applied sciences

MDPI

Article

Two-Stage Single-Channel Speech Enhancement with Multi-Frame Filtering

Shaoxiong Lin [1], Wangyou Zhang [1] and Yanmin Qian [1,2,*]

[1] X-LANCE Lab, MoE Key Lab of Artificial Intelligence, AI Institute, Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China; johnson-lin@sjtu.edu.cn (S.L.); wyz-97@sjtu.edu.cn (W.Z.)

[2] Suzhou Institute of Artificial Intelligence, Shanghai Jiao Tong University, Suzhou 215000, China

* Correspondence: yanminqian@sjtu.edu.cn

Abstract: Speech enhancement has been extensively studied and applied in the fields of automatic speech recognition (ASR), speaker recognition, etc. With the advances of deep learning, attempts to apply Deep Neural Networks (DNN) to speech enhancement have achieved remarkable results and the quality of enhanced speech has been greatly improved. In this study, we propose a two-stage model for single-channel speech enhancement. The model has two DNNs with the same architecture. In the first stage, only the first DNN is trained. In the second stage, the second DNN is trained to refine the enhanced output from the first DNN, while the first DNN is frozen. A multi-frame filter is introduced to help the second DNN reduce the distortion of the enhanced speech. Experimental results on both synthetic and real datasets show that the proposed model outperforms other enhancement models not only in terms of speech enhancement evaluation metrics and word error rate (WER), but also in its superior generalization ability. The results of the ablation experiments also demonstrate that combining the two-stage model with the multi-frame filter yields better enhancement performance and less distortion.

Keywords: speech enhancement; two-stage training; multi-frame filter

check for updates

Citation: Lin, S.; Zhang, W.; Qian, Y. Two-Stage Single-Channel Speech Enhancement with Multi-Frame Filtering. *Appl. Sci.* **2023**, *13*, 4926. https://doi.org/10.3390/app13084926

Academic Editor: Javier Hernando

Received: 26 February 2023
Revised: 30 March 2023
Accepted: 10 April 2023
Published: 14 April 2023

1. Introduction

In real-world environments, speech is always corrupted by background noise, which severely degrades speech intelligibility for human listeners and makes downstream tasks such as automatic speech recognition (ASR) more challenging. Therefore, speech enhancement, which suppresses the background noise to get clean speech, has been extensively studied for decades and numerous methods have been proposed in this field.

Two primary approaches exist for speech enhancement: single-channel and multi-channel. Single-channel speech enhancement operates on a single microphone input, while multi-channel speech enhancement utilizes multiple microphone inputs to enhance the speech signal. In this study, we mainly focus on single-channel speech enhancement, as single-channel speech signals can be collected using only one microphone, making them more common in real-life scenarios.

In the context of single-channel speech enhancement, traditional methods include spectral-subtraction algorithms [1], Wiener filtering [2], non-negative matrix factorization [3], spectrogram inversion [4], etc. These methods are generally computationally efficient and exhibit good domain generalization. However, they assume that the background noise is stationary, that is, its spectral and temporal characteristics remain constant over time, enabling accurate estimation of its statistical properties. In the presence of non-stationary background noise, however, accurately capturing its statistical properties becomes challenging as the noise continuously changes over time, leading to a substantial performance degradation of these methods [5].

In recent years, methods based on Deep Neural Networks (DNN) have shown their superior capability in dealing with non-stationary noise compared with traditional methods in both single-channel and multi-channel speech enhancement. These methods typically train DNNs to learn a mapping from noisy speech to clean speech. The speech enhanced by these methods tends to achieve high enhancement metrics scores. However, a recent study [6,7] has shown that the DNNs can introduce distortions into the enhanced speech, which will cause performance degradation of downstream tasks, such as ASR.

To minimize such distortions, in multi-channel speech enhancement, recent studies [8,9] have combined DNNs with low-distortion multi-channel filters. These studies have achieved excellent results not only on speech enhancement but also on the downstream ASR task.

