Using Generalized Gaussian Distributions to Improve Regression Error Modeling for Deep Learning-Based Speech Enhancement

Li Chai, Jun Du, Qing-Feng Liu, and Chin-Hui Lee, *Fellow, IEEE*

***Abstract*—From a statistical perspective, the conventional minimum mean squared error (MMSE) criterion can be considered as the maximum likelihood (ML) solution under an assumed homoscedastic Gaussian error model. However, in this paper, a statistical analysis reveals the super-Gaussian and heteroscedastic properties of the prediction errors in nonlinear regression deep neural network (DNN)-based speech enhancement when estimating clean log-power spectral (LPS) components at DNN outputs with noisy LPS features in DNN input vectors. Accordingly, we propose treating all dimensions of the prediction error vector as statistically independent random variables and model them with generalized Gaussian distributions (GGDs). Then, the objective function with the GGD error model is derived according to the ML criterion. Experiments on the TIMIT corpus corrupted by simulated additive noises show consistent improvements of our proposed DNN framework over the conventional DNN framework in terms of various objective quality measures under 14 unseen noise types evaluated and at various signal-to-noise ratio levels. Furthermore, the ML optimization objective with GGD outperforms the conventional MMSE criterion, achieving improved generalization and robustness.**

***Index Terms*—Speech enhancement, deep neural network, prediction error modeling, generalized Gaussian distribution, maximum likelihood estimation.**

I. Introduction

SINGLE-CHANNEL speech enhancement has attracted considerable research attention for decades as an important front-end of speech processing systems. It aims to reduce noise in input speech and improve the quality and intelligibility of speech signals in the context of automatic speech recognition, mobile communications and hearing aids [1], [2], among other applications. Nonetheless, the performance of speech enhancement in real-world adverse acoustic environments remains unsatisfactory and challenging, particularly due to the wide variety of unseen environmental noises.

Conventional speech enhancement, which includes a wide range of approaches, such as spectral subtraction [3], Wiener filtering [4], minimum mean squared error (MMSE) estimation [5], and the optimally-modified log-spectral amplitude (OM-LSA) speech estimator [6], [7], has been extensively studied in the past. An annoying artifact, called musical noise, is generally encountered in these techniques. Moreover, these techniques often fail to track non-stationary noises for real-world scenarios in unexpected acoustic conditions.

With the development of high-performance computing and a breakthrough in training deep architectures [8], [9], deep learning technologies have been successfully used for many applications [10]. Recently, different types of deep neural networks (DNNs) have been applied as regression models for speech separation and speech enhancement, such as feed-forward DNNs in [11]–[14], recurrent neural networks (RNNs) in [15]–[17], and convolutional neural networks (CNNs) in [18], [19]. These models have demonstrated superiority over conventional approaches due to their powerful modeling capabilities and requirement of fewer assumptions about the signals. DNN-based speech enhancement can be divided into two main groups: mapping-based methods and masking-based methods. One representative mapping-based method [13], [14] adopted feed-forward DNNs to model the complicated relationship between noisy and clean speech by learning deep and wide DNN architectures using a large collection of heterogeneous training data and abundant acoustic contextual information. This approach was able to address nonlinear and non-stationary noises effectively. Instead of performing direct mapping, Wang *et al.* [12], [20] employed a set of complementary features extracted from corrupted speech to estimate the ideal binary mask (IBM) or smoothed ideal ratio mask (IRM). The predicted mask was then applied to the noisy features to obtain the enhanced features. Speech enhancement using the IBM target can be formulated as a binary classification task that can effectively improve speech intelligibility. Speech enhancement using the soft IRM target is especially beneficial for improving the objective speech quality. In addition, Erdogan *et al.* [21] developed a phase-sensitive mask that incorporates the phase difference between noisy speech and clean speech.

Manuscript received April 12, 2019; revised July 2, 2019 and August 8, 2019; accepted August 12, 2019. Date of publication August 19, 2019; date of current version August 30, 2019. This work was supported in part by the National Key R&D Program of China under Contract 2017YFB1002202, in part by the National Natural Science Foundation of China under Grants 61671422 and U1613211, in part by the Key Science and Technology Project of Anhui Province under Grant 17030901005, and in part by Huawei Noah's Ark Lab. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Tan Lee. *(Corresponding author: Jun Du.)*

L. Chai, J. Du, and Q.-F. Liu are with the National Engineering Laboratory for Speech and Language Information Processing, University of Science and Technology of China, Hefei 230027, China (e-mail: cl122@mail.ustc.edu.cn; jundu@ustc.edu.cn; qfliu@iflytek.com).

C.-H. Lee is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: chl@ece.gatech.edu).

Digital Object Identifier 10.1109/TASLP.2019.2935803

To jointly enhance the magnitude and phase spectra, a complex IRM was proposed in [22].

From a machine learning perspective, one key challenge of DNN-based speech enhancement lies in optimizing the complicated and nonconvex objective function. DNN-based speech enhancement using the IBM target can naturally be treated as a supervised classification problem. A commonly used objective function for IBM estimation is cross entropy. In contrast, DNN-based speech enhancement using another target (e.g., magnitude spectrum or log power spectrum of a clean signal, or IRM) can be treated as a supervised regression problem. A commonly used objective function for target prediction is MMSE. Advanced objective functions for regression DNN-based speech enhancement have been investigated in recent studies. A technique to jointly optimize all the sources with a discriminative objective function was proposed in [15], [16], [23], [24]. A multi-objective learning scheme was adopted to utilize secondary targets in [19], [25]. Some studies have applied an elementwise weight function and a penalty term to the conventional MMSE by considering human auditory perception [26]–[30]. In addition, some differentiable objective metrics have been optimized directly by including them in the objective function [31]. More recently, [32]–[35] directly incorporated short-time objective intelligibility (STOI) [36] into the loss function to maximize speech intelligibility. However, these approaches showed no or only modest improvements in STOI compared to the conventional MMSE criterion. [35] also incorporated the perceptual evaluation of speech quality (PESQ) [37] into the loss function and then optimized it via a gradient approximation method to maximize speech quality. In addition, [38] directly utilized perceptual metrics, such as PESQ or STOI, as the objective function and then optimized it through reinforcement learning. However, the main problem encountered in [32], [35], [38] is that STOI- or PESQ-only optimization often leads to degradation of another evaluation metric. For example, they achieved improved STOI at the cost of a decrease in PESQ.

From a statistical perspective, the MMSE criterion can be considered as the maximum likelihood (ML) solution under an assumed independent, normally distributed and homoscedastic noise model [39]–[41]. However, in this study, the statistical analysis presented in Section III-A reveals the super-Gaussian nature and heteroscedasticity [42] of the prediction error distribution of the nonlinear regression DNN [14] for each log-power spectral (LPS) component. MMSE generally enjoys certain well-known optimality properties within strictly Gaussian parametric models. Accordingly, we should pay attention to the observation that MMSE may fail to provide optimal results in the case of non-Gaussian error distributions because it can be very sensitive to departures from normality, particularly for long-tailed situations [41], [43]–[45]. This is one of main reasons for why conventional DNN framework denoted as MMSE-DNN is not robust in adverse acoustic environments, e.g., it leads to over-smoothing and speech information loss problems in conditions with a low signal-to-noise ratio (SNR) [46]. To address this issue, we aim to improve prediction error modeling by considering a generalized Gaussian distribution (GGD) [47] for DNN-based speech enhancement. The family of GGDs is more flexible for data modeling, with the distributions ranging from a highly peaked one to a uniform density. Then, a new objective function with the GGD error model is derived according to the ML criterion. The main difference of our proposed ML framework from the conventional DNN-based regression using the MMSE criterion for function fitting is that we replace the conventional mean squared error (MSE) with the log-likelihood as the new DNN optimization objective. The scripts in our experiments can be downloaded from the official GitHub website.[1]

Our proposed ML framework denoted as ML-GGD-DNN, which still aims to learn the many-to-one mapping function between the input features and the target features by modeling the prediction error in a deterministic manner, is different from the mixture density network (MDN) [40], which is used to learn the conditional probability distribution of the target vector conditioned on the input vector to obtain a complete description of the data. When the DNN-based speech enhancement problem is considered to be an ill-posed inverse problem where the mapping cannot be uniquely determined given an input signal, the assumption of many-to-one mapping in either MMSE-DNN or ML-GGD-DNN is not theoretically correct and potentially limits the performance. Although [48] proposed a novel framework utilizing an MDN to take the estimation uncertainty into account, this novel framework required an extra statistical model-based enhancement approach to obtain the final clean speech estimation, and it showed only a slight improvement in log-spectral distortion (LSD) compared to the conventional MMSE-DNN. In contrast, although our proposed ML-GGD-DNN still aims to learn the many-to-one mapping function, it achieves consistent improvements in terms of various evaluation metrics compared to the conventional MMSE-DNN. Moreover, although both DNN parameters and GGD parameters need to be optimized in the training stage for our ML-GGD-DNN, no additional computational cost is introduced compared to the conventional MMSE-DNN because the number of DNN parameters is significantly greater than the number of GGD parameters.

This work is an extension of our previously published version [49] with the following new contributions:

- We present an extended general ML framework with the corresponding formulations for DNN-based regression that can be applied to various regression tasks.
- We elaborate a statistical analysis of the prediction errors, which provides guidance for the selection of the specific parameterized form of the error density.
- Based on the super-Gaussian and heteroscedastic properties of the prediction error distributions in the nonlinear regression DNN-based speech enhancement revealed by a detailed statistical analysis, a flexible and reasonable GGD is statistically investigated and applied to model the prediction error distribution in each dimension. Our previous work using a Gaussian distribution (GD) to model the prediction error [49] is only a special case of the newly proposed ML-GGD-DNN framework.

[1]https://github.com/LiChaiUSTC/Speech-enhancement-based-on-a-maximum-likelihood-criterion

- This work proposes two algorithms for optimization: one algorithm achieves better robustness, a slightly better performance and easier implementation, and the other is presented as an alternative.
- The presented experiments demonstrate that the super-Gaussian nature and heteroscedasticity represented by the shape and scale factors in the GGD, respectively, are the two main contributors to the success of the ML-GGD-DNN for our DNN-based speech enhancement task. More specifically, the ML-GGD-DNN with the assumption of heteroscedasticity achieves a faster convergence and a better generalization capability than that with the assumption of homoscedasticity. The ML-GGD-DNN with a super-Gaussian assumption achieves better robustness and performance than that with a Gaussian assumption.
- Compared with either the conventional MMSE-DNN or ML-GD-DNN in [49], the proposed approach achieves better robustness and generalization capability, with improvements in both speech intelligibility and quality.
- The proposed framework is successfully extended to the task of speech dereverberation. This provides strong support for its versatility for various regression tasks.

