DNN-based Source Enhancement to Increase Objective Sound Quality Assessment Score

Yuma Koizumi[1] *Member, IEEE,*, Kenta Niwa[1] *Member, IEEE,*, Yusuke Hioka[2] *Senior Member, IEEE*, Kazunori Kobayashi[1] and Yoichi Haneda[3] *Senior Member, IEEE,*

***Abstract*—We propose a training method for deep neural network (DNN)-based source enhancement to increase objective sound quality assessment (OSQA) scores such as the perceptual evaluation of speech quality (PESQ). In many conventional studies, DNNs have been used as a mapping function to estimate time-frequency masks and trained to minimize an analytically tractable objective function such as the mean squared error (MSE). Since OSQA scores have been used widely for sound-quality evaluation, constructing DNNs to increase OSQA scores would be better than using the minimum-MSE to create high-quality output signals. However, since most OSQA scores are not analytically tractable, *i.e.*, they are black boxes, the gradient of the objective function cannot be calculated by simply applying back-propagation. To calculate the gradient of the OSQA-based objective function, we formulated a DNN optimization scheme on the basis of *black-box optimization*, which is used for training a computer that plays a game. For a black-box-optimization scheme, we adopt the policy gradient method for calculating the gradient on the basis of a sampling algorithm. To simulate output signals using the sampling algorithm, DNNs are used to estimate the probability-density function of the output signals that maximize OSQA scores. The OSQA scores are calculated from the simulated output signals, and the DNNs are trained to increase the probability of generating the simulated output signals that achieve high OSQA scores. Through several experiments, we found that OSQA scores significantly increased by applying the proposed method, even though the MSE was not minimized.**

***Index Terms*—Sound-source enhancement, time-frequency mask, deep learning, objective sound quality assessment (OSQA) score.**

I. INTRODUCTION

SOUND-source enhancement has been studied for many years [1]–[6] because of the high demand for its use for various practical applications such as automatic speech recognition [7]–[9], hands-free telecommunication [10], [11], hearing aids [12]–[15], and immersive audio field representation [16], [17]. In this study, we aimed at generating an enhanced target source with high listening quality because the processed sounds are assumed perceived by humans.

[1]: NTT Media Intelligence Laboratories, NTT Corporation, Tokyo, Japan (e-mail: koizumi.yuma@ieee.org, niwa.kenta, kobayashi.kazunori@lab.ntt.co.jp)
[2]: Department of Mechanical Engineering, University of Auckland, 20 Symonds Street, Auckland, 1010 New Zealand (e-mail: yusuke.hioka@ieee.org)
[3]: Department of Informatics, The University of Electro-Communications, Tokyo, Japan (e-mail: haneda.yoichi@uec.ac.jp)

Recently, deep learning [18] has been successfully used for sound-source enhancement [8], [15], [19]–[35] . In many of these conventional studies, deep neural networks (DNNs) were used as a regression function to estimate time-frequency (T-F) masks [19]–[22] and/or amplitude-spectra of the target source [23]–[31]. The parameters of the DNNs were trained using back-propagation [36] to minimize an analytically tractable objective function such as the mean squared error (MSE) between supervised outputs and DNN outputs. In recent studies, advanced analytical objective functions were used such as the maximum-likelihood (ML) [31], [32], the combination of multi-types of MSE [25]–[27], the Kullback-Leibler and/or Itakura-Saito divergence [33], the modified short-time intelligibility measure (STOI) [22], the clustering cost [34], and the discriminative cost of a clean target source and output signal using a generative adversarial network (GAN) [35].

When output sound is perceived by humans, the objective function that reflects human perception may not be analytically tractable, *i.e.*, it is a black-box function. In the past few years, objective sound quality assessment (OSQA) scores, such as the perceptual evaluation of speech quality (PESQ) [37] and STOI [38], have been commonly used to evaluate output sound quality. Thus, it might be better to construct DNNs to increase OSQA scores directly. However, since typical OSQA scores are not analytically defined (*i.e.*, they are black-box functions), the gradient of the objective function cannot be calculated by simply applying back-propagation.

We previously proposed a DNN training method to estimate T-F masks and increase OSQA scores [39]. To overcome the problem that the objective function to maximize the OSQA scores is not analytically tractable, we developed a DNN-training method on the basis of the *black-box optimization* framework [40], as used in predicting the winning percentage of the game Go [41]. The basic idea of block-box optimization is estimating a gradient from randomly simulated output. For example, in the training of a DNN for the Go-playing computer, the computer determines a "*move*" (where to put a Go-stone) depending on the DNN output. Then, when the computer won the game, a gradient is calculated to increase the selection probability of the selected "*moves*". We adopt this strategy to increase the OSQA scores; some output signals are randomly simulated and a DNN is trained to increase the generation probability of the simulated output signals that achieved high OSQA scores. For the first trial, we prepared a finite number of T-F mask templates and trained DNNs to select the best template that maximizes the OSQA score. Although we found that the OSQA scores increased using this

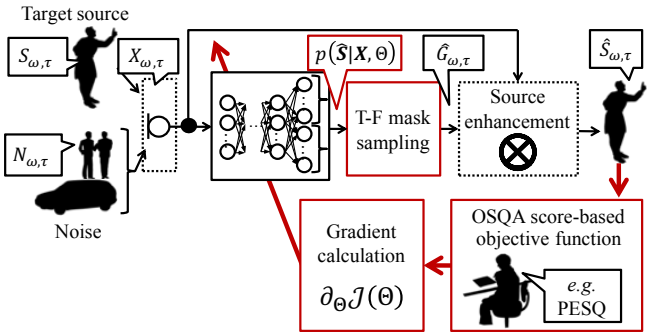


Fig. 1. Concept of proposed method

method, the output performances would improve by extending the method to a more flexible T-F mask design scheme from the template-selection scheme.

In this study, to arbitrarily estimate T-F masks, we modified the DNN source enhancement architecture to estimate the latent parameters in a continuous probability density function (PDF) of the T-F mask processing output signals, as shown in Fig. 1. To calculate the gradient of the objective function, we adopt the policy gradient method [42] as a black-box optimization scheme. With our method, the estimated latent parameters construct a continuous PDF as the "*policy*" of T-F-mask estimation to increase OSQA scores. On the basis of this policy, the output signals are directly simulated using the sampling algorithm. Then, the gradient of the DNN is estimated to increase/decrease the generation probability of output signals with high/low OSQA scores, respectively. The sampling from continuous PDF causes the estimate of the gradient to fluctuate, resulting in unstable training behavior. To avoid this problem, we additionally formulate two tricks: i) score normalization to reduce the variance in the estimated gradient, and ii) a sampling algorithm to simulate output signals to satisfy the constraint of T-F mask processing.

The rest of this paper is organized as follows. Section II introduces DNN source enhancement based on the ML approach. In Section III, we propose our DNN training method to increase OSQA scores on the basis of the black-box optimization. After investigating the sound quality of output signals through several experiments in Section IV, we conclude this paper in Section V.

II. CONVENTIONAL METHOD

A. Sound source enhancement with time-frequency mask

Let us consider the problem of estimating a target source $S_{\omega,\tau} \in \mathbb{C}$, which is surrounded by ambient noise $N_{\omega,\tau} \in \mathbb{C}$. A signal observed with a single microphone $X_{\omega,\tau} \in \mathbb{C}$ is assumed to be modeled as

$$X_{\omega,\tau} = S_{\omega,\tau} + N_{\omega,\tau}, \tag{1}$$

where $\omega = \{1, 2, ..., \Omega\}$ and $\tau = \{1, 2, ..., T\}$ denote the frequency and time indices, respectively.

In sound-source enhancement using T-F masks, the output signal $\hat{S}_{\omega,\tau}$ is obtained by multiplying a T-F mask by $X_{\omega,\tau}$ as

$$\hat{S}_{\omega,\tau} = G_{\omega,\tau} X_{\omega,\tau}, \tag{2}$$

where $0 \leq G_{\omega,\tau} \leq 1$ is a T-F mask. The IRM $G^{\mathrm{IRM}}_{\omega,\tau}$ [8] is an implementation of T-F mask, which is defined by

$$G^{\mathrm{IRM}}_{\omega,\tau} = \frac{|S_{\omega,\tau}|}{|S_{\omega,\tau}| + |N_{\omega,\tau}|}. \tag{3}$$

The IRM maximizes the signal-to-noise-ratio (SNR) when the phase spectrum of $S_{\omega,\tau}$ coincides with that of $N_{\omega,\tau}$. However, this assumption is almost never satisfied in most practical cases. To compensate for this mismatch, the phase sensitive spectrum approximation (PSA) [19], [20] was proposed

$$G^{\mathrm{PSA}}_{\omega,\tau} = \min\left(1, \max\left(0, \frac{|S_{\omega,\tau}|}{|X_{\omega,\tau}|}\cos\left(\theta^{(S)}_{\omega,\tau} - \theta^{(X)}_{\omega,\tau}\right)\right)\right), \tag{4}$$

where $\theta^{(S)}_{\omega,\tau}$ and $\theta^{(X)}_{\omega,\tau}$ are the phase spectra of $S_{\omega,\tau}$ and $X_{\omega,\tau}$, respectively. Since the PSA $G^{\mathrm{PSA}}_{\omega,\tau}$ is a T-F mask that minimizes the squared error between $S_{\omega,\tau}$ and $\hat{S}_{\omega,\tau}$ on the complex plane, we use this as a T-F masking scheme.

B. Maximum-likelihood-based DNN training for T-F mask estimation

In many conventional studies of DNN-based source enhancement, DNNs were used as a mapping function to estimate T-F masks. In this section, we explain DNN training based on ML estimation, on which the proposed method is based. Since the ML-based approach explicitly models the PDF of the target source, it becomes possible to simulate output signals by generating random numbers from the PDF.

In ML-based training, the DNNs are constructed to estimate the parameters of the conditional PDF of the target source providing the observation is given by $p(\boldsymbol{S}_\tau|\boldsymbol{X}_\tau, \Theta)$. Here, Θ denotes the DNN parameters. Its example on a fully connected DNN is described later (after (16)). The target and observation source are assumed to be vectorized for all frequency bins as

$$\boldsymbol{S}_\tau := (S_{1,\tau}, ..., S_{\Omega,\tau})^\top, \tag{5}$$

$$\boldsymbol{X}_\tau := (X_{1,\tau}, ..., X_{\Omega,\tau})^\top, \tag{6}$$

where $\top$ is transposition. Then Θ is trained to maximize the expectation of the log-likelihood as

$$\Theta \leftarrow \arg\max_\Theta \mathcal{J}^{\mathrm{ML}}(\Theta), \tag{7}$$

where the objective function $\mathcal{J}^{\mathrm{ML}}(\Theta)$ is defined by

$$\mathcal{J}^{\mathrm{ML}}(\Theta) = \mathbb{E}_{\boldsymbol{S},\boldsymbol{X}}\left[\ln p(\boldsymbol{S}|\boldsymbol{X}, \Theta)\right], \tag{8}$$

and $\mathbb{E}_x[\cdot]$ denotes the expectation operator for x. However, since (8) is difficult to analytically calculate, the expectation calculation is replaced with the average of the training dataset as

$$\mathcal{J}^{\mathrm{ML}}(\Theta) \approx \frac{1}{T}\sum_{\tau=1}^{T} \ln p(\boldsymbol{S}_\tau|\boldsymbol{X}_\tau, \Theta). \tag{9}$$

The back-propagation algorithm [36] is used in training Θ to maximize (9). When $p(\boldsymbol{S}_\tau|\boldsymbol{X}_\tau, \Theta)$ is composed of differentiable functions with respect to Θ, the gradient is calculated as

$$\partial_\Theta \mathcal{J}^{\mathrm{ML}}(\Theta) \approx \frac{1}{T}\sum_{\tau=1}^{T} \partial_\Theta \ln p(\boldsymbol{S}_\tau|\boldsymbol{X}_\tau, \Theta), \tag{10}$$

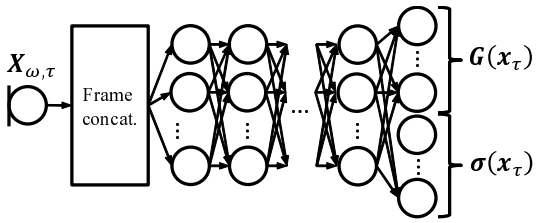


Fig. 2. ML-based DNN architecture used in T-F mask estimation

where ∂_x is a partial differential operator with respect to x.

