannot, "Theoretical analysis of binaural transfer function MVDR beamformers with interference cue preservation constraints," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 12, pp. 2449–2464, 2015.

[5] Simon Doclo, Thomas J. Klasen, Tim Van den Bogaert, Jan Wouters, and Marc Moonen, "Theoretical analysis of binaural cue preservation using multi-channel Wiener filtering and interaural transfer functions," in *Proc. International Workshop on Acoustitc Echo and Noise Control (IWAENC)*, 2006, pp. 1–4.

[6] Takuya Yoshioka and Tomohiro Nakatani, "Generalization of multi-channel linear prediction methods for blind MIMO impulse response shortening," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 10, pp. 2707–2720, 2012.

[7] Xingwei Sun, Risheng Xia, Junfeng Li, and Yonghong Yan, "A deep learning based binaural speech enhancement approach with spatial cues preservation," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 5766–5770.

[8] Cong Han, Yi Luo, and Nima Mesgarani, "Real-time binaural speech separation with preserved spatial cues," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6404–6408.

[9] Jung-Hee Kim, Jeonghwan Choi, Jinyoung Son, Gyeong-Su Kim, Jihwan Park, and Joon-Hyuk Chang, "MIMO noise suppression preserving spatial cues for sound source localization in mobile robot," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, 2021, pp. 1–5.

[10] Yen-Ju Lu, Zhong-Qiu Wang, Shinji Watanabe, Alexander Richard, Cheng Yu, and Yu Tsao, "Conditional diffusion probabilistic model for speech enhancement," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 7402–7406.

[11] Simon Welker, Julius Richter, and Timo Gerkmann, "Speech enhancement with score-based generative models in the complex STFT domain," in *Proc. Interspeech*, 2022, pp. 2928–2932.

[12] Joan Serrà, Santiago Pascual, Jordi Pons, R Oguz Araz, and Davide Scaini, "Universal speech enhancement with score-based diffusion," in *arXiv preprint arXiv:2206.03065*, 2022.

[13] Naoyuki Kamo, Marc Delcroix, and Tomohiro Nakatani, "Target speech extraction with conditional diffusion model," in *Proc. Interspeech*, 2023, pp. 176–180.

[14] George E. Uhlenbeck and Leonard S. Ornstein, "On the theory of the brownian motion," *Physical review*, vol. 36, no. 5, pp. 823–841, 1930.

[15] Brian David and Outram Anderson, "Reverse-time diffusion equation models," *Stochastic Processes and their Applications*, vol. 12, no. 3, pp. 313–326, 1982.

[16] Georgios Batzolis, Jan Stanczuk, Carola-Bibiane Schonlieb, and Christian Etmann, "Conditional image generation with score-based diffusion models," in *arXiv:2111.13606v1*, 2021.

[17] Simo Särkkä and Arno Solin, *Applied Stochastic Differential Equations*, Cambridge University Press, 2019.

[18] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. International conference on Conference Medical Image Computing and Computer-Assisted Intervention– (MICCAI)*, 2015, pp. 234–241.

[19] John S. Garofolo, David Graff, Doug Paul, and David Pallett, "CSR-I (WSJ0) complete," `https://catalog.ldc.upenn.edu/LDC93S6A`.

[20] Jon Barker, Ricard Marxer, Emmanuel Vincent, and Shinji Watanabe, "The third CHiME speech separation and recognition challenge: Dataset, task and baselines," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015, pp. 504–511.

[21] Keisuke Kinoshita, Marc Delcroix, Sharon Gannot, Emanuël A P. Habets, Reinhold Haeb-Umbach, Walter Kellermann, Volker Leutnant, Roland Maas, Tomohiro Nakatani, Bhiksha Raj, et al., "A summary of the reverb challenge: state-of-the-art and remaining challenges in reverberant speech processing research," *EURASIP Journal on Advances in Signal Processing*, vol. 2016, no. 7, pp. 1–19, 2016.

[22] Emmanuel Vincent, Rémi Gribonval, and Cédric Févotte, "Performance measurement in blind audio source separation," *IEE Trans. Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[23] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R Hershey, "SDR–half-baked or well done?," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 626–630.

[24] Antony W. Rix, John G. Beerends, Michael P. Hollier, and Andries P. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2001, vol. 2, pp. 749–752.

[25] Asger Heidemann Andersen, Jan Mark de Haan, Zheng-Hua Tan, and Jesper Jensen, "Refinement and validation of the binaural short time objective intelligibility measure for spatially diverse conditions," *Speech Communication*, vol. 102, pp. 1–13, 2018.

[26] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely, "The Kaldi speech recognition toolkit," in *Proc. IEEE workshop on Automatic Speech Recognition and Understanding*.

[27] Brian J. Kulis, Mátyás Sustik, and Inderjit S. Dhillon, "Learning low-rank kernel matrices," in *Proc. International Confefence on Machine Learning (ICML)*, 2006, pp. 505–512.

[28] Charles Knapp and Glifford Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 4, pp. 320–327, 1976.