The remainder of this paper is organized as follows. An extended general ML framework with the corresponding formulations for DNN-based regression is presented in Section II. Then, in Section III, a GGD is defined to model the prediction error of DNN-based speech enhancement based on the elaborated statistical analysis, followed by two training algorithms that jointly optimize the DNN and GGD parameters and special cases of ML-GGD-DNN. In Section IV, we present an in-depth evaluation of the ML-GGD-DNN approach with an extension to speech dereverberation. Finally, we conclude in Section V.

II. ML Framework for DNN-Based Regression

Before introducing the specific model for speech enhancement, we first propose a general ML framework for DNN regression that can potentially be applied to many research areas, such as computer vision [50], [51], handwriting recognition [52], speech dereverberation [53], speech enhancement and separation [14], [54]. Moreover, the DNN architectures can be feed-forward DNNs, CNNs or RNNs.

Suppose that we have a training set with N data pairs $(\boldsymbol{Y}, \boldsymbol{X}) = \{(\boldsymbol{y}_n, \boldsymbol{x}_n)|n = 1, 2, ...N\}$, where $\boldsymbol{y}_n$ and $\boldsymbol{x}_n$ are the input D_{in}-dim and the target D_{out}-dim feature vectors of the n-th sample, respectively; then, a regression DNN can be adopted to model the mapping $\mathcal{F}$ between the two high-dimensional vectors as follows:

$$\hat{\boldsymbol{x}}_n = \mathcal{F}(\boldsymbol{y}_n; \boldsymbol{W}), \tag{1}$$

where $\hat{\boldsymbol{x}}_n$ is the n-th output estimated D_{out}-dimensional feature vector and $\boldsymbol{W}$ is the parameter set of the regression DNN. The corresponding prediction error vector $\boldsymbol{e}_n$ is defined as

$$\boldsymbol{e}_n = \boldsymbol{x}_n - \hat{\boldsymbol{x}}_n. \tag{2}$$

From a probabilistic perspective, we can reasonably assume that the prediction error vector follows a specific parameterized density $g(\boldsymbol{e}_n|\boldsymbol{\Theta})$ with $\boldsymbol{\Theta}$ as the parameter set. If the target vector $\boldsymbol{x}_n$ is also a random vector and $\hat{\boldsymbol{x}}_n$ is considered as a fixed vector with an unknown parameter set $\boldsymbol{\Theta}$, then by using Eqs. (1) and (2), an equivalent conditional distribution of the target vector $\boldsymbol{x}_n$ can be derived as follows:

$$p(\boldsymbol{x}_n|\boldsymbol{y}_n, \boldsymbol{W}, \boldsymbol{\Theta}) = g(\boldsymbol{x}_n - \mathcal{F}(\boldsymbol{y}_n; \boldsymbol{W})|\boldsymbol{\Theta}). \tag{3}$$

By assuming that all data pairs are drawn independently from the distribution in Eq. (3), the log-likelihood function of $(\boldsymbol{Y}, \boldsymbol{X})$ is written as follows:

$$\ln p(\boldsymbol{X}|\boldsymbol{Y}, \boldsymbol{W}, \boldsymbol{\Theta}) = \sum_{n=1}^{N} \ln g(\boldsymbol{x}_n - \mathcal{F}(\boldsymbol{y}_n; \boldsymbol{W})|\boldsymbol{\Theta}), \tag{4}$$

where the parameter set $(\boldsymbol{W}, \boldsymbol{\Theta})$ is to be optimized by maximizing the log-likelihood function. In Section III-C, two training algorithms for optimizing $\boldsymbol{W}$ and $\boldsymbol{\Theta}$ are presented. Note that the DNN parameters $\boldsymbol{W}$ and the distribution parameters $\boldsymbol{\Theta}$ are jointly learned during training; only the DNN parameters $\boldsymbol{W}$ are needed during testing. The main difference of our proposed ML framework from the conventional DNN-based regression using the MMSE criterion for function fitting is that we replace the conventional MSE with the log-likelihood as the new DNN optimization objective. In other words, we design a novel optimization approach for the DNN parameters with a fixed DNN architecture.

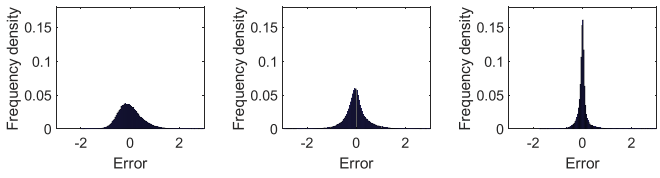
One key issue here is to set the specific form of the error density, $g(\boldsymbol{\Theta})$. According to our studies on speech enhancement, $g(\boldsymbol{\Theta})$ is determined by the input/target data properties and DNN type. For example, $g(\boldsymbol{\Theta})$ may differ for computer vision and speech processing. Even for the same task, e.g., speech enhancement, the design of different learning targets may also lead to different error densities. Thus, the selection should depend on a statistical analysis of prediction errors.

III. ML-GGD-DNN for Speech Enhancement

A. Selection of Prediction Error Distributions

This study primarily explores applying the ML framework in Section II for the feed-forward DNN-based speech enhancement approach proposed in [14]. To determine the form of the prediction error distribution, a statistical analysis is conducted on the prediction error vectors of the cross-validation set for the conventional feed-forward DNN framework in [14]. The cross-validation set consists of approximately 400 sentences randomly selected from the 80-hour multi-condition data set consisting of pairs of clean and noisy speech utterances. A detailed description of the cross-validation set is provided in Section IV-A. The prediction error vectors are calculated by subtracting the enhanced feature vectors from the target feature vectors, where the enhanced feature vectors are from the output of the conventional feed-forward DNN in [14] whose parameters have been well optimized in advance.

First, we show the normalized histograms of randomly selected dimensions (2, 128, 256) of the prediction error vectors in Fig. 1. These dimensions are selected to represent errors in low-frequency, middle-frequency, and high-frequency respectively. Fig. 1 indicates that the distributions of all dimensions are clearly unimodal and can well satisfy the assumptions of



(a) D_2 (b) D_{128} (c) D_{256}

Fig. 1. The normalized histograms for selected dimensions (D_2, D_{128}, D_{256}) of the prediction error vector on the cross-validation set. These selected three dimensions represent errors in low-frequency, middle-frequency, and high-frequency, respectively.

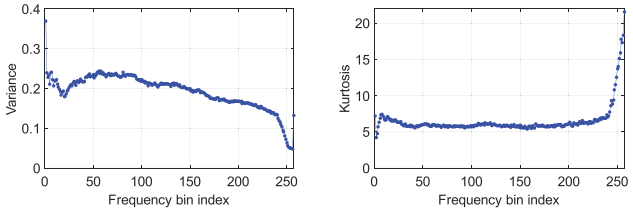


Fig. 2. The variance/kurtosis of each dimension of the prediction error vector on the cross-validation set. The left figure shows the variance values in 257 dimensions. The right figure shows the kurtosis values in 257 dimensions.

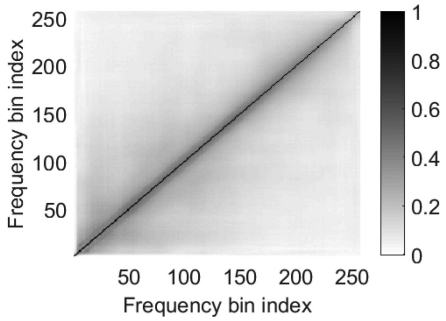
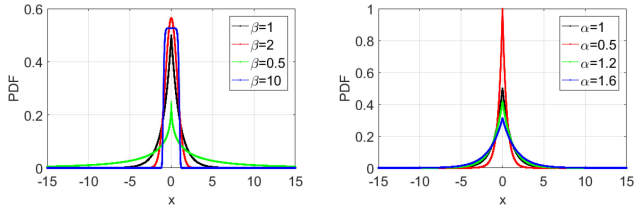


Fig. 3. The correlation coefficient matrix of the prediction error vector on the cross-validation set.

zero mean and heteroscedasticity. This partially explains why the MMSE criterion with the assumption of homoscedasticity [55], or an identical scale factor in all dimensions of the prediction error vectors, often leads to a poor generalization capability. Second, the statistics of all dimensions, including the variance and kurtosis, are shown in Fig. 2, where the variance curve explicitly demonstrates the heteroscedasticity and the kurtosis curve reveals that all dimensions follow the super-Gaussian distribution with a kurtosis greater than 3 (the kurtosis of a GD). Finally, Fig. 3 plots the correlation coefficient matrix of the prediction error vectors. This figure indicates that the correlations among most dimensions are weak despite the relatively strong correlations between adjacent dimensions. Consequently, it is reasonable to make the assumption of independence among the dimensions of the prediction error vector, which also simplifies the formula derivation.

Based on the above analysis, we utilize the univariate GGD to model the prediction error distribution in each dimension, whose probability density function (PDF) is defined as

$$p_{\mathrm{GGD}}(x|\mu,\alpha,\beta) = \frac{\beta}{2\alpha\Gamma(\frac{1}{\beta})}\exp\left(-\frac{|x-\mu|^{\beta}}{\alpha^{\beta}}\right), \quad (5)$$



(a) $\mu = 0, \alpha = 1$ (b) $\mu = 0, \beta = 1$

Fig. 4. The PDFs of GGDs, where the parameters β and α are measures of the distribution's kurtosis and variance, respectively. The left figure shows the variation tendency of the distribution with respect to β, where the distribution changes from sub-Gaussian to super-Gaussian as β decreases. The right figure shows the variation tendency of the distribution with respect to α, where the distribution becomes more concentrated around zero mean as α decreases.