To calculate (10), $p(\boldsymbol{S}_\tau|\boldsymbol{X}_\tau,\Theta)$ is modeled by assuming that the estimation error of $S_{\omega,\tau}$ is independent for all frequency bins and follows the zero-mean complex Gaussian distribution with the variance $\sigma^2_{\omega,\tau}$. The assumption is based on state-of-the-art methods, which train DNNs to minimize the MSE between $S_{\omega,\tau}$ and $\hat{G}_{\omega,\tau}X_{\omega,\tau}$ on the complex plane [19], [20]. The minimum-MSE (MMSE) on the complex plane is equivalent to assuming that the errors are independent for all frequency bins and follow the zero-mean complex Gaussian distribution with variance 1. Our assumption relaxes the assumption of the conventional methods; the variances of each frequency bin vary according to the error values to maximize the likelihood. Thus, since $\hat{S}_{\omega,\tau}$ is given by $\hat{G}_{\omega,\tau}X_{\omega,\tau}$, $p(\boldsymbol{S}_\tau|\boldsymbol{X}_\tau,\Theta)$ is modeled by the following complex Gaussian distribution as

$$p(\boldsymbol{S}_\tau|\boldsymbol{X}_\tau,\Theta)=\prod_{\omega=1}^{\Omega}\frac{1}{2\pi\sigma^2_{\omega,\tau}}\exp\left\{-\frac{\left|S_{\omega,\tau}-\hat{G}_{\omega,\tau}X_{\omega,\tau}\right|^2}{2\sigma^2_{\omega,\tau}}\right\}. \quad (11)$$

In this model, it can be regarded that the MSE between $S_{\omega,\tau}$ and $\hat{S}_{\omega,\tau}$ on the complex plane is extended to the likelihood of $S_{\omega,\tau}$ defined on the complex Gaussian distribution, the mean and variance parameters of which are $\hat{S}_{\omega,\tau}$ and $\sigma^2_{\omega,\tau}$, respectively. (11) includes unknown parameters: the T-F mask $\hat{G}_{\omega,\tau}$ and error variance $\sigma^2_{\omega,\tau}$. Thus, we construct DNNs to estimate $\hat{G}_{\omega,\tau}$ and $\sigma^2_{\omega,\tau}$ from $\boldsymbol{X}_\tau$, as shown in Fig. 2. The vectorized T-F masks and error variances for all frequency bins are defined as

$$\boldsymbol{G}(\boldsymbol{x}_\tau):=\left(\hat{G}_{1,\tau},...,\hat{G}_{\Omega,\tau}\right)^\top, \quad (12)$$

$$\boldsymbol{\sigma}(\boldsymbol{x}_\tau):=\left(\sigma^2_{1,\tau},...,\sigma^2_{\Omega,\tau}\right)^\top. \quad (13)$$

Here $\boldsymbol{x}_\tau$ is the input vector of DNNs that is prepared by concatenating several frames of observations to account for previous and future Q frames as $\boldsymbol{x}_\tau=(\boldsymbol{X}_{\tau-Q},...,\boldsymbol{X}_\tau,...,\boldsymbol{X}_{\tau+Q})^\top$, and $\boldsymbol{G}(\boldsymbol{x}_\tau)$ and $\boldsymbol{\sigma}(\boldsymbol{x}_\tau)$ are estimated by

$$\boldsymbol{G}(\boldsymbol{x}_\tau)\leftarrow\phi_g\left\{\mathbf{W}^{(\mu)}\boldsymbol{z}^{(L-1)}_\tau+\mathbf{b}^{(\mu)}\right\}, \quad (14)$$

$$\boldsymbol{\sigma}(\boldsymbol{x}_\tau)\leftarrow\phi_\sigma\left\{\mathbf{W}^{(\sigma)}\boldsymbol{z}^{(L-1)}_\tau+\mathbf{b}^{(\sigma)}\right\}+C_\sigma, \quad (15)$$

$$\boldsymbol{z}^{(l)}_\tau=\phi_h\left\{\mathbf{W}^{(l)}\boldsymbol{z}^{(l-1)}_\tau+\mathbf{b}^{(l)}\right\}, \quad (16)$$

where C_σ is a small positive constant value to prevent the variance from being very small. Here, l, L, $\mathbf{W}^{(l)}$, and $\mathbf{b}^{(\cdot)}$ are the layer index, number of layers, weight matrix, and bias vector, respectively. $\mathbf{W}^{(\mu)},\mathbf{W}^{(\sigma)}$ are the weight matrices and $\mathbf{b}^{(\mu)},\mathbf{b}^{(\sigma)}$ are the bias vectors to estimate the T-F mask and variance, respectively. The DNN parameters are composed of $\Theta=\{\mathbf{W}^{(\mu)},\mathbf{b}^{(\mu)},\mathbf{W}^{(\sigma)},\mathbf{b}^{(\sigma)},\mathbf{W}^{(l)},\mathbf{b}^{(l)}|l\in(2,...,L-1)\}$. The functions ϕ_g, ϕ_σ, and ϕ_h are nonlinear activation functions, and in conventional studies, sigmoid and exponential functions were used as an implementation of ϕ_g [19], [20] and ϕ_σ [32], respectively. The input vector $\boldsymbol{x}_\tau$ is passed to the first layer of the network as $\boldsymbol{z}^{(1)}_\tau=\boldsymbol{x}_\tau$.

III. PROPOSED METHOD

Our proposed DNN-training method increases OSQA scores. With the proposed method, the policy gradient method [42] is used to statistically calculate the gradient with respect to Θ by using a sampling algorithm, even though the objective function is not differentiable. However, sampling-based gradient estimation would frequently make the DNN training behavior become unstable. To avoid this problem, we introduce two tricks: i) score normalization that reduces the variance in the estimated gradient (in Sec. III-B), and ii) a sampling algorithm to simulate output signals to satisfy the constraint of T-F mask processing (in Sec. III-C). Finally, the overall training procedure of the proposed method is summarized in Sec. III-D.

A. Policy gradient-based DNN training for T-F mask estimation

Let $\mathcal{B}(\hat{\boldsymbol{S}},\boldsymbol{X})$ be a scoring function that quantifies the sound quality of the estimated sound signal $\hat{\boldsymbol{S}}:=(\hat{S}_1,...,\hat{S}_\Omega)^\top$ defined by (2). To implement $\mathcal{B}(\hat{\boldsymbol{S}},\boldsymbol{X})$, subjective evaluation is simple. However, it would be difficult to use in practical implementation because DNN training requires a massive amount of listening-test results. Thus, $\mathcal{B}(\hat{\boldsymbol{S}},\boldsymbol{X})$ quantifies the sound quality based on OSQA scores, as shown in Fig. 1, and the details of its implementation are discussed in Sec. III-B. We assume $\mathcal{B}(\hat{\boldsymbol{S}},\boldsymbol{X})$ is non-differentiable with respect to Θ, because most OSQA scores are black-box functions.

Let us consider the expectation maximization of $\mathcal{B}(\hat{\boldsymbol{S}},\boldsymbol{X})$ as a metric of performance of the sound-source enhancement that increases OSQA scores as

$$\mathbb{E}_{\hat{S},X}\left[\mathcal{B}(\hat{\boldsymbol{S}},\boldsymbol{X})\right]=\iint\mathcal{B}(\hat{\boldsymbol{S}},\boldsymbol{X})p(\hat{\boldsymbol{S}},\boldsymbol{X})d\hat{\boldsymbol{S}}d\boldsymbol{X}. \quad (17)$$

Since the output signal $\hat{\boldsymbol{S}}$ is calculated from the observation $\boldsymbol{X}$, we decompose the joint PDF $p(\hat{\boldsymbol{S}},\boldsymbol{X})$ into the conditional PDF of the output signal given the observation $p(\hat{\boldsymbol{S}}|\boldsymbol{X})$ and the marginal PDF of the observation $p(\boldsymbol{X})$ as $p(\hat{\boldsymbol{S}},\boldsymbol{X})=p(\hat{\boldsymbol{S}}|\boldsymbol{X})p(\boldsymbol{X})$. Then, (17) can be reformed as

$$\mathbb{E}_{\hat{S},X}\left[\mathcal{B}(\hat{\boldsymbol{S}},\boldsymbol{X})\right]=\int p(\boldsymbol{X})\int\mathcal{B}(\hat{\boldsymbol{S}},\boldsymbol{X})p(\hat{\boldsymbol{S}}|\boldsymbol{X})d\hat{\boldsymbol{S}}d\boldsymbol{X}. \quad (18)$$

We use DNNs to estimate the parameters of the conditional PDF of the output signal $p(\hat{\boldsymbol{S}}|\boldsymbol{X},\Theta)$, as with the case of ML-based training. For example, the complex Gaussian distribution in (11) can be used as $p(\hat{\boldsymbol{S}}|\boldsymbol{X},\Theta)$. To train Θ, $\mathbb{E}_{\hat{S},X}[\mathcal{B}(\hat{\boldsymbol{S}},\boldsymbol{X})]$ is used as an objective function by replacing the conditional PDF $p(\hat{\boldsymbol{S}}|\boldsymbol{X})$ with $p(\hat{\boldsymbol{S}}|\boldsymbol{X},\Theta)$ as

$$\mathcal{J}(\Theta)=\mathbb{E}_{\hat{S},X}\left[\mathcal{B}(\hat{\boldsymbol{S}},\boldsymbol{X})\right], \quad (19)$$

$$=\int p(\boldsymbol{X})\int\mathcal{B}(\hat{\boldsymbol{S}},\boldsymbol{X})p(\hat{\boldsymbol{S}}|\boldsymbol{X},\Theta)d\hat{\boldsymbol{S}}d\boldsymbol{X}. \quad (20)$$

Since $\mathcal{B}(\hat{\boldsymbol{S}}, \boldsymbol{X})$ is non-differentiable with respect to Θ, the gradient of (20) cannot be analytically obtained by simply applying back-propagation. Hence, we apply the policy-gradient method [42], which can statistically calculate the gradient of a black-box objective function. By assuming that the function form of $\mathcal{B}(\hat{\boldsymbol{S}}, \boldsymbol{X})$ is smooth, $\mathcal{B}(\hat{\boldsymbol{S}}, \boldsymbol{X})$ is a continuous function and its derivative exists. In addition, we assume $p(\hat{\boldsymbol{S}}|\boldsymbol{X}, \Theta)$ is composed with differentiable functions with respect to Θ. Then, the gradient of (20) can be calculated using a log-derivative trick [42] $\partial_x p(\boldsymbol{x}) = p(\boldsymbol{x})\partial_x \ln p(\boldsymbol{x})$ as

$$\partial_\Theta \mathcal{J}(\Theta) = \int p(\boldsymbol{X}) \int \mathcal{B}(\hat{\boldsymbol{S}}, \boldsymbol{X}) \partial_\Theta p(\hat{\boldsymbol{S}}|\boldsymbol{X}, \Theta) d\hat{\boldsymbol{S}} d\boldsymbol{X}, \tag{21}$$

$$= \mathbb{E}_{\boldsymbol{X}} \left[\mathbb{E}_{\hat{\boldsymbol{S}}|\boldsymbol{X}} \left[\mathcal{B}(\hat{\boldsymbol{S}}, \boldsymbol{X}) \partial_\Theta \ln p(\hat{\boldsymbol{S}}|\boldsymbol{X}, \Theta) \right] \right]. \tag{22}$$

Since the expectation in (22) cannot be analytically calculated, the expectation with respect to $\boldsymbol{X}$ is approximated by averaging the training data, and the average of $\hat{\boldsymbol{S}}$ is calculated using the sampling algorithm as

$$\partial_\Theta \mathcal{J}(\Theta) \approx \frac{1}{T} \sum_{\tau=1}^{T} \frac{1}{K} \sum_{k=1}^{K} \mathcal{B}(\hat{\boldsymbol{S}}_\tau^{(k)}, \boldsymbol{X}_\tau) \partial_\Theta \ln p(\hat{\boldsymbol{S}}_\tau^{(k)}|\boldsymbol{X}_\tau, \Theta), \tag{23}$$

$$\hat{\boldsymbol{S}}_\tau^{(k)} \sim p(\hat{\boldsymbol{S}}|\boldsymbol{X}_\tau, \Theta), \tag{24}$$

where $\hat{\boldsymbol{S}}_\tau^{(k)}$ is the k-th simulated output signal and K is the number of samplings, which is assumed to be sufficiently large. The superscript (k) represents the variable of the k-th sampling, and $\sim$ is a sampling operator from the right-side distribution. The details of the sampling process for (24) are described in Sec. III-C.

Most OSQA scores, such as PESQ, are designed for their scores to be calculated using several time frames such as one utterance of a speech sentence. Since $\mathcal{B}(\hat{\boldsymbol{S}}_\tau^{(k)}, \boldsymbol{X}_\tau)$ of every time frame τ cannot be obtained, the gradient cannot be calculated by (23). Thus, instead of using the average of τ, we use the average of $\mathcal{I}$ utterances. We define the observation of the i-th utterance as $\mathbf{X}^{(i)} := (\boldsymbol{X}_1^{(i)}, ..., \boldsymbol{X}_{T^{(i)}}^{(i)})$, and the k-th output signal of the i-th utterance as $\hat{\mathbf{S}}^{(i,k)} := (\hat{\boldsymbol{S}}_1^{(i,k)}, ..., \hat{\boldsymbol{S}}_{T^{(i)}}^{(i,k)})$. Then the gradient can be calculated as

$$\partial_\Theta \mathcal{J}(\Theta) \approx \frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} \partial_\Theta \mathcal{J}^{(i)}(\Theta), \tag{25}$$

$$\partial_\Theta \mathcal{J}^{(i)}(\Theta) \approx \sum_{k=1}^{K} \frac{\mathcal{B}\left(\hat{\mathbf{S}}^{(i,k)}, \mathbf{X}^{(i)}\right)}{K T^{(i)}} \sum_{\tau=1}^{T^{(i)}} \partial_\Theta \ln p(\hat{\boldsymbol{S}}_\tau^{(i,k)}|\boldsymbol{X}_\tau^{(i)}, \Theta), \tag{26}$$

where $T^{(i)}$ is the frame length of the i-th utterance, and we assume that the output signal of each time frame is calculated independently. The details of the deviation of (25) are described in the Appendix A.

B. Scoring-function design for stable training

We now introduce a design of a scoring function $\mathcal{B}(\hat{\mathbf{S}}, \mathbf{X})$ to stabilize the training process. Because the expectation for the gradient calculation in (22) is approximated using the sampling algorithm, the training may become unstable. One reason for unstable training behavior is that the variance in the estimated gradient becomes large in accordance with the large variance in the scoring-function output [42]. To stabilize the training, instead of directly using a raw OSQA score as $\mathcal{B}(\hat{\mathbf{S}}, \mathbf{X})$, a normalized OSQA score is used to reduce its variance. Hereafter, a raw OSQA score calculated from $\mathbf{S}$, $\mathbf{X}$ and $\hat{\mathbf{S}}$ is written as $\mathcal{Z}(\hat{\mathbf{S}}, \mathbf{X})$ to distinguish between a raw OSQA score $\mathcal{Z}(\hat{\mathbf{S}}, \mathbf{X})$ and normalized OSQA score $\mathcal{B}(\hat{\mathbf{S}}, \mathbf{X})$.

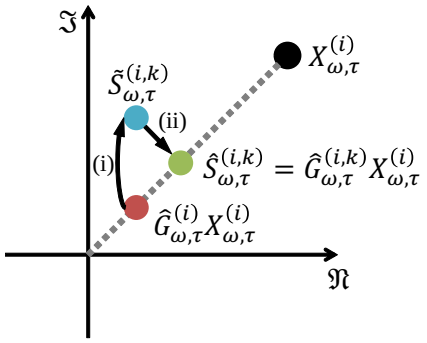


Fig. 3. T-F mask sampling procedure of proposed method on complex plane. The black, red, blue, and green points represent $X_{\omega,\tau}^{(i)}$, $\hat{G}_{\omega,\tau}^{(i)} X_{\omega,\tau}^{(i)}$, $\tilde{S}_{\omega,\tau}^{(i,k)}$, and $\hat{G}_{\omega,\tau}^{(i,k)} X_{\omega,\tau}^{(i)}$, respectively. First, the parameters of $p(\hat{S}_{\omega,\tau}|X_{\omega,\tau}^{(i)}, \Theta)$, *i.e.*, the T-F mask $\hat{G}_{\omega,\tau}^{(i)}$ and the variance are estimated using a DNN. Then, $\tilde{S}_{\omega,\tau}^{(i,k)}$ is sampled from $p(\hat{S}_{\omega,\tau}|X_{\omega,\tau}^{(i)}, \Theta)$ by using a typical sampling algorithm; which is shown as arrow-(i). Finally, the simulated T-F mask $\hat{G}_{\omega,\tau}^{(i,k)}$ is calculated to minimize the MSE between $\tilde{S}_{\omega,\tau}^{(i,k)}$ and the simulated output signal $\hat{G}_{\omega,\tau}^{(i,k)} X_{\omega,\tau}^{(i)}$ by (29); which is shown as arrow-(ii).

From (25) and (26), the total gradient $\partial_\Theta \mathcal{J}(\Theta)$ is a weighted sum of the i-th gradient of the log-likelihood function, and $\mathcal{B}(\hat{\mathbf{S}}, \mathbf{X})$ is used as its weight. Since typical OSQA scores vary not only by the performance of source enhancement but also by the SNRs of each input signal $\mathbf{X}^{(1,...,I)}$, $\partial_\Theta \mathcal{J}(\Theta)$ also varies by the OSQA scores and SNRs of $\mathbf{X}^{(1,...,I)}$. To reduce the variance in the estimate of the gradient, it would be better to remove such external factors according to the input conditions of each input signal, *e.g.*, input SNRs. As a possible solution, the external factors involved in the OSQA score would be estimated by calculating the expectation of the OSQA score of the input signal. Thus, subtracting the conditional expectation of $\mathcal{Z}(\hat{\mathbf{S}}, \mathbf{X})$ given by each input signal $\mathbb{E}_{\hat{\mathbf{S}}|\mathbf{X}}[\mathcal{Z}(\hat{\mathbf{S}}, \mathbf{X})]$ from $\mathcal{Z}(\hat{\mathbf{S}}, \mathbf{X})$ might be effective in reducing the variance as

$$\mathcal{B}\left(\hat{\mathbf{S}}, \mathbf{X}\right) = \mathcal{Z}(\hat{\mathbf{S}}, \mathbf{X}) - \mathbb{E}_{\hat{\mathbf{S}}|\mathbf{X}} \left[\mathcal{Z}(\hat{\mathbf{S}}, \mathbf{X}) \right]. \tag{27}$$

This implementation is known as "baseline-subtraction" [42], [43]. Here, $\mathbb{E}_{\hat{\mathbf{S}}|\mathbf{X}}[\mathcal{Z}(\hat{\mathbf{S}}, \mathbf{X})]$ cannot be analytically calculated, so we replace the expectation with the average of OSQA scores. Then the scoring function is designed as

$$\mathcal{B}\left(\hat{\mathbf{S}}^{(i,k)}, \mathbf{X}^{(i)}\right) = \mathcal{Z}(\hat{\mathbf{S}}^{(i,k)}, \mathbf{X}^{(i)}) - \frac{1}{K} \sum_{j=1}^{K} \mathcal{Z}(\hat{\mathbf{S}}^{(i,j)}, \mathbf{X}^{(i)}). \tag{28}$$

C. Sampling-algorithm to simulate T-F-mask-processed output signal

The sampling operator used in (24) is an intuitive method that uses a typical pseudo random number generator such as the Mersenne-Twister [44]. However, this sampling operator would in fact be difficult to use because typical sampling algorithms simulate output signals that do not satisfy the

constraint of real-valued T-F-mask processing defined by (2). To avoid this problem, we calculate the T-F mask $\hat{G}^{(i,k)}_{\omega,\tau}$ and output signal $\hat{S}^{(i,k)}_{\omega,\tau}$ from the simulated output signal by using a typical sampling algorithm $\tilde{S}^{(i,k)}_{\omega,\tau}$, so that $\hat{G}^{(i,k)}_{\omega,\tau}$ and $\hat{S}^{(i,k)}_{\omega,\tau}$ satisfy the constraint of T-F-mask processing and minimize the squared error between $\hat{S}^{(i,k)}_{\omega,\tau}$ and $\tilde{S}^{(i,k)}_{\omega,\tau}$.

Figure 3 illustrates the overview of the problem and the proposed solution on the complex plane. In this study, we use the real-value T-F mask within the range of $0 \leq G_{\omega,\tau} \leq 1$. Thus, the output signal is constrained to exist on the dotted line in Fig. 3, *i.e.*, T-F mask processing affects only the norm of $\hat{S}^{(i,k)}_{\omega,\tau}$. However, since $p(\hat{\boldsymbol{S}}|\boldsymbol{X},\Theta)$ is modeled by a continuous PDF such as the complex Gaussian distribution in (11), a typical sampling algorithm possibly generates output signals that do not satisfy the T-F-mask constraint, *i.e.*, the phase spectrum of $\tilde{S}^{(i,k)}_{\omega,\tau}$ does not coincide with that of $X^{(i)}_{\omega,\tau}$. To solve this problem, we formulate the PSA-based T-F-mask recalculation. First, a temporary output signal $\tilde{S}^{(i,k)}_{\omega,\tau}$ is sampled using a sampling algorithm (Fig. 3 arrow-(i)). Then, the T-F mask $\hat{G}^{(i,k)}_{\omega,\tau}$ that minimizes the squared error between $\tilde{S}^{(i,k)}_{\omega,\tau}$ and $\hat{G}^{(i,k)}_{\omega,\tau}X^{(i)}_{\omega,\tau}$ is calculated using the PSA equation as

$$\hat{G}^{(i,k)}_{\omega,\tau} = \min\left(1, \max\left(0, \frac{|\tilde{S}^{(i,k)}_{\omega,\tau}|}{|X^{(i)}_{\omega,\tau}|}\cos\left(\theta^{(\tilde{S}^{(i,k)})}_{\omega,\tau} - \theta^{(X^{(i)})}_{\omega,\tau}\right)\right)\right), \quad (29)$$

where $\theta^{(\tilde{S}^{(i,k)})}_{\omega,\tau}$ and $\theta^{(X^{(i)})}_{\omega,\tau}$ are the phase spectra of $\tilde{S}^{(i,k)}_{\omega,\tau}$ and $X^{(i)}_{\omega,\tau}$, respectively. Then, the output signal is calculated by

$$\hat{S}^{(i,k)}_{\omega,\tau} = \hat{G}^{(i,k)}_{\omega,\tau}X^{(i)}_{\omega,\tau}, \quad (30)$$

as shown with arrow-(ii) in Fig. 3.

D. Training procedure

We describe the overall training procedure of the proposed method, as shown in Fig. 4. Hereafter, to simplify the sampling algorithm, we use the complex Gaussian distribution as $p(\hat{\boldsymbol{S}}|\boldsymbol{X},\Theta)$ described in (11)–(16).