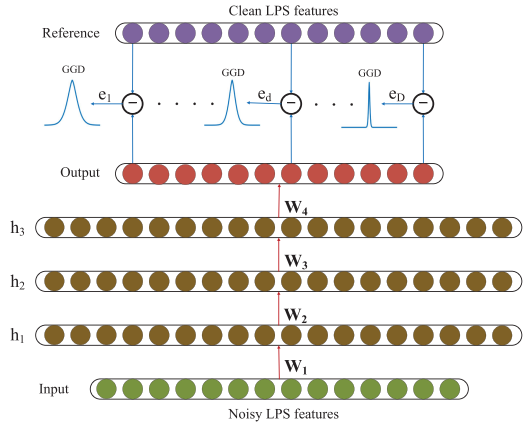


Fig. 5. The ML-GGD-DNN architecture for speech enhancement.

where $\Gamma(\cdot)$ denotes the Gamma function and μ is the mean parameter. α is a scale factor that plays the role of variance, while β is a shape factor that measures the kurtosis [56] and controls the GGD deviation from normality, as intuitively shown in Fig. 4. By setting β, many classic distributions can be characterized, including uniform ($\beta = \infty$), Gaussian ($\beta = 2$), Laplacian ($\beta = 1$), and other sub-Gaussian ($\beta > 2$) and super-Gaussian ($\beta < 2$) densities. Then, the ML estimation described in Section II is adopted for DNN parameter learning. The proposed ML-GGD-DNN architecture is shown in Fig. 5. Note that the same MMSE-DNN structure is used.

B. Derivation of the New Objective Function

The DNN input is the $(2\tau+1)D$-dimensional LPS feature vector of noisy speech with τ frames in both the left and right context, while the DNN target output is the D-dimensional LPS feature vector of clean speech. Suppose that the DNN output vector at the n-th frame is $\hat{\boldsymbol{x}}_n(\boldsymbol{y}_{n-\tau}^{n+\tau}, \boldsymbol{W})$ with the input vector $\boldsymbol{y}_{n-\tau}^{n+\tau}$ and the DNN parameter set $\boldsymbol{W}$, while the corresponding reference vector is $\boldsymbol{x}_n$. Accordingly, the prediction error vector at the n-th frame can be defined as follows:

$$\boldsymbol{e}_n = \boldsymbol{x}_n - \hat{\boldsymbol{x}}_n(\boldsymbol{y}_{n-\tau}^{n+\tau}, \boldsymbol{W}), \quad (6)$$

where $\hat{\boldsymbol{x}}_n(\boldsymbol{y}_{n-\tau}^{n+\tau}, \boldsymbol{W}) = \mathcal{F}(\boldsymbol{y}_{n-\tau}^{n+\tau}; \boldsymbol{W})$ and $\mathcal{F}$ refers to a mapping modeled by the DNN. Based on the statistical analysis elaborated in Section III-A, it is assumed that each dimension of the prediction error vector independently follows the GGD

in Eq. (5) with zero mean, the same known shape factor, and different unknown scale factors. Therefore, the prediction error distribution is defined as

$$g(\boldsymbol{e}_n|\boldsymbol{\alpha}) = \prod_{d=1}^{D} p_{\text{GGD}}(e_{n,d}|0,\alpha_d,\beta) = \prod_{d=1}^{D} \frac{\beta}{2\alpha_d\Gamma(\frac{1}{\beta})} \exp\left(-\frac{|e_{n,d}|^\beta}{\alpha_d^\beta}\right), \quad (7)$$

where $e_{n,d}$ is the d-th element of $\boldsymbol{e}_n$, α_d is the d-th element of $\boldsymbol{\alpha}$, and β is a fixed value for all dimensions. By inserting $\boldsymbol{e}_n$ from Eq. (6) into Eq. (7), the equivalent conditional distribution in Eq. (3) can be rewritten as follows:

$$p(\boldsymbol{x}_n|\boldsymbol{y}_{n-\tau}^{n+\tau},\boldsymbol{W},\boldsymbol{\alpha}) = \prod_{d=1}^{D} p_{\text{GGD}}(x_{n,d}-\hat{x}_{n,d}(\boldsymbol{y}_{n-\tau}^{n+\tau},\boldsymbol{W})|0,\alpha_d,\beta) = \prod_{d=1}^{D} \frac{\beta}{2\alpha_d\Gamma(\frac{1}{\beta})} \exp\left(-\frac{|x_{n,d}-\hat{x}_{n,d}(\boldsymbol{y}_{n-\tau}^{n+\tau},\boldsymbol{W})|^\beta}{\alpha_d^\beta}\right). \quad (8)$$

This is the PDF of the conditional target distribution [57] based on an assumed heteroscedastic GGD error model. Given a set with N data pairs $(\boldsymbol{Y},\boldsymbol{X}) = \{(\boldsymbol{y}_{n-\tau}^{n+\tau},\boldsymbol{x}_n)|n=1,2,...,N\}$ and assuming that they are drawn independently from the distribution in Eq. (8), the corresponding likelihood function is simply:

$$p(\boldsymbol{X}|\boldsymbol{Y},\boldsymbol{W},\boldsymbol{\alpha}) = \prod_{n=1}^{N} p(\boldsymbol{x}_n|\boldsymbol{y}_{n-\tau}^{n+\tau},\boldsymbol{W},\boldsymbol{\alpha}). \quad (9)$$

Finally, the log-likelihood function in Eq. (4) is specified as:

$$\ln p(\boldsymbol{X}|\boldsymbol{Y},\boldsymbol{W},\boldsymbol{\alpha}) = \sum_{n=1}^{N}\sum_{d=1}^{D}\left(\ln\frac{\beta}{2\alpha_d\Gamma(\frac{1}{\beta})} - \frac{|x_{n,d}-\hat{x}_{n,d}(\boldsymbol{y}_{n-\tau}^{n+\tau},\boldsymbol{W})|^\beta}{\alpha_d^\beta}\right) = C - N\sum_{d=1}^{D}\ln\alpha_d - \sum_{n=1}^{N}\sum_{d=1}^{D}\frac{|x_{n,d}-\hat{x}_{n,d}(\boldsymbol{y}_{n-\tau}^{n+\tau},\boldsymbol{W})|^\beta}{\alpha_d^\beta}, \quad (10)$$

where $C = ND\ln\frac{\beta}{2\Gamma(\frac{1}{\beta})}$ is a constant not related to $\boldsymbol{W}$ and $\boldsymbol{\alpha}$. If we assume that all the scale factors are the same, namely, the homoscedasticity assumption, the log-likelihood function in Eq. (10) shares all dimensions with the same scale factor α and thus becomes:

$$\ln p(\boldsymbol{X}|\boldsymbol{Y},\boldsymbol{W},\alpha) = C - ND\ln\alpha - \sum_{n=1}^{N}\sum_{d=1}^{D}\frac{|x_{n,d}-\hat{x}_{n,d}(\boldsymbol{y}_{n-\tau}^{n+\tau},\boldsymbol{W})|^\beta}{\alpha^\beta}. \quad (11)$$

Then the optimization of the proposed objective function in Eq. (10) is regressed to minimizing the classic β-norm function, where $\beta = 1$ corresponds to the L1-norm and $\beta = 2$ corresponds to the L2-norm.

By maximizing the log-likelihood in Eq. (10), the solution of the parameter set $(\boldsymbol{W},\boldsymbol{\alpha})$ can be derived. Note that the scale vector $\boldsymbol{\alpha}$ is adopted only in the training stage. In the enhancement stage, the GGD information is no longer used.

Algorithm 1: Procedure of ML-GGD-DNN Training.

Step 1: Initialization
Initialize $\boldsymbol{W}$ randomly and $\alpha_d = 1$ $(d = 1,2,...,D)$.
Step 2: Update $\boldsymbol{W}$ and $\boldsymbol{\alpha}$ in each minibatch
Update $\boldsymbol{W}$ and $\boldsymbol{\alpha}$ simultaneously via Eqs. (13) and (14).
Step 3: Go to Step 2 for the next iteration

C. ML-GGD-DNN Training Procedure

To estimate $\boldsymbol{W}$ and $\boldsymbol{\alpha}$ via Eq. (10), we propose two procedures for training ML-GGD-DNN, as shown in Algorithms 1 and 2. In Algorithm 1, stochastic gradient descent (SGD) [58] is performed in minibatch mode of M sample frames to update $\boldsymbol{W}$ and $\boldsymbol{\alpha}$ simultaneously. Correspondingly, we minimize the following error function rather than maximizing the log-likelihood in Eq. (10):

$$E(\boldsymbol{W},\boldsymbol{\alpha}) = M\sum_{d=1}^{D}\ln\alpha_d + \sum_{m=1}^{M}\sum_{d=1}^{D}\frac{|x_{m,d}-\hat{x}_{m,d}(\boldsymbol{y}_{m-\tau}^{m+\tau},\boldsymbol{W})|^\beta}{\alpha_d^\beta}. \quad (12)$$

Based on Eq. (12), the update formulas of $\boldsymbol{W}$ and $\boldsymbol{\alpha}$ can be written as follows:

$$\boldsymbol{W}^{(t+1)} = \boldsymbol{W}^{(t)} - \eta\frac{\nabla E_{\boldsymbol{W}}}{M} \quad (13)$$

$$\boldsymbol{\alpha}^{(t+1)} = \boldsymbol{\alpha}^{(t)} - \xi\frac{\nabla E_{\boldsymbol{\alpha}}}{M} \quad (14)$$

where t is the iteration index for the model parameter update; η and ξ are the learning rates; and $\nabla E_{\boldsymbol{W}}$ and $\nabla E_{\boldsymbol{\alpha}}$ are the gradients of $E(\boldsymbol{W},\boldsymbol{\alpha})$ with respect to $\boldsymbol{W}$ and $\boldsymbol{\alpha}$, respectively. $\nabla E_{\boldsymbol{\alpha}}$ is a D-dimensional vector whose d-th element is

$$\nabla E_{\alpha_d} = \frac{M}{\alpha_d} - \frac{\beta}{(\alpha_d)^{\beta+1}}\sum_{m=1}^{M}|x_{m,d}-\hat{x}_{m,d}(\boldsymbol{y}_{m-\tau}^{m+\tau},\boldsymbol{W})|^\beta. \quad (15)$$

The gradient $\nabla E_{\boldsymbol{W}}$ is often obtained by using the chain rule via the intermediate variables, namely, the DNN output, which are the same as those in MMSE-DNN [14]. Therefore, we only need to derive the gradient with respect to the output:

$$\nabla E_{\hat{x}_{m,d}} = \text{sgn}\left(\hat{x}_{m,d}-x_{m,d}\right)\frac{\beta}{\alpha_d^\beta}|\hat{x}_{m,d}-x_{m,d}|^{\beta-1}, \quad (16)$$

where $\hat{x}_{m,d}$ is a shorthand notation of $\hat{x}_{m,d}(\boldsymbol{y}_{m-\tau}^{m+\tau},\boldsymbol{W})$. Note that when the sample prediction error is equal to zero for $\beta < 1$, $\nabla E_{\hat{x}_{m,d}}$ is infinite; that is, this point is non-differentiable. The sample prediction error is equal to zero, which indicates this point is learned very well and thus should not contribute to the backpropagation [59]. Accordingly, we set the value of $\nabla E_{\hat{x}_{m,d}}$ at this point to zero.