First, the i-th observation utterance $\mathbf{X}^{(i)}$ is simulated by (1) using a randomly selected target-source file and a noise source with equal frame size from the training dataset. Next, the T-F mask $\boldsymbol{G}(\boldsymbol{x}^{(i)}_\tau)$ and variance $\boldsymbol{\sigma}(\boldsymbol{x}^{(i)}_\tau)$ are estimated by (11)–(16). Then, to simulate the k-th output signal $\hat{\mathbf{S}}^{(i,k)}$, the temporary output signal $\tilde{S}^{(i,k)'}_{\omega,\tau}$ is sampled from the complex Gaussian distribution using a pseudo random number generator, such as the Mersenne-Twister [44], as

$$\begin{bmatrix}\Re\left(\tilde{S}^{(i,k)}_{\omega,\tau}\right)\\ \Im\left(\tilde{S}^{(i,k)}_{\omega,\tau}\right)\end{bmatrix} \sim \mathcal{N}_{\mathbb{C}}\left(\hat{G}^{(i)}_{\omega,\tau}\begin{bmatrix}\Re\left(X^{(i)}_{\omega,\tau}\right)\\ \Im\left(X^{(i)}_{\omega,\tau}\right)\end{bmatrix}, \sigma^2_{\omega,\tau}\boldsymbol{I}\right), \quad (31)$$

where $\boldsymbol{I}$ is the 2×2 identity matrix, and $\Re$ and $\Im$ denote the real and imaginary parts of the complex number, respectively. After that, T-F mask $\hat{G}^{(i,k)}_{\omega,\tau}$ is calculated using (29). To accelerate the algorithm convergence, we additionally use the ϵ-greedy algorithm to calculate $\hat{G}^{(i,k)}_{\omega,\tau}$. With probability $1-\epsilon$ applied to each time-frequency bin, the maximum a posteriori (MAP) T-F mask $\hat{G}^{(i)}_{\omega,\tau}$ estimated using DNNs is used instead of $\hat{G}^{(i,k)}_{\omega,\tau}$ as

$$\hat{G}^{(i,k)}_{\omega,\tau} \leftarrow \begin{cases}\hat{G}^{(i,k)}_{\omega,\tau} & (\text{with prob. } \epsilon)\\ \hat{G}^{(i)}_{\omega,\tau} & (\text{otherwise})\end{cases}. \quad (32)$$

In addition, a large gradient value $\partial_\Theta \mathcal{J}(\Theta)$ leads to unstable training. One reason for the large gradient is that the log-likelihood $\partial_\Theta \ln p(\hat{\boldsymbol{S}}^{(i,k)}_\tau|\boldsymbol{X}^{(i)}_\tau,\Theta)$ in (26) becomes large. To reduce the gradient of the log-likelihood, the difference between the mean T-F mask $\hat{G}^{(i)}_{\omega,\tau}$ and simulated T-F mask $\hat{G}^{(i,k)}_{\omega,\tau}$ is truncated to confine it within the range of $[-\lambda,\lambda]$ as

$$\Delta\hat{G}^{(i,k)}_{\omega,\tau} \leftarrow \hat{G}^{(i,k)}_{\omega,\tau} - \hat{G}^{(i)}_{\omega,\tau} \quad (33)$$

$$\Delta\hat{G}^{(i,k)}_{\omega,\tau} \leftarrow \begin{cases}\lambda & (\Delta\hat{G}^{(i,k)}_{\omega,\tau} > \lambda)\\ \Delta\hat{G}^{(i,k)}_{\omega,\tau} & (-\lambda \leq \Delta\hat{G}^{(i,k)}_{\omega,\tau} \leq \lambda)\\ -\lambda & (\Delta\hat{G}^{(i,k)}_{\omega,\tau} < -\lambda)\end{cases}, \quad (34)$$

$$\hat{G}^{(i,k)}_{\omega,\tau} \leftarrow \hat{G}^{(i)}_{\omega,\tau} + \Delta\hat{G}^{(i,k)}_{\omega,\tau}. \quad (35)$$

Then, the output signal $\hat{\mathbf{S}}^{(i,k)}$ is calculated by T-F-mask processing (30), and the OSQA scores $\mathcal{Z}(\hat{\mathbf{S}}^{(i,k)},\mathbf{X}^{(i)})$ and $\mathcal{B}(\hat{\mathbf{S}}^{(i,k)},\mathbf{X}^{(i)})$ are calculated by (28). After applying these procedures for $\mathcal{I}$ utterances, Θ is updated using the back-propagation algorithm using the gradient calculated by (25).

IV. EXPERIMENTS

We conducted objective experiments to evaluate the performance of the proposed method. The experimental conditions are described in Sec. IV-A. To investigate whether a DNN source-enhancement function can be trained to increase OSQA scores, we first investigated the relationship between the number of updates and OSQA scores (Sec. IV-B). Second, the source enhancement performance of the proposed method was compared with those of conventional methods by using several objective measurements (Sec. IV-C). Finally, subjective evaluations for sound quality and ineligibility were conducted (Sec. IV-D). For comparison methods, we used four DNN source-enhancement methods; two T-F-mask mapping functions trained using an MMSE-based objective function [19] and the ML-based objective function described in Sec. II-B, and two T-F-mask selection functions trained for increasing the PESQ and STOI [39].

A. Experimental conditions

1) Dataset: The ATR Japanese speech database [45] was used as the training dataset of the target source. The dataset consists of 6640 utterances spoken by 11 males and 11 females. The utterances were randomly separated into 5976 for the development set and 664 for the validation set. As the training dataset of noise, a noise dataset of CHiME-3 was used that consisted of four types of background noise files including noise in *cafes*, *street junctions*, *public transport*, and *pedestrian areas* [46]. The noisy-mixture dataset was generated by mixing clean speech utterances with various noisy and SNR conditions using the following procedure; i) the noise is randomly selected from noise dataset, ii) the amplitude of noise is adjusted to be the desired SNR-level, and iii) the speech and noise source is added in the time-domain. As the test dataset, a Japanese speech database consisting of 300 utterances spoken by 3 males and 3 females was used for target-source dataset, and an ambient noise database recorded at *airports* (Airp.), *amusement parks* (Amuse.), *offices* (Office),

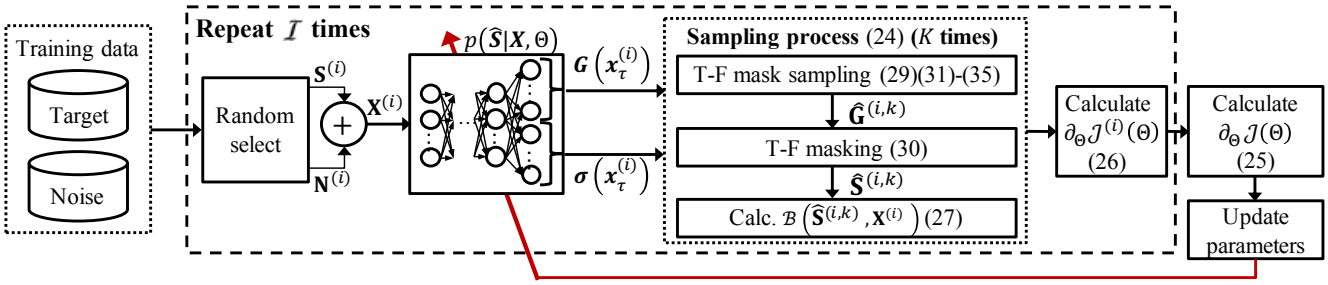


Fig. 4. Training procedure of proposed method

TABLE I
EXPERIMENTAL CONDITIONS

Parameters for signal processing	
Sampling rate	16.0 kHz
FFT length	512 pts
FFT shift length	256 pts
# of mel-filterbanks	64
Smoothing parameter β	0.3
Lower threshold $G^{\min}$	0.158 (= −16 dB)
Training SNR (dB)	-6, 0, 6, 12
DNN architecture	
# of hidden layers for DNNs	3
# of hidden units for DNNs	1024
Activation function (T-F mask, ϕ_g)	sigmoid
Activation function (variance, ϕ_σ)	exponential
Activation function (hidden, ϕ_h)	ReLU
Context window size Q	5
Variance regularization parameter C_σ	10^{-4}
Parameters for MMSE and ML-based DNN training	
Initial step-size	10^{-4}
Step-size threshold for early-stopping	10^{-7}
Dropout probability (input layer)	0.2
Dropout probability (hidden layer)	0.5
L_2 normalization parameter	10^{-4}
Parameters for T-F mask selection	
# of T-F mask templates	128
ϵ-greedy parameter ϵ	0.01
Parameters for proposed DNN training	
Step-size	10^{-6}
# of utterance $\mathcal{I}$	10
# of T-F mask sampling K	20
Clipping parameter λ	0.05
ϵ-greedy parameter ϵ	0.05

and *party rooms* (Party) was used as the noisy dataset. All samples were recorded at the sampling rate of 16 kHz. The SNR levels of the training/test dataset were -6, 0, 6, and 12 dB.

2) DNN architecture and setup: For the proposed and all conventional methods, a fully connected DNN was used that has 3 hidden layers and 1024 hidden units. All input vectors were mean-and-variance normalized using the training data statistics. The activation functions for the T-F mask ϕ_g, variance ϕ_σ, and hidden units ϕ_h were the sigmoid function, exponential function, and rectified linear unit (ReLU), respectively. The context window size was $Q = 5$, and the variance regularization parameter in (15) was $C_\sigma = 10^{-41}$. The Adam method [47] was used as a gradient method. To avoid overfitting, input vectors and DNN outputs, *i.e.*, the T-F masks and error variances, were compressed using a $\mathsf{B} = 64$ Mel-transformation matrix, and the estimated T-F masks and error variances were transformed into a linear frequency domain using the Mel-transform's pseudo-inverse [48].

A PSA objective function [19], [20] was used as the MMSE-based objective function. Since the PSA objective function does not use the variance parameter $\boldsymbol{\sigma}(\boldsymbol{x}_\tau)$, DNNs estimate only T-F masks $\boldsymbol{G}(\boldsymbol{x}_\tau)$. For the ML-based objective function, we used (9) with the complex Gaussian distribution described in Sec. II-B. To train both methods, the dropout algorithm was used and initialized by layer-by-layer pre-training [49]. An early-stopping algorithm [17] was used for fine-tuning with the initial step-size 10^{-4} and the step-size threshold 10^{-7}, and L2 normalization with the parameter 10^{-4} was used as a regularization algorithm.

For the T-F-mask selection-based method [39], to improve the flexibility of T-F-mask selection, we used 128 T-F-mask templates. The DNN architecture, except for the output layer, is the same as MMSE- and ML-based methods.

For the proposed method, DNN parameters were initialized by ML-based training, and their step-size was 10^{-6}. To calculate $\partial_\Theta \mathcal{J}(\Theta)$, the iteration parameters $\mathcal{I} = 10$ and $K = 20$ were used. The ϵ-greedy parameter ϵ was 0.05, and the clipping parameter λ was determined as 0.05 according to preliminary informal experiments[2]. As the OSQA scores, we used the PSEQ, which is a speech quality measure, and the STOI, which is a speech intelligibility measure. To avoid adjusting the step-size of the gradient method for each OSQA, we normalized OSQA scores to uniform the range of the each OSQA score. In this experiments, each OSQA score was normalized so that its maximum and minimum values were 100 and 0 as

$$\mathcal{Z}^{\text{PESQ}}(\hat{\mathbf{S}},\mathbf{X}) = 20.0 \times \left(\text{PESQ}(\hat{\mathbf{S}},\mathbf{X}) + 0.5\right),$$
$$\mathcal{Z}^{\text{STOI}}(\hat{\mathbf{S}},\mathbf{X}) = 100.0 \times \text{STOI}(\hat{\mathbf{S}},\mathbf{X}).$$

[1]In preliminary experiments using candidate values $C_\sigma \in \{10^{-2}, 10^{-3}, 10^{-4}\}$, there were no distinct differences in training stability and results. Thus, to eliminate the effect of regularization, we used the minimum parameter of the candidate values.

[2]We tested some possible combinations of these parameters by grid-search. Then, we found that the listed parameters achieved a stable training and realistic computational time (2 days using an Intel Xeon Processor E5-2630 v3 CPU and a Tesla M-40 GPU).

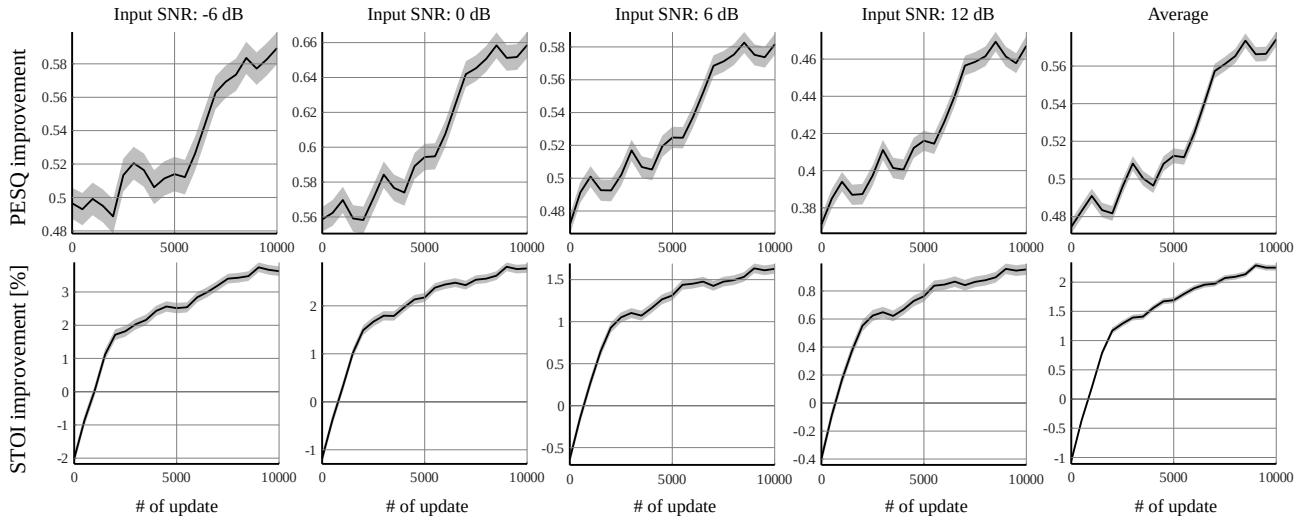


Fig. 5. OSQA score improvement depending on number of updates. X-axis shows number of updates, and y-axis shows average difference between OSQA score of proposed method and that of observed signal. Solid lines and gray area are average and standard-error, respectively.

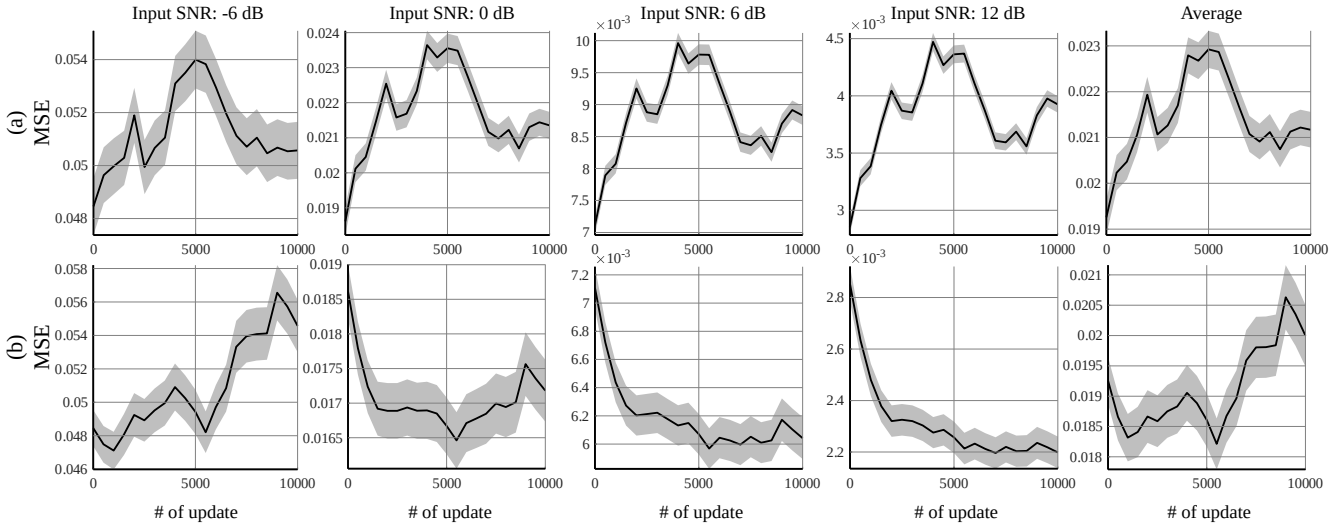


Fig. 6. Mean squared error (MSE) depending on number of updates. OSQA scores used for training of proposed method were (a) PESQ and (b) STOI. X-axis shows number of updates, and y-axis shows MSE. Solid lines and gray area are average and standard-error, respectively.

The training algorithm was stopped after 10,000 times of executing the whole parameter update process shown in Fig. 4.

3) Other conditions: It is known that T-F-mask processing causes artificial distortion, so-called musical noise [50]. For all methods, to reduce musical noise, flooring [6], [51] and smoothing [52], [53] were applied to $\hat{G}_{\omega,\tau}$ before T-F-mask processing as

$$\hat{G}_{\omega,\tau} \leftarrow \max\left(G^{\min}, \hat{G}_{\omega,\tau}\right), \tag{36}$$

$$\hat{G}_{\omega,\tau} \leftarrow \beta\hat{G}_{\omega,\tau} + (1-\beta)\hat{G}_{\omega,\tau-1}, \tag{37}$$

where we used the lower threshold of the T-F mask $G^{\min} = 0.158$ and smoothing parameter $\beta = 0.3$. The frame size of the short-time Fourier transform (STFT) was 512, and the frame was shifted by 256 samples. All the above-mentioned conditions are summarized in Table I.

TABLE II
CORRELATION COEFFICIENTS BETWEEN MSE AND OSQA SCORE IMPROVEMENTS

	-6 dB	0 dB	6 dB	12 dB	Average
PESQ	−0.120	−0.081	0.020	0.089	−0.020
STOI	0.756	−0.672	−0.951	−0.980	0.482

B. Investigation of relationship between number of updates and OSQA score

To investigate whether the DNN source-enhancement function can be trained to increase OSQA scores, we first investigated the relationship between the number of updates and improvement of the OSQA scores. We define "OSQA score improvement" as the difference in the score value from the baseline OSQA score. For the baseline, we use the OSQA score obtained from the observed signal. Since the DNN parameters of the proposed method were initialized by ML-based training, each OSQA score was compared with the OSQA

TABLE III
EVALUATION RESULTS ON THREE OBJECTIVE MEASUREMENTS. ASTERISKS INDICATE SCORES SIGNIFICANTLY HIGHER THAN THAT OF MMSE AND ML IN PAIRED ONE-SIDED T-TEST. GRAY CELLS INDICATE THE HIGHEST SCORE IN SAME NOISE AND INPUT SNR CONDITION.

Input SNR: -6 dB

	SDR [dB]					PESQ					STOI [%]				
Method	Airp.	Amuse.	Office	Party	Ave.	Airp.	Amuse.	Office	Party	Ave.	Airp.	Amuse.	Office	Party	Ave.
OBS	−4.28	−6.98	−5.64	−1.50	−4.6	1.24	1.38	1.33	1.14	1.27	72.1	76.7	73.8	69.1	72.9
MMSE	3.22	5.87	4.66	3.77	4.38	1.66	1.89	1.80	1.48	1.71	68.9	73.6	71.0	66.7	70.1
ML	**3.31**	6.12	**4.87**	3.63	**4.48**	1.68	1.95	1.80	1.54	1.74	69.2	74.3	72.0	64.9	70.1
C-PESQ	−0.28	1.38	−0.03	1.67	0.69	1.55	1.77	1.64	1.44	1.60	∗72.2	∗76.4	∗73.4	∗70.4	∗73.2
C-STOI	0.21	2.02	0.68	2.17	1.27	1.48	1.64	1.56	1.34	1.50	**∗75.0**	∗79.8	**∗76.6**	∗71.1	∗75.6
P-PESQ	3.13	∗6.34	4.72	3.50	4.42	**∗1.78**	∗2.07	**∗1.91**	∗1.57	**∗1.83**	∗71.0	∗76.0	∗72.4	∗67.9	∗71.8
P-STOI	2.18	**∗6.60**	3.90	**∗4.15**	4.21	1.63	1.93	1.73	**∗1.59**	1.72	∗74.9	**∗80.1**	**∗76.6**	**∗71.3**	**∗75.7**
P-MIX	2.93	∗6.20	4.39	3.49	4.25	∗1.77	**∗2.08**	∗1.89	**∗1.59**	**∗1.83**	∗72.1	∗77.4	∗73.8	∗68.2	∗72.9

Input SNR: 0 dB

	SDR [dB]					PESQ					STOI [%]				
Method	Airp.	Amuse.	Office	Party	Ave.	Airp.	Amuse.	Office	Party	Ave.	Airp.	Amuse.	Office	Party	Ave.
OBS	1.67	−1.19	0.36	4.46	1.32	1.71	1.88	1.81	1.54	1.73	84.5	87.8	85.2	82.9	85.1
MMSE	8.03	10.0	9.55	8.44	9.00	2.17	2.36	2.27	2.09	2.22	80.7	84.7	83.1	80.1	82.1
ML	**8.62**	10.4	**9.97**	8.66	9.40	2.20	2.42	2.30	2.14	2.27	82.5	86.4	84.6	79.6	83.3
C-PESQ	6.36	7.08	6.49	7.89	6.95	2.11	2.33	2.23	2.00	2.16	∗83.7	86.2	84.0	∗82.7	∗84.2
C-STOI	7.30	8.07	7.18	8.70	7.81	2.03	2.18	2.10	1.89	2.05	**∗86.8**	∗89.9	∗87.4	∗84.7	∗87.2
P-PESQ	8.40	10.3	9.77	8.28	9.19	∗2.30	∗2.55	**∗2.41**	∗2.20	∗2.37	∗82.7	86.4	84.1	∗80.3	∗83.4
P-STOI	8.45	**∗11.2**	9.52	**∗9.74**	**∗9.74**	2.12	2.36	2.21	2.11	2.20	∗86.7	**∗90.0**	**∗87.5**	**∗85.0**	**∗87.3**
P-MIX	8.09	9.85	9.12	8.11	8.79	**∗2.31**	**∗2.57**	∗2.41	**∗2.23**	**∗2.38**	∗84.2	∗87.8	∗85.5	∗81.6	∗84.7

Input SNR: 6 dB

	SDR [dB]					PESQ					STOI [%]				
Method	Airp.	Amuse.	Office	Party	Ave.	Airp.	Amuse.	Office	Party	Ave.	Airp.	Amuse.	Office	Party	Ave.
OBS	7.67	4.96	6.29	10.5	7.34	2.18	2.33	2.28	2.02	2.20	92.2	93.8	92.7	91.8	92.6
MMSE	12.1	13.6	13.4	12.6	12.9	2.54	2.68	2.63	2.49	2.58	88.9	91.2	90.4	88.6	89.8
ML	13.1	14.2	14.1	13.5	13.7	2.59	2.77	2.69	2.54	2.65	91.1	93.0	92.2	89.8	91.5
C-PESQ	11.5	11.9	11.4	12.6	11.9	2.54	2.75	2.69	2.45	2.61	90.5	91.8	90.9	89.9	90.8
C-STOI	13.2	13.6	13.1	14.3	13.5	2.50	2.62	2.57	2.38	2.52	**∗93.4**	∗94.8	∗93.9	**∗92.8**	**∗93.8**
P-PESQ	12.6	13.8	13.6	12.6	13.2	**∗2.70**	∗2.89	**∗2.80**	∗2.64	**∗2.76**	90.2	92.1	91.2	89.1	90.6
P-STOI	**∗13.4**	**∗15.3**	**∗14.3**	**∗14.8**	**∗14.4**	2.49	2.69	2.60	2.45	2.56	**∗93.4**	**∗94.9**	**∗94.0**	**∗92.8**	**∗93.8**
P-MIX	11.5	12.3	12.1	11.6	11.9	∗2.69	**∗2.90**	∗2.79	**∗2.66**	**∗2.76**	∗91.5	∗93.1	∗92.3	∗90.4	∗91.8

Input SNR: 12 dB

	SDR [dB]					PESQ					STOI [%]				
Method	Airp.	Amuse.	Office	Party	Ave.	Airp.	Amuse.	Office	Party	Ave.	Airp.	Amuse.	Office	Party	Ave.
OBS	13.6	11.0	12.3	16.4	13.3	2.61	2.76	2.72	2.47	2.64	96.1	96.9	96.4	96.2	96.4
MMSE	15.9	16.9	16.8	16.3	16.5	2.84	2.95	2.92	2.77	2.87	93.5	94.7	94.4	93.2	94.0
ML	17.5	18.0	18.0	18.1	17.9	2.95	3.09	3.03	2.88	2.98	95.5	96.3	96.0	94.9	95.7
C-PESQ	15.5	15.8	15.3	16.3	15.7	2.95	∗3.14	∗3.08	2.86	∗3.01	94.2	94.9	94.4	94.0	94.4
C-STOI	**∗18.2**	∗18.6	∗18.2	∗19.0	∗18.5	2.94	3.05	3.01	2.81	2.95	∗96.7	∗97.4	∗97.0	∗96.6	∗96.9
P-PESQ	16.5	17.2	17.1	16.6	16.8	**∗3.04**	**∗3.19**	**∗3.12**	**∗2.97**	**∗3.08**	94.4	95.2	94.9	93.8	94.6
P-STOI	**∗18.2**	**∗19.5**	**∗18.8**	**∗19.7**	**∗19.1**	2.85	3.02	2.96	2.78	2.90	**∗96.8**	**∗97.5**	**∗97.1**	**∗96.7**	**∗97.0**
P-MIX	13.6	13.9	13.9	13.8	13.8	∗3.01	∗3.18	∗3.10	**∗2.97**	∗3.07	95.3	96.0	95.7	94.7	95.4

score that had zero updates. Thus, if DNN parameters were successfully trained with the proposed method, the OSQA score improvement would increase in accordance with the number of updates.