Algorithm 2: Procedure of ML-GGD-DNN Training.

Step 1: Initialization
Initialize $\boldsymbol{W}$ randomly.
Step 2: Fix $\boldsymbol{W}$ and update $\boldsymbol{\alpha}$ in each minibatch
Update $\boldsymbol{\alpha}$ via Eq. (17).
Step 3: Fix $\boldsymbol{\alpha}$ and update $\boldsymbol{W}$ in each minibatch
Update $\boldsymbol{W}$ via Eq. (13).
Step 4: Go to Step 2 for the next iteration

An alternating two-step optimization scheme is used in Algorithm 2 for the optimization of $\boldsymbol{W}$ and $\boldsymbol{\alpha}$. First, a closed-form solution of $\boldsymbol{\alpha}$ can be derived by fixing $\boldsymbol{W}$ and minimizing $E(\boldsymbol{W}, \boldsymbol{\alpha})$ in Eq. (12):

$$\alpha_d = \left[\frac{\beta}{M} \sum_{m=1}^{M} |x_{m,d} - \hat{x}_{m,d}(\boldsymbol{y}_{m-\tau}^{m+\tau}, \boldsymbol{W})|^{\beta} \right]^{\frac{1}{\beta}}. \quad (17)$$

Second, by fixing $\boldsymbol{\alpha}$, $\boldsymbol{W}$ can be optimized by the backpropagation procedure with the SGD method in Eq. (13). The error function given by Eq. (12) is guaranteed to decrease monotonically with the above update procedures because Eq. (17) and Eq. (13) are derived as transformations that minimize Eq. (12) with the fixed $\boldsymbol{W}$ and $\boldsymbol{\alpha}$, respectively. The convergence of the above iterative optimization is also guaranteed because Eq. (12) is lower-bounded as long as $\boldsymbol{\alpha}$ is updated by Eq. (17).

The main difference between Algorithms 1 and 2 is that Algorithm 2 uses a closed-form solution for $\boldsymbol{\alpha}$ without tuning the learning rate ξ. The experiments in Section IV-B show that Algorithm 2 is more robust because Algorithm 1 is sensitive to ξ. For both procedures in Algorithm 1 and 2, training is repeated until a predefined convergence criterion is satisfied or a maximum number of epochs is reached. Note that although the two algorithms are shown based on the SGD and backpropagation, their underlying ideas could easily be adapted to other learning algorithms, such as backpropagation through time [60].

D. Special Cases of ML-GGD-DNN

1) ML-GGD-DNN With $\beta = 2$: With $\beta = 2$, GGD corresponds to the GD. We denote this DNN model as ML-GD-DNN. Accordingly, the error function in Eq. (12) becomes:

$$E_2(\boldsymbol{W}, \boldsymbol{\alpha}) = M \sum_{d=1}^{D} \ln \alpha_d + \sum_{m=1}^{M} \sum_{d=1}^{D} \left(\frac{x_{m,d} - \hat{x}_{m,d}(\boldsymbol{y}_{m-\tau}^{m+\tau}, \boldsymbol{W})}{\alpha_d} \right)^2, \quad (18)$$

which is exactly the formulation of our previous work using a GD to model the prediction error [49]. If we further make the homoscedasticity assumption, then the objective function of ML-GD-DNN in Eq. (18) degenerates into the MSE function of MMSE-DNN [14].

2) ML-GGD-DNN with $\beta = 1$: With $\beta = 1$, GGD corresponds to the Laplacian distribution (LD). We denote this DNN model as ML-LD-DNN. Accordingly, the error function in Eq. (12) becomes:

$$E_1(\boldsymbol{W}, \boldsymbol{\alpha}) = M \sum_{d=1}^{D} \ln \alpha_d + \sum_{m=1}^{M} \sum_{d=1}^{D} \frac{|x_{m,d} - \hat{x}_{m,d}(\boldsymbol{y}_{m-\tau}^{m+\tau}, \boldsymbol{W})|}{\alpha_d}. \quad (19)$$

Similarly, ML-LD-DNN will degenerate into DNN based on the least absolute deviation (LAD) [61] as opposed to least squares (LS or MMSE), denoted as LAD-DNN, situations under the homoscedasticity assumption. The LAD is more robust and less affected by outlier contamination than the MMSE [43]–[45], [62]. In particular, the LAD is well suited for longer-tailed error distributions. In [63], the LAD is adopted as the optimization criterion of the shallow neural network rather than the MMSE criterion. In this study, experiments are conducted to compare the LAD with the MMSE as the special cases of ML-GGD for the feed-forward DNN-based speech enhancement task, which can be found in Sections IV-C and IV-E.

IV. Experiments and Result Analysis

A. Experimental Setup

In this study, all experiments were conducted on speech waveforms with 16 kHz sampling. A total of 115 noise types, including 100 noise types in [64] and 15 homemade noise types, were adopted for training to improve the robustness to unseen noise conditions. The clean speech data were derived from the TIMIT corpus [65]. All 4620 utterances from the training set of TIMIT were corrupted with the abovementioned 115 noise types at six levels of SNRs, i.e., 20 dB, 15 dB, 10 dB, 5 dB, 0 dB and -5 dB, to build an 80-hour multi-condition data set consisting of pairs of clean and noisy speech utterances. Approximately 400 sentences randomly selected from the 80-hour data set were used as the cross-validation set, and the rest were used as the training set. The 192 utterances from the core test set of TIMIT were used to construct the test set for each combination of unseen noise type and seen SNR level. In this study, 14 unseen noise types were adopted for testing, i.e., Jet Cockpit2, Destroyer Engine, Destroyer Operations, F-16 Cockpit, Factory1, Factory2, HF Channel, Military Vehicle, M109 Tank, Machine Gun, Pink, Volvo, Speech Babble and White noise, denoted as N1, N2, N3, N4, N5, N6, N7, N8, N9, N10, N11, N12, N13 and N14, respectively. All were collected from the NOISEX-92 corpus [66], where Military Vehicle, Pink, Volvo and White noise are stationary and the others are non-stationary. The six levels of SNRs in the training set were also adopted for testing.

For feature extraction, a 512-point short-time Fourier transform was used to compute the spectra of each overlapping windowed frame. Here, a 32-ms Hamming window and a 16-ms window shift were adopted. The input and target feature design were investigated in [67]. In our experiments, 257-dimensional ($D = 257$) LPS vectors were generated to train the DNNs. Mean and variance normalizations were applied to the input and target feature vectors of the DNN. All DNN architectures were fixed as 1799-2048-2048-2048-257, namely, the sizes were 1799 for the input layer (7-frame expansion), 2048 for the three hidden layers, and 257 for the output layer. Note that these specific DNN

TABLE I
A COMPARISON OF THE AVERAGE PERFORMANCE OF TWO TRAINING PROCEDURES FOR ML-GGD-DNN ($\beta = 2$) ON THE TEST SET AVERAGED OVER ALL EXAMINED SNRS AND 14 UNSEEN NOISE TYPES

Metrics	STOI	LSD	SegSNR	PESQ
Algorithm 1	0.865	3.18	4.94	2.92
Algorithm 2	0.869	3.11	5.13	2.92

architectures were selected with reference to [13], [14]. The sigmoid activation function was utilized for all hidden layers, and the output layer was linear. For the parameter update in MMSE-DNN and ML-GGD-DNN, which were both initialized randomly, the learning rate for supervised fine-tuning was set to 0.1 ($\eta = 0.1$) for the first 10 epochs and decreased at a rate of 90% after every epoch in the next 40 epochs with a minibatch size of 128 ($M = 128$). For waveform reconstruction, the phase of noisy speech was combined with the enhanced LPS features.

The enhancement performance was assessed by using PESQ for measuring speech quality [37], STOI [36] for measuring speech intelligibility and segmental SNR (SegSNR in dB) and LSD (in dB) for evaluating signal differences in the time domain and the frequency domain, respectively [68].

B. Training Procedure of ML-GGD-DNN

In Section III-C, we propose two training procedures of ML-GGD-DNN listed in Algorithms 1 and 2. To compare them, a set of experiments was conducted for ML-GGD-DNN with $\beta = 2$ (namely, ML-GD-DNN with a GD), as shown in Table I. We examined the average performance of four measures at different SNRs across 14 unseen noise types. For Algorithm 1, the main issue is the setting of the learning rate ξ for updating $\boldsymbol{\alpha}$. If we set ξ to be the same as $\eta = 0.1$, the learning process cannot converge. Thus, in our experiment, ξ was set to 10^{-7} to guarantee convergence. In contrast, Algorithm 2 has a closed-form solution for $\boldsymbol{\alpha}$ in Eq. (17), which can be considered as a version of Algorithm 1 with an automatic setting of the learning rate ξ. Based on the results in Table I, Algorithm 2 yielded slight but consistent improvements in STOI/SegSNR/LSD measures over Algorithm 1. For PESQ, the two procedures generated comparable results. Overall, Algorithm 2 demonstrated better performance and easier implementation. Therefore, we selected Algorithm 2 as the default training procedure in all subsequent experiments.

C. Learning Behavior of ML-GGD-DNN

In Fig. 6, we present the learning curves of the square roots of MSEs and log-likelihoods among MMSE-DNN, LAD-DNN and ML-GGD-DNN with different shape factors on the cross-validation set. The log-likelihoods of ML-GGD-DNN with different shape factors are computed using Eq. (10), while the log-likelihoods of MMSE-DNN and LAD-DNN are computed using Eq. (11). Note that the closed-form solution of the scale factor α in Eq. (11) for Algorithm 2 is as follows:

$$\alpha = \left[\frac{\beta}{MD} \sum_{m=1}^{M} \sum_{d=1}^{D} |x_{m,d} - \hat{x}_{m,d}(\boldsymbol{y}_{m-\tau}^{m+\tau}, \boldsymbol{W})|^{\beta} \right]^{\frac{1}{\beta}}. \quad (20)$$

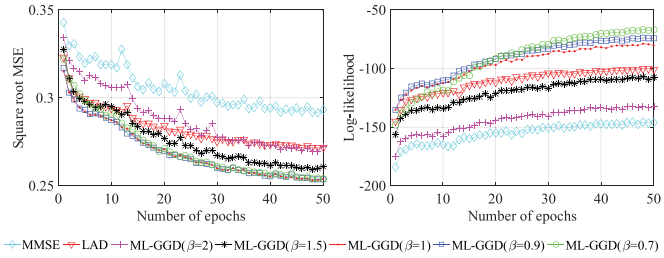


Fig. 6. A comparison of learning curves among MMSE-DNN, LAD-DNN and ML-GGD-DNN with different shape factors on the cross-validation set. The left figure shows the variation curves of the square root of the MSE on the cross-validation set with respect to the epoch. The right figure shows the variation curves of the log-likelihood on the cross-validation set with respect to the epoch.

TABLE II
AVERAGE LOG-LIKELIHOODS AND SQUARE ROOTS OF MSES OF ML-GGD-DNN SYSTEMS ON THE TEST SET AVERAGED OVER ALL EXAMINED SNRS AND 14 UNSEEN NOISE TYPES

	$\beta = 2$	$\beta = 1.5$	$\beta = 1$	$\beta = 0.9$	$\beta = 0.7$
Square root MSE	0.65	0.64	0.64	0.65	0.66
Log-likelihood	-375	-322	-267	-266	-254

Accordingly, the scale factor α in Eq. (11) for the computation of log-likelihoods on the cross-validation set is obtained by inserting a mini-batch of M sample frames from the training set into Eq. (20). Based on Fig. 6, several observations can be made. First, the learning curves of the square roots of MSEs reveal that using LAD-DNN to minimize the absolute error on the training data consistently generated smaller MSEs on the cross-validation set than using MMSE-DNN to minimize the MSE on the training data, which implies that LAD-DNN can achieve better robustness and generalization capability than MMSE-DNN. Second, MMSE-DNN and LAD-DNN consistently yielded greater MSE values and a slower convergence than ML-GD-DNN (namely, ML-GGD-DNN with $\beta = 2$) and ML-LD-DNN (namely, ML-GGD-DNN with $\beta = 1$), respectively. This result implies that the objective function based on the heteroscedastic error model can achieve a better generalization capability and faster convergence than the homoscedastic error model. Third, the MSE is minimized for MMSE-DNN training, whereas the log-likelihood is maximized for ML-GGD-DNN training. From the learning curves of the square roots of the MSEs, all ML-GGD-DNNs with different shape factors clearly achieved a better and faster convergence than MMSE-DNN. In addition, the MSEs of ML-GGD-DNNs using different shape factors could not be well differentiated. Conversely, the learning curves of log-likelihood were clearly distinguished for ML-GGD-DNNs with different shape factors. The log-likelihood of ML-GGD-DNN on the cross-validation set was increased as β decreased from 2 to 0.7. This is consistent with the variation tendency of the evaluation metrics on the test set with respect to the shape factors shown in Table III. Accordingly, these results indicate that ML-GGD-DNN using the scale factor has a better generalization capability than MMSE-DNN.

Table II shows the average log-likelihoods and square roots of MSEs of ML-GGD-DNN systems on the test set averaged over all examined SNRs and 14 unseen noise types. By comparing

TABLE III
AVERAGE PERFORMANCE COMPARISON OF DIFFERENT SYSTEMS ON THE TEST SET AT DIFFERENT SNRS AVERAGED OVER 14 UNSEEN NOISE TYPES

Performance measurement: STOI

SNR(dB)	-5	0	5	10	15	20	Average
Noisy	0.623	0.720	0.812	0.887	0.939	0.971	0.825
OM-LSA [7]	0.593	0.708	0.812	0.890	0.942	0.972	0.819
MMSE	0.670	0.775	0.855	0.908	0.940	**0.960**	0.851
LAD	0.700*	0.795*	0.869*	0.921*	0.953*	0.972*	0.868*
ML-GGD:							
$\beta = 2$	0.705	0.800	0.872	0.920	0.949	0.966	0.869
$\beta = 1.5$	0.708	0.804	0.876	0.924	0.954	0.971	0.873
$\beta = 1$	0.711	0.806	0.879	0.928	0.958	0.975	0.876
$\beta = 0.9$	0.710	0.806	0.879	0.928	0.958	0.976	0.876
$\beta = 0.7$	0.711	0.808	0.881	0.929	0.959	0.977	0.877
Fusion	0.718	0.812	0.883	0.930	0.960	0.977	0.880

Performance measurement: SegSNR(dB)

SNR(dB)	-5	0	5	10	15	20	Average
Noisy	-6.52	-4.03	-0.94	2.59	6.39	10.40	1.31
OM-LSA [7]	-0.10	2.20	4.78	7.60	10.71	14.02	6.54
MMSE	0.19	1.84	3.71	5.60	7.40	**8.95**	4.61
LAD	0.21	2.30*	4.60*	6.96*	9.32*	11.46*	5.81*
ML-GGD:							
$\beta = 2$	0.05	2.05	4.19	6.30	8.27	9.94	5.13
$\beta = 1.5$	0.28	2.34	4.61	6.88	9.06	11.01	5.70
$\beta = 1$	0.38	2.57	5.01	7.51	9.99	12.31	6.29
$\beta = 0.9$	0.37	2.60	5.06	7.59	10.13	12.52	6.38
$\beta = 0.7$	0.19	2.51	5.05	7.65	10.26	12.78	6.41
Fusion	0.42	2.68	5.19	7.76	10.37	12.87	6.55

Performance measurement: LSD(dB)

SNR(dB)	-5	0	5	10	15	20	Average
Noisy	13.54	10.17	7.44	5.23	3.48	2.16	7.00
OM-LSA [7]	5.74	4.34	3.32	2.49	1.79	1.22	3.15
MMSE	5.36	4.13	3.28	2.64	2.19	1.95	3.26
LAD	5.00*	3.89*	2.97*	2.27*	1.77*	1.45*	2.89*
ML-GGD:							
$\beta = 2$	5.12	4.05	3.17	2.51	2.04	1.77	3.11
$\beta = 1.5$	5.00	3.91	3.00	2.31	1.83	1.53	2.93
$\beta = 1$	4.89	3.77	2.85	2.16	1.66	1.32	2.78
$\beta = 0.9$	4.90	3.77	2.85	2.14	1.63	1.28	2.76
$\beta = 0.7$	5.13	3.90	2.91	2.15	1.60	1.21	2.82
Fusion	4.89	3.74	2.81	2.10	1.58	1.23	2.72

Performance measurement: PESQ

SNR(dB)	-5	0	5	10	15	20	Average
Noisy	1.51	1.83	2.18	2.52	2.86	3.19	2.35
OM-LSA [7]	1.69	2.14	2.57	2.94	3.26	3.54	2.69
MMSE	2.01	2.41	2.76	3.06	3.32	3.55	2.85
LAD	2.10*	2.46*	2.80*	3.09*	3.35*	3.58*	2.90*
ML-GGD:							
$\beta = 2$	2.12	2.49	2.82	3.12	3.38	3.61	2.92
$\beta = 1.5$	2.14	2.50	2.83	3.12	3.39	3.62	2.93
$\beta = 1$	2.14	2.50	2.83	3.13	3.39	3.62	2.93
$\beta = 0.9$	2.15	2.50	2.83	3.12	3.39	3.62	2.93
$\beta = 0.7$	2.16	2.51	2.83	3.12	3.38	3.61	2.94
Fusion	2.17	2.52	2.85	3.14	3.40	3.63	2.95

Table II and the last column in Table III, we can observe that the MSE could not accurately evaluate the objective metrics. In contrast, the variation tendencies of all objective metrics with respect to β were closer to those of the log-likelihood measure rather than the MSE measure. This suggests that our proposed new optimization function is highly correlated with objective measures for speech enhancement.

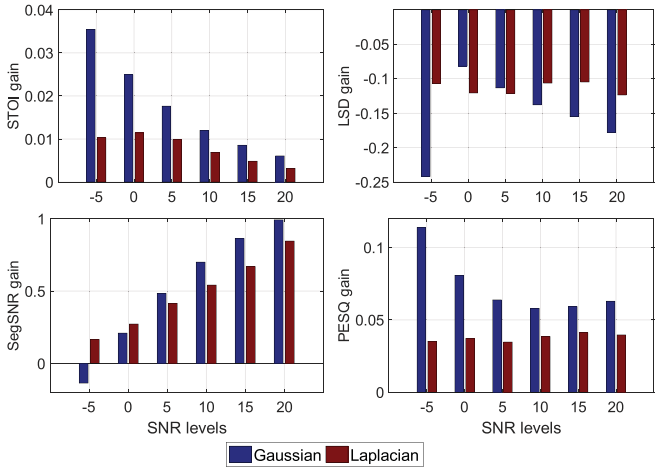


Fig. 7. Average performance comparison between ML-GGD-DNNs ($\beta = 2$ for Gaussian and $\beta = 1$ for Laplacian) with heteroscedasticity and homoscedasticity assumptions for the STOI/LSD/SegSNR/PESQ measures at different SNRs averaged over 14 unseen noise types.

D. Influence of the Scale Factors

Fig. 6 demonstrates the improved performance of ML-GGD-DNN using the scale factors based on the learning curve of the square root of the MSE. To further verify its effectiveness for enhancing performance, we present an average performance comparison between ML-GGD-DNNs with heteroscedasticity and homoscedasticity assumptions in Fig. 7, where "Gaussian" denotes the gains of ML-GD-DNN (namely, ML-GGD-DNN with $\beta = 2$) with the heteroscedasticity assumption over MMSE-DNN with the homoscedasticity assumption, and "Laplacian" denotes the gains of ML-LD-DNN (namely, ML-GGD-DNN with $\beta = 1$) with the heteroscedasticity assumption over LAD-DNN with the homoscedasticity assumption. The corresponding evaluation metrics of the unprocessed system denoted as Noisy, MMSE-DNN system and LAD-DNN system are shown in Table III. Fig. 7 shows that for both Gaussian and Laplacian densities, the heteroscedastic models (ML-GD-DNN and ML-LD-DNN) outperformed the corresponding homoscedastic models (MMSE-DNN and LAD-DNN) in most cases with only one exception for the SegSNR measure of GD at -5 dB SNR. The reason for this exception is that the MMSE-DNN with the homoscedastic Gaussian assumption aggressively reduces noise, especially in very low SNR cases, which yields better SegSNR scores but leads to larger speech distortions. Generally, the heteroscedastic model with the optimization of scale factors generated better speech intelligibility (STOI gain), fewer speech distortions (LSD gain), better speech quality (PESQ gain), and greater noise reductions (SegSNR gain). In addition, the performance gains for Gaussian were often larger than those for Laplacian. This result is reasonable because according to the statistical analysis in Section III-A, the LD is closer than the GD to the true distribution of the prediction error. Accordingly, it is potentially easier to achieve performance gains for the Gaussian case. Overall, we can conclude that the heteroscedasticity assumption is one main contributor to the success of ML-GGD-DNN by using the scale factors.