Figure 5 shows the OSQA score improvements evaluated on the test dataset. Both OSQA score improvements increased as the number of updates increased for all SNR conditions. These results suggest that the proposed method is effective at increasing arbitrary OSQA scores, such as the PESQ and STOI.

We also investigated the relationship between the number of updates and MSE using the test dataset. Figure 6 shows MSE depending on the number of updates. Under most SNR conditions, MSE did not decrease despite OSQA scores increasing. Table II shows the correlation coefficients between OSQA score improvements and MSE values. There was little correlation between PESQ improvement and MSE, and the correlation between STOI improvement and MSE depended on the input SNR condition. Thus, these results suggest that minimization of MSE does not necessarily maximize OSQA scores.

TABLE IV
OBJECTIVE SCORES OF EXAMPLE RESULTS SHOWN IN FIG. 7.

	Performance measurement		
Method	SDR [dB]	PESQ	STOI [%]
OBS	2.36	1.79	81.5
MMSE	9.31	2.32	80.0
ML	**11.3**	2.48	82.1
P-PESQ	10.7	**2.55**	81.4
P-STOI	11.2	2.40	**86.3**
P-MIX	11.2	**2.55**	83.4

C. Objective evaluation

The source-enhancement performance of the proposed method was compared with those of conventional methods

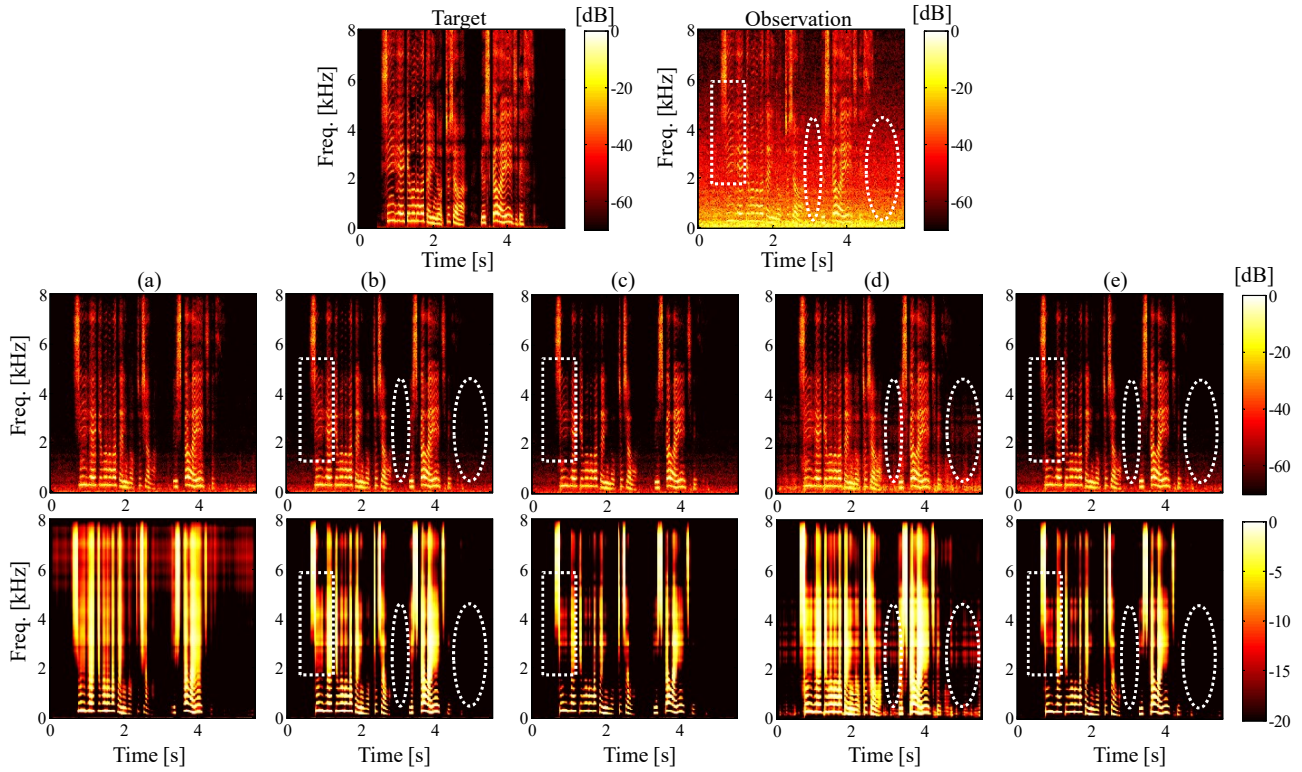


Fig. 7. Examples of estimated T-F mask and output signal. Top figures show spectrogram of target source $S_{\omega,\tau}$ (left) and observed signal $X_{\omega,\tau}$ (right), respectively. Middle figures show spectrogram of output signal $\hat{S}_{\omega,\tau}$ and bottom figures show estimated T-F mask $\hat{G}_{\omega,\tau}$, respectively. White dotted box and circle show larger or less noise reduction areas which modified by training of `P-PESQ` and `P-STOI`, respectively. (a) `MMSE`, (b) `ML`, (c) `P-PESQ`, (d) `P-STOI`, and (e) `P-MIX`.

using three objective measurements: the signal-to-distortion ratio (SDR), PESQ, and STOI. The SDR was defined as

$$\text{SDR [dB]} := 10\log_{10}\frac{\sum_{\tau=1}^{T}\sum_{\omega=1}^{\Omega}|S_{\omega,\tau}|^2}{\sum_{\tau=1}^{T}\sum_{\omega=1}^{\Omega}|S_{\omega,\tau}-\hat{S}_{\omega,\tau}|^2}, \tag{38}$$

and calculated using the "BSS-Eval toolbox [54]." These measurements were evaluated on the observed signal (`OBS`), the MMSE- and ML-based DNN training (`MMSE` and `ML`), a T-F-mask selection method to increase the PESQ and STOI [39] (`C-PESQ` and `C-STOI`), and the proposed method to increase the PESQ and STOI (`P-PESQ` and `P-STOI`). To investigate whether the proposed method enables training of a DNN to increase a metric that consists of multiple OSQA scores, we also trained a DNN to increase a mixed-OSQA score (`P-MIX`). As the first trial, we mixed the PESQ and the STOI. The mixed-OSQA is defined as

$$\mathcal{Z}^{\text{MIX}}(\hat{\mathbf{S}},\mathbf{X}) = \gamma\mathcal{Z}^{\text{PESQ}}(\hat{\mathbf{S}},\mathbf{X}) + (1-\gamma)\,\mathcal{Z}^{\text{STOI}}(\hat{\mathbf{S}},\mathbf{X}).$$

In this trial, in order to confirm whether multiple OSQA scores increase simultaneously, the additive coefficient $\gamma = 0.5$ was determined in such a way that both OSQA scores had the same contribution to $\mathcal{Z}^{\text{MIX}}(\hat{\mathbf{S}},\mathbf{X})$.

Table III lists the evaluation results of each objective measurement on four noise types and four input SNR conditions. The asterisk indicates that the score was significantly higher than both `MMSE` and `ML` in a paired one-sided t-test ($\alpha = 0.05$). The SDRs tended to be higher when using the conventional MMSE/ML-based objective function than the proposed method under low SNR conditions. The PESQ and STOI of `P-PESQ` and `P-STOI` were higher than those of `MMSE` and `ML`, respectively. For each method, the PESQ and STOI improved by around 0.1 and 2–5 %, respectively, and significant differences were observed for all noise and SNR conditions. These results suggest that the proposed method was able to train the DNN source-enhancement function to directly increase black-box OSQA scores.

In mixed-OSQA experiments, both PESQ and STOI of `P-MIX` were higher than those of `MMSE` and `ML` under almost all noise and SNR conditions. In the comparison to the results of the mixed-OSQA and single-OSQA (*i.e.* `P-PESQ` and `P-STOI`), `P-MIX` achieved almost the same or slightly lower PESQ and STOI scores than `P-PESQ` and `P-STOI`, respectively. In addition, `P-MIX` outperformed STOI and PESQ scores than `P-PESQ` and `P-STOI`, respectively. These results suggest that the use of the mixed-OSQA would be an effective way to increase multiple-perceptual qualities.

In Table III we also show that the proposed method outperformed the T-F mask selection-based methods [39] in terms of the target OSQA under almost all noise types and SNR conditions. Such favorable experimental results would have been observed because of the flexibility of the T-F mask estimation achieved by the proposed method. In this experiment, the number of the T-F mask template (= 128) was larger than that used in the previous work (= 32) [39]. However, since the T-F masks were generated by a combination of the finite number of templates, the patterns of the T-F mask were still

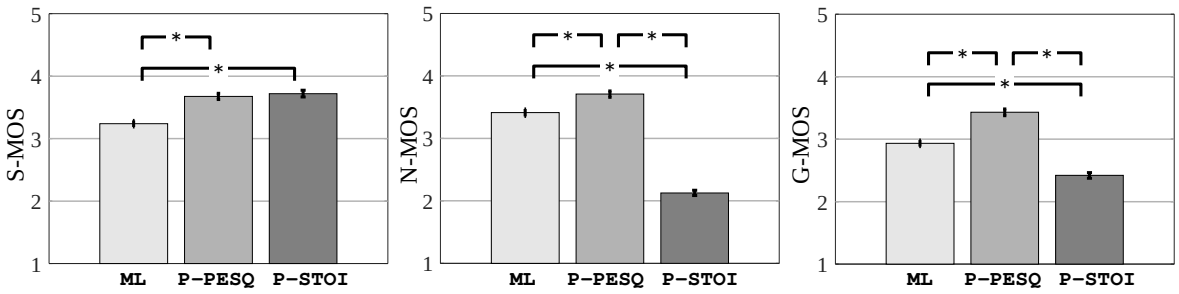


Fig. 8. Evaluation results of sound-quality test according to ITU-T P.835. Bar graphs and error bar indicate average and standard error, respectively. Asterisks indicate significant difference observed in paired one-sided t-test.