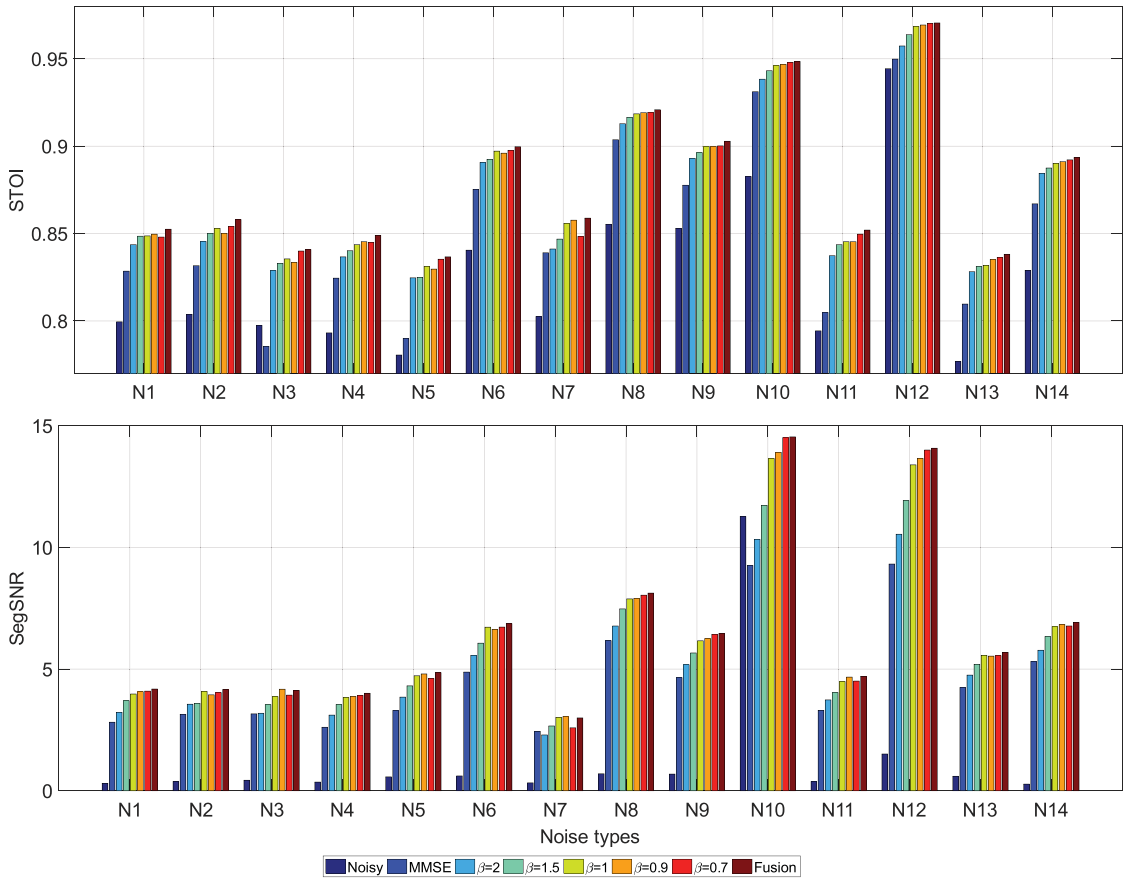


Fig. 8. Average STOI and SegSNR comparison of different systems among 14 unseen noise types averaged over all examined SNRs.

E. Influence of the Shape Factors

If we assume that all the scale factors are the same, namely, the homoscedasticity assumption, then the optimization of the proposed objective function in Eq. (10) is regressed to minimizing the classic β-norm function. In this case, we can simply discuss the influence of the shape factor β. Here, we perform a comparison between LAD-DNN ($\beta = 1$) and MMSE-DNN ($\beta = 2$), as shown in Table III. An asterisk indicates that the score of LAD-DNN was significantly higher than that of MMSE-DNN in a paired one-sided t-test [69] (p-value ≤ 0.05). We notice that MMSE-DNN suffered in STOI and SegSNR compared to the unprocessed system at 20dB SNR. This is because the intelligibility of noisy speech is high in such low-noise conditions, and MMSE-DNN could easily introduce some artifacts and cause inevitable speech distortions. In contrast, LAD-DNN showed significant advantages over the MMSE-DNN across all evaluation metrics, and consistent improvements were observed over all SNR levels. For example, the STOI increased from 0.670 to 0.700 at -5 dB SNR and from 0.775 to 0.795 at 0 dB SNR, gains of 2.51 dB were achieved in SegSNR at 20 dB SNR, and the STOI of 0.972 was better than those of both MMSE-DNN and the unprocessed system at 20dB SNR. All these results indicate that the super-Gaussian model (LAD-DNN) is more robust and yields better generalization capability than the conventional Gaussian model (MMSE-DNN).

For the more general case, namely, the heteroscedasticity assumption with different scale factors at different vector components, we conducted experiments on the shape factors $\beta < 2$ of ML-GGD-DNN based on the elaborated statistical analysis shown in Section III-A that the prediction errors follow super-Gaussian distributions. It is impractical to show the results of all shape factors of $\beta < 2$; therefore, we select only several representative values to show the relationship between the speech enhancement performance and shape factors. As shown in Table III, with decreasing shape factor β, better STOIs were achieved and SegSNRs were first improved and then decreased for all SNR levels. Additionally, the STOI was saturated when $\beta = 0.7$ for all SNR levels. However, there were minor differences for the best configuration of SegSNR for each SNR level. For example, SegSNR yielded the best results with $\beta = 1$ at low SNRs (e.g., -5 dB) and $\beta = 0.7$ at high SNRs (e.g., 20 dB), respectively.

To examine the effects of different noise scenarios, Fig. 8 shows the average STOI and SegSNR of different enhancement systems among 14 unseen noise types averaged over all SNR levels. From Table III and Fig. 8, we can observe that both the STOI and SegSNR of the systems based on the GD were worse than those of the systems based on the super-Gaussian distribution for all SNR levels and unseen noise types, indicating that the GD cannot model the prediction errors well. Consequently, both MMSE-DNN and ML-GD-DNN lack robustness

since they adopt GD to model the prediction errors. All of the results indicate that the objective function based on the super-Gaussian error model is more robust and yields better generalization capability than that based on the Gaussian error model.We can conclude that the super-Gaussian assumption is another main contributor to the success of ML-GGD-DNN with $\beta < 2$.

F. Post-Processing With Fusion

In this section, the goal is to further improve the performance via post-processing of the top-performing systems. Here, we provide a post-processing strategy by fusion techniques. Fusion techniques [70], also named ensemble or aggregation techniques, precisely aim at combining several methods to better solve a given problem. Transposed to the context of speech enhancement, fusion [71], [72] is opposed to selection as it consists in using several speech enhancement methods and combining their solutions rather than selecting the best solution according to some criterion. To combine the advantages of the top-performing systems, we conducted a fusion via the simple average of enhanced LPS features from the three top-performing systems ($\beta = 1$, $\beta = 0.9$ and $\beta = 0.7$) as follows:

$$\hat{\boldsymbol{x}}_n = \frac{\hat{\boldsymbol{x}}_n^1 + \hat{\boldsymbol{x}}_n^2 + \hat{\boldsymbol{x}}_n^3}{3}, \tag{21}$$

where $\hat{\boldsymbol{x}}_n^1$, $\hat{\boldsymbol{x}}_n^2$ and $\hat{\boldsymbol{x}}_n^3$ are the n-th D-dimensional enhanced LPS feature vectors of the three top-performing systems, respectively, and $\hat{\boldsymbol{x}}_n$ is the fusion result. From Table III and Fig. 8, we notice that the fusion system consistently generated the best results for almost all evaluation metrics, SNR levels and unseen noise types. These results demonstrate the complementarity of different settings of β. Moreover, the advantages of the top-performing systems were fully utilized by the fusion technique. Note that although the fusion increases the computation, it provides a solution to offline applications in which the best overall performance of different measures compared to systems with a single shape factor can be achieved regardless of algorithm complexity.

G. Overall Comparison

As a summary, the average performances of the different systems on the test set averaged over all examined SNRs and 14 unseen noise types are listed in the last column in Table III. Our proposed ML-GGD-DNN consistently outperformed MMSE-DNN in terms of all 4 measures. On average, improvements of 0.025 in STOI, 1.77dB in SegSNR, 0.5dB in LSD, and 0.08 in PESQ were obtained for ML-GGD-DNN with $\beta = 0.9$. More importantly, the ML-GGD-DNN systems with $\beta < 2$ yielded consistent and significant improvements in a paired one-sided t-test (p-value ≤ 0.05) over both the MMSE-DNN and ML-GD-DNN systems across all evaluation metrics, demonstrating the better robustness and generalization capability of the super-Gaussian error model. Moreover, the fusion system of $\beta = 1$ $\beta = 0.9$ and $\beta = 0.7$ settings yielded additional improvements for all settings.

TABLE IV
OVERALL COMPARISON OF THE AVERAGE PERFORMANCE ON THE TEST SET AVERAGED OVER ALL THE UNSEEN SNRS AND 14 UNSEEN NOISE TYPES

	Noisy	MMSE	ML-GGD					
			$\beta = 2$	$\beta = 1.5$	$\beta = 1$	$\beta = 0.9$	$\beta = 0.7$	Fusion
STOI	0.716	0.773	0.781	0.784	0.788	0.789	0.790	0.795
SegSNR	-3.52	1.89	2.19	2.47	2.81	2.73	2.75	2.89
LSD	10.69	4.36	4.21	4.09	3.90	4.04	4.01	3.92
PESQ	1.86	2.42	2.47	2.48	2.48	2.48	2.49	2.51

Although matching the SNR level is the least critical aspect in acquiring good performance for DNN-based speech enhancement algorithms compared to matching the noise type and target speaker [73], to consider a more realistic condition in which both the noise types and SNR levels of the training and test sets are mismatched, we also present a comparison of the average performance for unseen SNR levels. The corresponding test set was built with the 192 utterances from the core test set of TIMIT corrupted with the same 14 unseen noise types at six unseen levels of SNRs, i.e., -9 dB, -6 dB, -3 dB, 3 dB, 6 dB and 9 dB. Table IV shows the average performances of different systems on this test set averaged over all the unseen SNRs and 14 unseen noise types. The improved performance of ML-GGD-DNN at unseen SNR levels is also demonstrated.