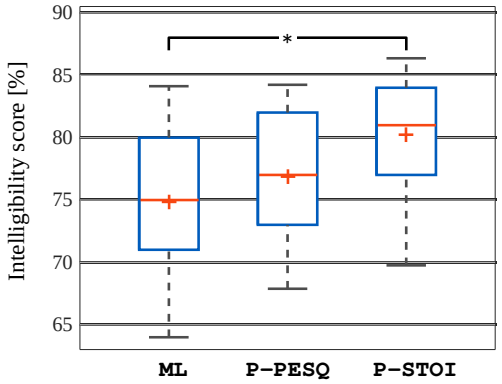


Fig. 9. Evaluation results of word-intelligibility test. Asterisks indicate significant difference observed in unpaired one-sided t-test.

limited. These results suggested that by adopting the policy-gradient method to optimize the parameters of a continuous PDF of the T-F mask processing, the flexibility of the T-F mask estimation was improved.

Figure 7 shows examples of the estimated T-F masks and output signal, and Table IV lists its objective scores. The SNR of the observed signal was adjusted to 0 dB using *amusement parks* noise. Figure 7 shows that the estimated T-F masks reflect the characteristics of each objective function. In comparison to the results of `MMSE` and `ML` that reduced the distortion of the target source on average, the T-F mask estimated by `P-PESQ` strongly reduced the residual noise, even when it distorted the target sound at a middle/high frequency (*e.g.* Fig. 7 white dotted box), and achieved the best PESQ. In contrast, the T-F mask estimated by `P-STOI` weakly reduced noise to avoid distorting the target source, even when the noise remained in the non-speech frames (*e.g.* Fig. 7 white dotted circle), and achieved the best STOI. This may be because the residual noise degrades the sound quality and the distortion of the target source degrades speech intelligibility. The T-F mask estimated by `P-MIX` involved both characteristics and relaxed the disadvantage of `P-PESQ` and `P-STOI`, and both OSQA scores were higher than those of `ML` and `MMSE`. Namely, speech distortion at a middle/high frequency was reduced (*e.g.* Fig. 7 white dotted box) and residual noise in the non-speech frames were reduced (*e.g.* Fig. 7 white dotted circle).

D. Subjective evaluation

1) Sound quality evaluation: To investigate the sound quality of the output signals, subjective speech-quality tests were conducted according to ITU-T P.835 [55]. In the tests, the participants rated three different factors in the samples:

- Speech mean-opinion-score (S-MOS): the speech sample was rated 5–not distorted, 4–slightly distorted, 3–somewhat distorted, 2–fairly distorted, or 1–very distorted.
- Subjective noise MOS (N-MOS): the background of the sample was 5–not noticeable, 4–slightly noticeable, 3–noticeable but not intrusive, 2–somewhat intrusive, or 1–very intrusive.
- Overall MOS (G-MOS): the sound quality of the sample was 5–excellent, 4–good, 3–fair, 2–poor, or 1–bad.

Sixteen participants evaluated the sound quality of the output signals of `ML`, `P-PESQ`, and `P-STOI`. The participants evaluated 20 files for each method; the 20 files consisted of five randomly selected files from the test dataset for each of the four types of noise. The input SNR was 6 dB.

Figure 8 shows the results of the subjective tests. For all factors, `P-PESQ` achieved a higher score than `ML`, and statistically significant differences from `ML` were observed in a paired one-sided t-test (p-value = 0.05). The reason for this result suggested that participants may have perceived the degrade of the speech quality from both the speech distortion and the residual noise in speech frame in the output signal of `ML`. In addition, although there was no statistically significant difference between `P-PESQ` and `P-STOI` in terms of S-MOS score, N-MOS score of `P-STOI` was significantly lower than that of `P-PESQ`. Thus, G-MOS score of `P-STOI` was also lower than that of `P-PESQ`. It would be because `P-STOI` weakly reduced noise to avoid distorting the target source, even when the noise remained in the non-speech frames as shown in Sec. IV.C.

2) Speech intelligibility test: We conducted a word-intelligibility test to investigate speech intelligibility. We selected 50 low familiarity words from familiarity-controlled word lists 2003 (FW03) [56] as the test dataset of speech. The selected dataset consisted of Japanese four-mora words whose accent type was Low-High-High-High. The noisy test dataset was created by adding a randomly selected noise at SNR of 6 dB from the noisy dataset, which was used in the objective evaluation. Sixteen participants attempted to write a

phonetic transcription for output signals of ML, P-PESQ, and P-STOI. The percentage of correct answers was used as the intelligibility score.

Figure 9 shows the intelligibility score of each method. P-STOI achieved the highest score. In addition, statistically significant differences from ML were observed in an unpaired one-sided t-test (p-value = 0.05). From both sound-quality and speech-intelligibility tests, we found that the proposed method could improve the specific hearing quality corresponding to the OSQA score used as the objective function.

V. CONCLUSIONS

We proposed a training method for the DNN-based source-enhancement function to increase OSQA scores such as the PESQ. The difficulty is that the gradient of OSQA scores may not be analytically calculated by simply applying the back-propagation algorithm because most OSQA scores are black boxes. To calculate the gradient of the OSQA-based objective function, we formulated a DNN-optimization scheme on the basis of the policy-gradient method. In the experiment, 1) it was revealed that the DNN-based source-enhancement function can be trained using the gradient of the OSQA obtained with the policy-gradient method. In addition, 2) the OSQA score and specific hearing quality corresponding to the OSQA score used as the objective function improved. Therefore, it can be concluded that this method made it possible to use not only analytical objective functions but also black-box functions for the training of the DNN-based source-enhancement function.

Although we focused on maximization of OSQA in this study, the proposed method potentially increases other black-box measurements. In the future, we will aim to adopt the proposed method to increase other black-box objective measures such as the subjective score obtained from a "human-in-the-loop" audio-system [57] and word accuracy of a black-box automatic-speech-recognition system [58]. We found that both the PESQ and STOI could increase simultaneously by mixing multiple OSQA scores as an objective function. In the future, we will also investigate the optimality of the OSQA score and its mixing ratio for the proposed method.

REFERENCES

[1] J. Benesty, S. Makino, and J. Chen, Eds., "Speech enhancement," Springer, 2005.

[2] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator," *IEEE Trans. Audio, Speech and Language Processing*, pp.1109–1121, 1984.

[3] R. Zelinski "A microphone array with adaptive post-filtering for noise reduction in reverberant rooms," in *Proc. ICASSP*, pp. 2578 –2581, 1988.

[4] Y. Hioka, K. Furuya, K. Kobayashi, K. Niwa and Y. Haneda, "Underdetermined sound source separation using power spectrum density estimated by combination of directivity gain," *IEEE Trans. Audio, Speech and Language Processing*, pp.1240–1250, 2013.

[5] K. Niwa, Y. Hioka, and K. Kobayashi, "Optimal Microphone Array Observation for Clear Recording of Distant Sound Sources," *IEEE/ACM Trans. Audio, Speech and Language Processing*, pp.1785–1795, 2016.

[6] L. Lightburn, E. D. Sena, A. Moore, P. A. Naylor, M. Brookes, "Improving the perceptual quality of ideal binary masked speech," in *Proc. ICASSP*, 2017.

[7] T. Yoshioka, A. Sehr, M. Delcroix, K. Kinoshita, R. Maas, T. Nakatani, and W. Kellermann, "Making machines understand us in reverberant rooms: robustness against reverberation for automatic speech recognition," *IEEE Signal Processing Magazine*, pp. 114–126, 2012.

[8] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *Proc. ICASSP*, 2013.

[9] T. Ochiai, S. Watanabe, T. Hori, and J. R. Hershey, "Multichannel End-to-end Speech Recognition," in *Proc. ICML*, 2017.

[10] K. Kobayashi, Y. Haneda, K. Furuya, and A. Kataoka, "A hands-free unit with noise reduction by using adaptive beamformer," *IEEE Trans. on Consumer Electronics*, Vol.54-1, 2008.

[11] Y. Hioka, K. Furuya, K. Kobayashi, S. Sakauchi, and Y. Haneda, "Angular region-wise speech enhancement for hands-free speakerphone," *IEEE Trans. on Consumer Electronics*, Vol.58-4, 2012.

[12] B. C. J. Moore, "Speech processing for the hearing-impaired: successes, failures, and implications for speech mechanisms," *Speech Communication*, Vol. 41, Issue 1, pp.81–91, 2003.

[13] D. L. Wang, "Time-frequency masking for speech separation and its potential for hearing aid design," *Trends in Amplification*, vol. 12, pp. 332–353, 2008.

[14] T. Zhang, F. Mustiere, and C. Micheyl, "Intelligent Hearing Aids: The Next Revolution," In *Proc. EMBC,* 2016.

[15] Y. Zhao, D. Wang, I. Merks, and T. Zhang, "DNN-based enhancement of noisy and reverberant speech," In *Proc. ICASSP*, 2016.

[16] R. Oldfield, B. Shirley and J. Spille, "Object-based audio for interactive football broadcast," *Multimedia Tools and Applications*, Vol. 74, pp.2717–2741, 2015.

[17] Y. Koizumi, K. Niwa, Y. Hioka, K. Kobayashi and H. Ohmuro, "Informative acoustic feature selection to maximize mutual information for collecting target sources," *IEEE/ACM Trans. Audio, Speech and Language Processing*, pp.768–779, 2017.

[18] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, 521, pp.436–444, 2015.

[19] F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. L. Roux, J. R. Hershey, and B. Schuller, "Speech Enhancement with LSTM Recurrent Neural Networks and its Application to Noise-Robust ASR," in *Proc. LVA/ICA*, 2015.

[20] H. Erdogan, J. R. Hershey, S. Watanabe, and J. L. Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. ICASSP*, 2015.

[21] D. S. Williamson and D. L. Wang, "Time-frequency masking in the complex domain for speech dereverberation and denoising," *IEEE/ACM Trans. Audio, Speech and Language Processing*, 2017.

[22] Y. Zhao, B. Xu, R. Giri, and T. Zhang, "Perceptually Guided Speech Enhancement using deep neural networks," in *Proc. ICASSP*, 2018.

[23] Y. Xu, J. Du, L. R. Dai, and C. H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Processing Letters*, pp.65–68, 2014.

[24] Y. Xu, J. Du, L. R. Dai and C. H. Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Trans. Audio, Speech and Language Processing*, pp.7–19, 2015.

[25] Y. Xu, J. Du, Z. Huang, L. R. Dai, and C. H. Lee, "Multi-objective learning and mask-based post-processing for deep neural network based speech enhancement," in *Proc. INTERSPEECH*, 2015.

[26] T. Gao, J. Du, L. R. Dai, and C. H. Lee, "SNR-Based Progressive Learning of Deep Neural Network for Speech Enhancement," in *Proc. INTERSPEECH*, 2016.

[27] Q. Wang, J. Du, L. R. Dai and C. H. Lee, "A multiobjective learning and ensembling approach to high-performance speech enhancement with compact neural network architectures," *IEEE/ACM Trans. Audio, Speech and Language Processing*, pp.1181–1193, 2018.

[28] T. Kawase, K. Niwa, K. Kobayashi, and Y. Hioka, "Application of neural network to source PSD estimation for Wiener filter based sound source separation," in *Proc. IWAENC*, 2016.

[29] K. Niwa, Y. Koizumi, T. Kawase, K. Kobayashi and Y. Hioka, "Supervised Source Enhancement Composed of Non-negative Auto-Encoders and Complementarity Subtraction" in *Proc. ICASSP*, 2017.

[30] P. Smaragdis and S. Venkataramani, "A Neural Network Alternative to Non-Negative Audio Models," in *Proc. ICASSP*, 2017.

[31] L. Chai, J. Du and Y. Wang, "Gaussian Density Guided Deep Neural Network For Single-Channel Speech Enhancement," in *Proc. MLSP*, 2017.

[32] K. Kinoshita, M. Delcroix, A. Ogawa, T. Higuchi, and T. Nakatani, "Deep Mixture Density Network for Statistical Model-based Feature Enhancement," in *Proc. ICASSP*, 2017.

[33] A. A. Nugraha, A. Liutkus, and E. Vincent, "Multichannel Audio Source Separation With Deep Neural Networks," *IEEE/ACM Trans. Audio, Speech and Language Processing*, 2016.

[34] J. Hershy, Z. Chen, J. L. Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," In *Proc. ICASSP*, 2016.

[35] S. Pascual, A. Bonafonte, and J. Serra, "SEGAN: Speech Enhancement Generative Adversarial Network," In *Proc INTERSPEECH*, 2017.