Fig. 9 shows a spectrogram comparison of utterances corrupted by four representative noise types. Although MMSE-DNN could perform well in terms of noise reduction, large speech distortions and even removal of some speech segments exist. Our proposed ML-GGD-DNN ($\beta = 0.9$) can satisfactorily alleviate these problems, resulting in less speech distortions and better speech preservation, for example, as shown in Fig. 9, the black solid box areas. In summary, the proposed ML-GGD-DNN can simultaneously improve speech intelligibility and quality over MMSE-DNN.

H. Extension to Speech Dereverberation

Although this paper primarily explores ML-GGD-DNN for DNN-based speech enhancement, the general ML framework for DNN regression presented in Section II can be applied to many other regression tasks, where the specific form of error density should be selected based on a statistical analysis of the prediction errors. To provide strong support for the versatility of our proposed extended ML framework shown in Section II, we applied the ML framework to DNN-based speech dereverberation [53]. Because the statistical analysis reveals the same super-Gaussian and heteroscedasticity of the prediction error distributions in DNN-based speech dereverberation as in DNN-based speech enhancement, we applied the ML-GGD-DNN framework directly to speech dereverberation. A series of experiments were performed on the training set consisting of 7138 reverberant utterances generated by convolving the anechoic speech data from the WSJ0 [74] SI-84 training set with a series of impulse responses recorded from several different rooms in the REVERB challenge [75]. The test set was created by convolving the 330 utterances from the WSJ0 5K test set with another three different impulse responses. The results shown in Table V demonstrate the effectiveness of the ML framework for speech dereverberation.

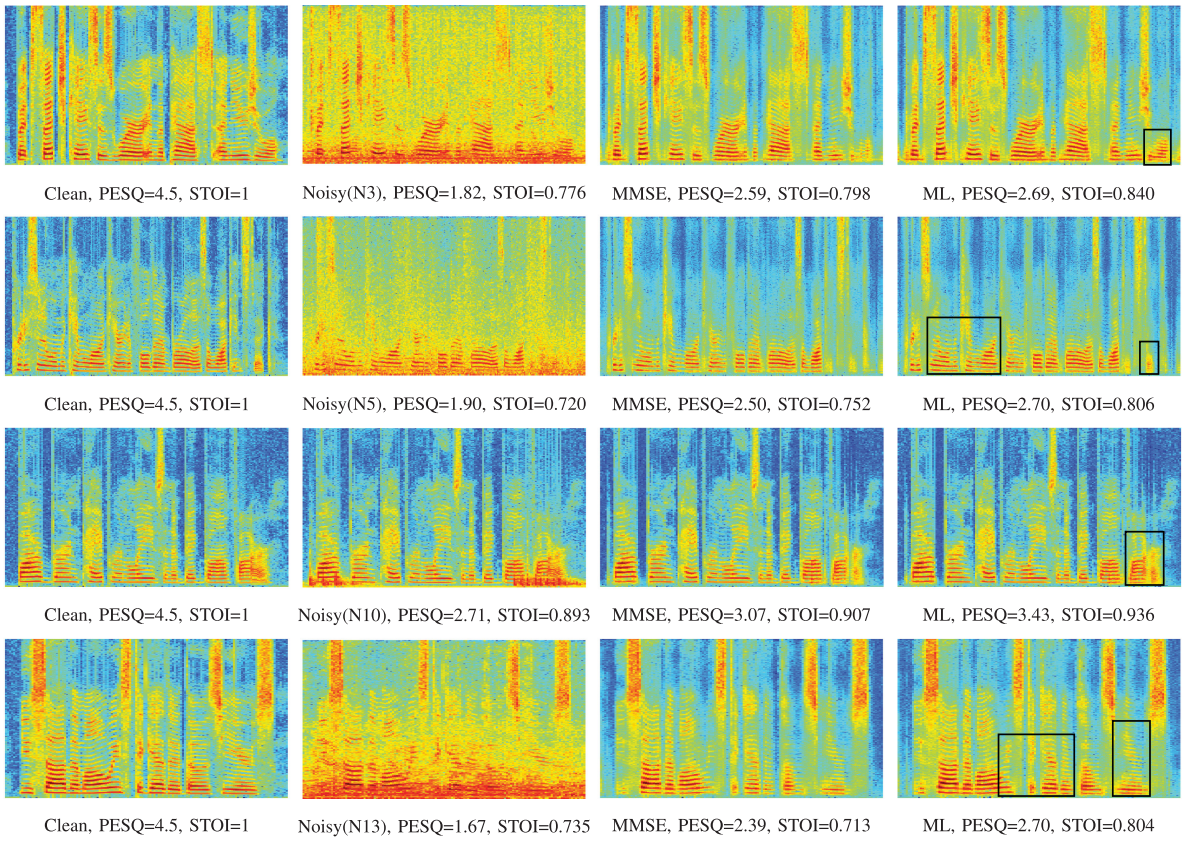


Fig. 9. The spectrograms of utterances corrupted by N3 (Destroyer Operations), N5 (Factory1), N10 (Machine Gun), and N13 (Speech Babble) at 5 dB. Each row corresponds to one example set with the clean speech, noisy speech, MMSE-DNN and ML-GGD-DNN ($\beta = 0.9$) enhanced speech.

TABLE V
COMPARISON OF THE AVERAGE PERFORMANCE ON THE TEST SET

	Noisy	MMSE	ML-GGD				
			$\beta = 2$	$\beta = 1.5$	$\beta = 1$	$\beta = 0.9$	$\beta = 0.7$
STOI	0.790	0.855	0.867	0.870	0.873	0.874	0.875
PESQ	2.198	2.629	2.709	2.710	2.713	2.713	2.713
LSD	4.26	3.35	3.16	3.14	3.10	3.07	3.04

V. CONCLUSION

In this paper, we have proposed an ML approach to modeling error vectors in regression DNN-based speech enhancement under the assumption that each dimension component of the prediction error vector is statistically independent and follows a GGD. Then, the conventional MSE function can now be replaced with the log-likelihood function as the new DNN optimization objective. The proposed ML approach is general and applicable to various DNN-based regression tasks. Its effectiveness for DNN-based speech enhancement and speech dereverberation tasks has been demonstrated. More specifically, the proposed ML objective function, assuming super-Gaussian heteroscedasticity, can achieve better robustness and generalization capability than the conventional MMSE criterion based on a homoscedastic Gaussian error assumption with performance improvements in both speech intelligibility and quality. Finally, estimating different shape factors for different prediction error vector components and applying the general ML framework to other DNN-based regression tasks are also topics of interest for our future studies.

REFERENCES

[1] J. Benesty, S. Makino, and J. D. Chen, *Speech Enhancement*. Berlin, Germany: Springer, 2005, ch. 1.
[2] P. C. Loizou, *Speech Enhancement: Theory and Practice*. Boca Raton, FL, USA: CRC Press, 2013, ch. 3.
[3] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 27, no. 2, pp. 113–120, Apr. 1979.
[4] J. S. Lim and A. V. Oppenheim, "All-pole modeling of degraded speech," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 26, no. 3, pp. 197–210, Jun. 1978.
[5] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 33, no. 2, pp. 443–445, Apr. 1985.
[6] I. Cohen and B. Berdugo, "Speech enhancement for non-stationary noise environments," *Signal Process.*, vol. 81, no. 11, pp. 2403–2418, 2001.
[7] I. Cohen, "Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 5, pp. 466–475, Sep. 2003.
[8] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.
[9] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
[10] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, 2015, Art. no. 436.
[11] E. M. Grais, M. U. Sen, and H. Erdogan, "Deep neural networks for single channel source separation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 3734–3738.

[12] Y. Wang, A. Narayanan, and D. Wang, "On training targets for supervised speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 1849–1858, Dec. 2014.

[13] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Process. Lett.*, vol. 21, no. 1, pp. 65–68, Jan. 2014.

[14] 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, Lang. Process.*, vol. 23, no. 1, pp. 7–19, Jan. 2015.

[15] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Deep learning for monaural speech separation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 1562–1566.

[16] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint optimization of masks and deep recurrent neural networks for monaural source separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 12, pp. 2136–2147, Dec. 2015.

[17] F. Weninger *et al.*, "Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR," in *Proc. Int. Conf. Latent Variable Anal. Signal Separation*, 2015, pp. 91–99.

[18] S.-W. Fu, Y. Tsao, X. Lu, and H. Kawai, "Raw waveform-based speech enhancement by fully convolutional networks," in *Proc. Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, 2017, pp. 006–012.

[19] S.-W. Fu, Y. Tsao, and X. Lu, "SNR-aware convolutional neural network modeling for speech enhancement," in *Proc. INTERSPEECH*, 2016, pp. 3768–3772.

[20] Y. Wang and D. Wang, "Towards scaling up classification-based speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 7, pp. 1381–1390, Jul. 2013.

[21] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 708–712.

[22] D. S. Williamson, Y. Wang, and D. Wang, "Complex ratio masking for monaural speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 3, pp. 483–492, Mar. 2016.

[23] G.-X. Wang, C.-C. Hsu, and J.-T. Chien, "Discriminative deep recurrent neural networks for monaural speech separation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2016, pp. 2544–2548.

[24] Y. Sun, L. Zhu, J. A. Chambers, and S. M. Naqvi, "Monaural source separation based on adaptive discriminative criterion in neural networks," in *Proc. 22nd Int. Conf. Digit. Signal Process.*, 2017, pp. 1–5.

[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, pp. 1508–1512.

[26] B. Xia and C. Bao, "Wiener filtering based speech enhancement with weighted denoising auto-encoder and noise classification," *Speech Commun.*, vol. 60, pp. 13–29, 2014.

[27] P. G. Shivakumar and P. G. Georgiou, "Perception optimized deep denoising autoencoders for speech enhancement," in *Proc. INTERSPEECH*, 2016, pp. 3743–3747.

[28] W. Han, X. Zhang, G. Min, X. Zhou, and W. Zhang, "Perceptual weighting deep neural networks for single-channel speech enhancement," in *Proc. 12th World Congr. Intell. Control Autom.*, 2016, pp. 446–450.

[29] Q. Liu *et al.*, "A perceptually-weighted deep neural network for monaural speech enhancement in various background noise conditions," in *Proc. 25th Eur. Signal Process. Conf.*, 2017, pp. 1270–1274.

[30] T. G. Kang, J. W. Shin, and N. S. Kim, "DNN-based monaural speech enhancement with temporal and spectral variations equalization," *Digit. Signal Process.*, vol. 74, pp. 102–110, 2018.

[31] S.-W. Fu, T.-Y. Hu, Y. Tsao, and X. Lu, "Complex spectrogram enhancement by convolutional neural network with multi-metrics learning," in *Proc. IEEE 27th Int. Workshop Mach. Learn. Signal Process.*, 2017, pp. 1–6.

[32] S.-W. Fu, T.-W. Wang, Y. Tsao, X. Lu, and H. Kawai, "End-to-end waveform utterance enhancement for direct evaluation metrics optimization by fully convolutional neural networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 9, pp. 1570–1584, Sep. 2018.

[33] M. Kolbcek, Z.-H. Tan, and J. Jensen, "Monaural speech enhancement using deep neural networks by maximizing a short-time objective intelligibility measure," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 5059–5063.

[34] Y. Zhao, B. Xu, R. Giri, and T. Zhang, "Perceptually guided speech enhancement using deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 5074–5078.

[35] H. Zhang, X. Zhang, and G. Gao, "Training supervised speech separation system to improve STOI and PESQ directly," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 5374–5378.

[36] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, Sep. 2011.

[37] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2001, vol. 2, pp. 749–752.

[38] Y. Koizumi, K. Niwa, Y. Hioka, K. Koabayashi, and Y. Haneda, "DNN-based source enhancement to increase objective sound quality assessment score," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 10, pp. 1780–1792, Oct. 2018.

[39] C. M. Bishop, *Pattern Recognition and Machine Learning*. Berlin, Germany: Springer, 2006, ch. 3.

[40] C. M. Bishop, "Mixture density networks," Tech. Rep., Citeseer, 1994.

[41] H. Jacqmin-Gadda, S. Sibillot, C. Proust, J.-M. Molina, and R. Thiébaut, "Robustness of the linear mixed model to misspecified error distribution," *Comput. Statist. Data Anal.*, vol. 51, no. 10, pp. 5142–5154, 2007.

[42] A. Hyvärinen, K. Zhang, S. Shimizu, and P. O. Hoyer, "Estimation of a structural vector autoregression model using non-Gaussianity," *J. Mach. Learn. Res.*, vol. 11, pp. 1709–1731, 2010.

[43] G. Bassett, Jr., and R. Koenker, "Asymptotic theory of least absolute error regression," *J. Amer. Statistical Assoc.*, vol. 73, no. 363, pp. 618–622, 1978.

[44] S. C. Narula, P. H. Saldiva, C. D. Andre, S. N. Elian, A. F. Ferreira, and V. Capelozzi, "The minimum sum of absolute errors regression: A robust alternative to the least squares regression," *Statist. Med.*, vol. 18, no. 11, pp. 1401–1417, 1999.

[45] P. Kumar and J. N. Singh, "Regression model estimation using least absolute deviations, least squares deviations and minimax absolute deviations criteria," *Int. J. Comput. Sci. Electron. Eng.*, vol. 3, no. 4, pp. 2320–4028, 2015.

[46] T. Gao, J. Du, L.-R. Dai, and C.-H. Lee, "A unified DNN approach to speaker-dependent simultaneous speech enhancement and speech separation in low SNR environments," *Speech Commun.*, vol. 95, pp. 28–39, 2017.

[47] S. Gazor and W. Zhang, "Speech probability distribution," *IEEE Signal Process. Lett.*, vol. 10, no. 7, pp. 204–207, Jul. 2003.

[48] K. Kinoshita, M. Delcroix, A. Ogawa, T. Higuchi, and T. Nakatani, "Deep mixture density network for statistical model-based feature enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2017, pp. 251–255.

[49] L. Chai, J. Du, and Y.-N. Wang, "Gaussian density guided deep neural network for single-channel speech enhancement," in *Proc. IEEE 27th Int. Workshop Mach. Learn. Signal Process.*, 2017, pp. 1–6.

[50] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[51] Y. Zhu and J. Du, "Sliding line point regression for shape robust scene text detection," in *Proc. Int. Conf. Pattern Recognit (ICPR).*, 2018, pp. 3735–3740.

[52] J. Du, "Irrelevant variability normalization via hierarchical deep neural networks for online handwritten Chinese character recognition," in *Proc. Int. Conf. Frontiers Handwriting Recognit.*, 2014, pp. 303–308.

[53] K. Han, Y. Wang, D. Wang, W. S. Woods, I. Merks, and T. Zhang, "Learning spectral mapping for speech dereverberation and denoising," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 6, pp. 982–992, Jun. 2015.

[54] J. Du, Y. Tu, C.-H. Lee, and L.-R. Dai, "A regression approach to single-channel speech separation via high-resolution deep neural networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 8, pp. 1424–1437, Aug. 2016.

[55] O. M. Kvalheim, F. Brakstad, and Y. Liang, "Preprocessing of analytical profiles in the presence of homoscedastic or heteroscedastic noise," *Analytical Chemistry*, vol. 66, no. 1, pp. 43–51, 1994.

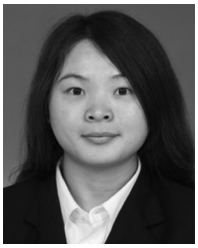
[56] K. P. Balanda and H. MacGillivray, "Kurtosis: A critical review," *Amer. Statistician*, vol. 42, no. 2, pp. 111–119, 1988.

[57] D. A. Nix and A. S. Weigend, "Learning local error bars for nonlinear regression," in *Proc. Adv. Neural Inf. Process. Syst.*, 1995, pp. 489–496.

[58] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. COMPSTAT*, 2010, pp. 177–186.

[59] R. Hecht-Nielsen, "Theory of the backpropagation neural network," in *Proc. Neural Netw. Perception*, 1992, pp. 65–93.

[60] P. J. Werbos, "Backpropagation through time: What it does and how to do it," in *Proc. IEEE*, vol. 78, no. 10, pp. 1550–1560, Oct. 1990.
[61] D. Pollard, "Asymptotics for least absolute deviation regression estimators," *Econometric Theory*, vol. 7, no. 2, pp. 186–199, 1991.
[62] E. Denoël and J.-P. Solvay, "Linear prediction of speech with a least absolute error criterion," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-33, no. 6, pp. 1397–1403, Dec. 1985.
[63] H.-C. Ong and S. H. Quah, "Error backpropagation using least absolute criterion," *Int. J. Comput. Math.*, vol. 82, no. 3, pp. 301–312, 2005.
[64] G. Hu, "100 nonspeech environmental sounds," 2004. [Online]. Available: http://web.cse.ohio-state.edu/pnl/corpus/HuNonspeech/HuCorpus.html
[65] V. Zue, S. Seneff, and J. Glass, "Speech database development at MIT: Timit and beyond," *Speech Commun.*, vol. 9, pp. 351–356, 1990.
[66] A. Varga and H. J. Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 12, no. 3, pp. 247–251, 1993.
[67] Q. Wang, J. Du, and L.-R. Dai, "Boosting DNN-based speech enhancement via explicit transformations," in *Proc. Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, 2016, pp. 1–4.
[68] J. Du and Q. Huo, "A speech enhancement approach using piecewise linear approximation of an explicit model of environmental distortions," in *Proc. INTERSPEECH*, 2008, pp. 569–572.
[69] H. Hsu and P. A. Lachenbruch, "Paired t test," Wiley encyclopedia of clinical trials, pp. 1–3, 2007.
[70] I. Bloch *et al.*, "Fusion: General concepts and characteristics," *Int. J. Intell. Syst.*, vol. 16, no. 10, pp. 1107–1134, 2001.
[71] J. Le Roux, S. Watanabe, and J. R. Hershey, "Ensemble learning for speech enhancement," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2013, pp. 1–4.
[72] X. Jaureguiberry, E. Vincent, and G. Richard, "Fusion methods for speech enhancement and audio source separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 7, pp. 1266–1279, Jul. 2016.
[73] M. Kolbk, Z.-H. Tan, J. Jensen, M. Kolbk, Z.-H. Tan, and J. Jensen, "Speech intelligibility potential of general and specialized deep neural network based speech enhancement systems," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 1, pp. 153–167, Jan. 2017.
[74] J. Garofalo, D. Graff, D. Paul, and D. Pallett, "Csr-i (wsj0) complete," Linguistic Data Consortium, Philadelphia, PA, USA, 2007.
[75] K. Kinoshita *et al.*, "The REVERB challenge: A common evaluation framework for dereverberation and recognition of reverberant speech," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2013, pp. 1–4.



Li Chai received the B.S. degree from the Department of Electronic Science and Technology, Xidian University, Xi'an, China, in 2016. She is currently working toward the Ph.D. degree with the University of Science and Technology of China, Hefei, China. Her current research mainly includes speech enhancement and robust speech recognition.

Jun Du received the B.Eng. and Ph.D. degrees from the Department of Electronic Engineering and Information Science, University of Science and Technology of China (USTC), Hefei, China, in 2004 and 2009, respectively. From July 2009 to June 2010, he worked with iFlytek Research on speech recognition. From July 2010 to January 2013, he joined Microsoft Research Asia as an Associate Researcher, working on handwriting recognition, OCR, and speech recognition. Since February 2013, he has been with the National Engineering Laboratory for Speech and Language Information Processing, USTC.

Qing-Feng Liu received the B.Eng. and Ph.D. degrees from the Department of Electronic Engineering and Information Science, University of Science and Technology of China (USTC), Hefei, China, in 1998 and 2003, respectively. He is the Founder, CEO, and President of iFLYTEK, the Director of the National Speech and Language Engineering Laboratory of China, a Professor and Doctoral Advisor with USTC, the Director General of the Union of Speech Industry of China, and the Director General of the Union of National University Student Innovation and Entrepreneurship.

Chin-Hui Lee is a Professor with the School of Electrical and Computer Engineering, Georgia Institute of Technology. Before joining academia in 2001, he had 20 years of industrial experience, ending at Bell Laboratories, Murray Hill, NJ, USA, as a Distinguished Member of Technical Staff, and the Director of the Dialogue Systems Research Department. He has authored or coauthored more than 500 papers and 30 patents, and has been cited more than 34 000 times for his original contributions with an h-index of 80 on Google Scholar. He has received numerous awards, including the Bell Labs President's Gold Award in 1998. He also won SPS's 2006 Technical Achievement Award for "Exceptional Contributions to the Field of Automatic Speech Recognition." In 2012, he was invited by ICASSP to give a plenary talk on the future of speech recognition. In the same year, he was awarded the ISCA Medal in scientific achievement for pioneering and seminal contributions to the principles and practice of automatic speech and speaker recognition. He is a Fellow of ISCA.