[36] D. E. Rumelhart, G. E. Hinton, E. Geoffrey and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, 323, pp.533–536, 1986.

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

[38] 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, pp.2125–2136, 2011.

[39] Y. Koizumi, K. Niwa, Y. Hioka, K. Kobayashi and Y. Haneda, 'DNN-based Source Enhancement Self-optimized by Reinforcement Learning using Sound Quality Measurements," in *Proc. ICASSP*, 2017.

[40] E. S. Sutton and A. G. Barto, "Reinforcement Learning: An Introduction," *A Bradford Book,* 1998.

[41] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel and D. Hassabis, " Mastering the game of Go with deep neural networks and tree search," *Nature*, pp.484—489, 2016.

[42] R. J. Williams, "Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning," *Machine Learning*, Vol. 8, 1992.

[43] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, "Policy Gradient Methods for Reinforcement Learning with Function Approximation," In *Proc. NIPS*, 1999.

[44] M. Matsumoto and T. Nishimura, "Mersenne Twister: A 623-dimensionally Equidistributed Uniform Pseudorandom Number Generator," ACM Trans. on Modeling and Computer Simulations, 1998.

[45] A. Kurematsu, K. Takeda, Y. Sagisaka, S. Katagiri, H. Kuwabara, and K. Shikano, "ATR Japanese speech database as a tool of speech recognition and synthesis," *Speech communication*, pp.357–363, 1990.

[46] J. Barker, R. Marxer, E. Vincent and S. Watanabe, "The third 'CHiME' speech separation and recognition challenge: dataset, task and baseline," in *Proc. ASRU*, 2015.

[47] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc ICLR*, 2015.

[48] F. Weninger, J. R. Hershey, J. L. Roux and B. Schuller, "Discriminatively Trained Recurrent Neural Networks for Single-Channel Speech Separation," in *Proc. GlobalSIP*, 2014.

[49] F. Seide, G. Li, X. Chen and D. Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *Proc. ASRU*, pp. 24–29, 2011.

[50] R. Miyazaki, H. Saruwatari, T. Inoue, Y. Takahashi, K. Shikano and K. Kondo, "Musical-Noise-Free Speech Enhancement Based on Optimized Iterative Spectral Subtraction," *IEEE Transactions on Audio, Speech and Language Processing*, Vol. 20, pp.2080–2094, 2012.

[51] I. Cohen, "Optimal Speech Enhancement Under Signal Presence Uncertainty Using Log-Spectral Amplitude Estimator," *IEEE Signal Processing Letters*, Vol. 9, pp.113–116, 2002.

[52] E. Vincent, "An Experimental Evaluation of Wiener Filter Smoothing Techniques Applied to Under-Determined Audio Source Separation," in *Proc. LVA/ICA*, 2010.

[53] K. Niwa, Y. Hioka, and K. Kobayashi, "Post-Filter Design for Speech Enhancement in Various Noisy Environments," in *Proc IWAENC*, 2014.

[54] E. Vincent, R. Gribonval and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech and Language Processing*, 14(4), pp.1462–1469, 2006.

[55] ITU-T Recommendation P.835, "Subjective test methodology for evaluating speech communication systems that include noise suppression algorithm," 2003.

[56] S. Amano, S. Sakamoto, T. Kondo, and Y. Suzuki, "Development of familiarity-controlled word lists 2003 (FW03) to assess spoken-word intelligibility in Japanese," *Speech Communication*, pp. 76–82, 2009.

[57] K. Niwa, K. Ohtani and K, Takeda, "Music Staging AI," in *Proc. ICASSP*, 2017.

[58] S. Watanabe and J. L. Roux, "Black Box Optimization for Automatic Speech Recognition," in *Proc. ICASSP*, 2014.

Appendix

A. Deviation of (25)

We describe the deviation of (25). First, as with (19) and (20), the objective function is defined as the expectation of $\mathcal{B}(\hat{\mathbf{S}}, \mathbf{X})$ as

$$\mathcal{J}(\Theta) = \mathbb{E}_{\hat{\mathbf{S}},\mathbf{X}}\left[\mathcal{B}(\hat{\mathbf{S}}, \mathbf{X})\right], \tag{39}$$

$$= \int p(\mathbf{X}) \int \mathcal{B}(\hat{\mathbf{S}}, \mathbf{X}) p(\hat{\mathbf{S}}|\mathbf{X}, \Theta) d\hat{\mathbf{S}} d\mathbf{X}. \tag{40}$$

Then, the gradient of (40) can be calculated using a log-derivative trick as

$$\partial_\Theta \mathcal{J}(\Theta) = \mathbb{E}_{\mathbf{X}}\left[\mathbb{E}_{\hat{\mathbf{S}}|\mathbf{X}}\left[\mathcal{B}(\hat{\mathbf{S}}, \mathbf{X}) \partial_\Theta \ln p(\hat{\mathbf{S}}|\mathbf{X}, \Theta)\right]\right]. \tag{41}$$

By approximating the expectation on $\mathbf{X}$ by the average on $\mathcal{I}$ utterances and that of $\hat{\mathbf{S}}$ by the average on K times sampling, (41) can be calculated as

$$\partial_\Theta \mathcal{J}(\Theta) \approx \frac{1}{\mathcal{I}} \sum_{\tau=1}^{\mathcal{I}} \frac{1}{K} \sum_{k=1}^{K} \mathcal{B}(\hat{\mathbf{S}}^{(i,k)}, \mathbf{X}^{(i)}) \partial_\Theta \ln p(\hat{\mathbf{S}}^{(i,k)}|\mathbf{X}^{(i)}, \Theta). \tag{42}$$

We assume that the output signal on each time frame is calculated independently. Then, $\ln p(\hat{\mathbf{S}}|\mathbf{X}, \Theta)$ can be reformed to

$$\ln p(\hat{\mathbf{S}}|\mathbf{X}, \Theta) = \sum_{\tau=1}^{T} \ln p(\hat{\boldsymbol{S}}_\tau|\boldsymbol{X}_\tau, \Theta), \tag{43}$$

and its gradient can be calculated by

$$\partial_\Theta \ln p\left(\hat{\mathbf{S}}^{(i,k)}|\mathbf{X}^{(i)}, \Theta\right) = \sum_{\tau=1}^{T^{(i)}} \partial_\Theta \ln p(\hat{\boldsymbol{S}}_\tau^{(i,k)}|\boldsymbol{X}_\tau^{(i)}, \Theta), \tag{44}$$

$$\approx \frac{1}{T^{(i)}} \sum_{\tau=1}^{T^{(i)}} \partial_\Theta \ln p(\hat{\boldsymbol{S}}_\tau^{(i,k)}|\boldsymbol{X}_\tau^{(i)}, \Theta). \tag{45}$$

To normalize the difference in frame length $T^{(i)}$, we multiplied $1/T^{(i)}$ by the original gradient. The log-likelihood function $\ln p(\hat{\boldsymbol{S}}_\tau^{(i,k)}|\boldsymbol{X}_\tau^{(i)}, \Theta)$ can be expanded as

$$\ln p(\hat{\boldsymbol{S}}_\tau^{(i,k)}|\boldsymbol{X}_\tau^{(i)}, \Theta) \stackrel{c}{=} -\sum_{\omega=1}^{\Omega} \ln(\sigma_{\omega,\tau}^2)^{(i)} + \frac{\mathcal{L}_{\Re,\omega,\tau}^{(i,k)} + \mathcal{L}_{\Im,\omega,\tau}^{(i,k)}}{2(\sigma_{\omega,\tau}^2)^{(i)}}, \tag{46}$$

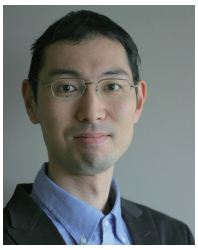
$$\mathcal{L}_{\Re,\omega,\tau}^{(i,k)} = \left(\hat{G}_{\omega,\tau}^{(i,k)} \Re\left(X_{\omega,\tau}^{(i)}\right) - \hat{G}_{\omega,\tau}^{(i)} \Re\left(X_{\omega,\tau}^{(i)}\right)\right)^2, \tag{47}$$

$$\mathcal{L}_{\Im,\omega,\tau}^{(i,k)} = \left(\hat{G}_{\omega,\tau}^{(i,k)} \Im\left(X_{\omega,\tau}^{(i)}\right) - \hat{G}_{\omega,\tau}^{(i)} \Im\left(X_{\omega,\tau}^{(i)}\right)\right)^2, \tag{48}$$

where $\hat{G}_{\omega,\tau}^{(i)}$ and $(\sigma_{\omega,\tau}^2)^{(i)}$ can be estimated by forward-propagation of the DNN as (12)–(16), and $\hat{G}_{\omega,\tau}^{(i,k)}$ is given by the sampling algorithm of the proposed method. By using above procedure, $\partial_\Theta \mathcal{J}(\Theta)$ can be calculated by simply applying back-propagation with respect to $\hat{G}_{\omega,\tau}^{(i)}$ and $(\sigma_{\omega,\tau}^2)^{(i)}$. Please note that since the simulated output signal $\hat{\boldsymbol{S}}_\tau^{(i,k)}$ deals with the "label data", the back-propagation algorithm is not applied for $\hat{G}_{\omega,\tau}^{(i,k)}$.

Yuma Koizumi (M'15) received the B.S. and M.S. from Hosei University, Tokyo, in 2012 and 2014, and the Ph.D. degree from the University of Electro-Communications in 2017. Since joining the Nippon Telegraph and Telephone Corporation (NTT) in 2014, he has been researching acoustic signal processing and machine learning. He was awarded the IPSJ Yamashita SIG Research Award from the Information Processing Society of Japan (IPSJ) in 2014 and the Awaya Prize from the Acoustical Society of Japan (ASJ) in 2017. He is a member of the ASJ and the Institute of Electronics, Information and Communication Engineers (IEICE).

Kenta Niwa (M'09) received his B.E., M.E., and Ph.D. in information science from Nagoya University in 2006, 2008, and 2014. Since joining the NTT in 2008, he has been engaged in research on microphone array signal processing as a research engineer at NTT Media Intelligence Laboratories. From 2017, he is also a visiting researcher at Victoria University of Wellington, New Zealand. He was awarded the Awaya Prize by the ASJ in 2010. He is a member of the ASJ and the IEICE.



Yusuke Hioka (S'04-M'05-SM'12) received his B.E., M.E., and Ph.D. degrees in engineering in 2000, 2002, and 2005 from Keio University, Yokohama, Japan. From 2005 to 2012, he was with the NTT Cyber Space Laboratories (now NTT Media Intelligence Laboratories), NTT in Tokyo. From 2010 to 2011, he was also a visiting researcher at Victoria University of Wellington, New Zealand. In 2013 he permanently moved to New Zealand and was appointed as a Lecturer at the University of Canterbury, Christchurch. Then in 2014, he joined the Department of Mechanical Engineering, the University of Auckland, Auckland, where he is currently a Senior Lecturer. His research interests include audio and acoustic signal processing especially microphone arrays, room acoustics, human auditory perception and psychoacoustics. He is a Senior Member of IEEE and a Member of the Acoustical Society of New Zealand, ASJ, and the IEICE.

Kazunori Kobayashi received the B.E., M.E., and Ph.D. degrees in Electrical and Electronic System Engineering from Nagaoka University of Technology in 1997, 1999, and 2003. Since joining NTT in 1999, he has been engaged in research on microphone arrays, acoustic echo cancellers and hands-free systems. He is now Senior Research Engineer of NTT Media Intelligence Laboratories. He is a member of the ASJ and the IEICE.

Yoichi Haneda (M'97-SM'06) received the B.S., M.S., and Ph.D. degrees from Tohoku University, Sendai, in 1987, 1989, and 1999. From 1989 to 2012, he was with the NTT, Japan. In 2012, he joined the University of Electro-Communications, where he is a Professor. His research interests include modeling of acoustic transfer functions, microphone arrays, loudspeaker arrays, and acoustic echo cancellers. He received paper awards from the ASJ and from the IEICE of Japan in 2002. Dr. Haneda is a senior member of IEICE, and a member of AES, ASA and ASJ.