Although [8,9] have demonstrated that low-distortion filters can improve the performance of neural networks in multi-channel scenarios, it should be noted that in multi-channel scenarios, filters can exploit the spatial information inherent in microphone arrays, which is not accessible in single-channel scenarios. Therefore, inspired by these studies, we aim to investigate the performance of combining DNNs and conventional filters on single-channel speech enhancement and the downstream ASR task. We adopt a similar two-stage framework to [8], since we believe single-stage networks may suffer from performance bottlenecks in recovering clean speech from degraded ones when faced with challenging scenarios.

The main difference between our method and the one proposed in [8] is that our method focuses on single-channel speech enhancement, so the networks and filters in our method are designed for single-channel speech signals. Specifically, our method consists of two DNNs, which have the same architecture but do not share the parameters, and a single-channel filter module. In the first stage, the first DNN is trained to generate an enhanced spectrum from the noisy spectrum. In the second stage, the first DNN is fixed, and its output is used to compute the single-channel filter. The filtered result and the output of the first DNN are used as extra features to guide the training of the second DNN for better enhancement.

The rest of this study is organized as follows. In Section 2, we introduce related research. The framework and DNN architecture of the proposed method, the training strategy, and the single-channel filter are introduced in Section 3. In Section 4, the experimental results and analysis are provided. Conclusions are given in Section 5.

2. Related Research

2.1. Speech Enhancement Based on Filters

Multi-channel filters are widely applied in multi-channel speech enhancement, since they can utilize the extra spatial information contained in speech signals, thus achieving good enhancement results. There has been a long history of research on how to design filters with better enhancement performance, and many filters have been proposed, among which the classical ones are the Wiener filter, minimum variance distortionless response (MVDR) [10], minimum power distortionless response (MPDR) [11], etc.

Recently, low-distortion filters for single-channel speech enhancement have been proposed, such as MFMVDR [12] and multi-frame Wiener filter (MFWF) [13]. Since single-channel signals have no additional spatial information, these single-channel filters focus on modeling the relationship between adjacent frames.

2.2. Speech Enhancement Based on DNNs

Existing speech enhancement methods can be divided into time-frequency (T-F) domain and time domain methods. For T-F domain methods, the input features are usually the magnitude or the real and imaginary (RI) components of the short-time Fourier transform (STFT) spectrum but the training targets can be different. Based on the training targets, T-F domain methods can be further divided into mask-based methods and mapping-based methods. Mask-based methods estimate a mask, which multiplies with the noisy

T-F spectrum to get the enhanced spectrum. Commonly used masks include ideal ratio mask (IRM) [14], phase-sensitive mask (PSM) [15], and complex ratio mask (CRM) [16]. Mapping-based methods estimate the enhanced spectrum directly. Common neural network architectures used in T-F domain include long short-term memory (LSTM) [17], convolutional recurrent network (CRN) [18], and U-Net [19].

Time domain methods directly estimate clean speech waveforms from noisy speech waveforms. This end-to-end manner can work around the difficult phase estimation problem and help the model learn proper representations that are suitable for enhancing speech. Conv-TasNet [20] and dual-path recurrent neural network (DPRNN) [21] are two typical works in time domain speech enhancement.

2.3. Combination of DNNs and Filters

A favorable property of multi-channel filters is that they introduce little distortion to the enhanced speech. Thus, many studies in multi-channel speech enhancement combine DNNs with multi-channel filters to obtain speech with higher quality and fewer distortions.

In [8], a two-stage approach was proposed for multi-channel speech enhancement. First, the RI components of different channels are concatenated as input features for the first DNN, which estimates the RI components of the target speech. The predicted speech is then used to compute signal statistics for filters. Many kinds of filters have been tested, such as MVDR, MPDR, multi-channel Wiener filter (MCWF), etc. Finally, the predicted speech and the result of the filter are used as extra features to train the second DNN. This model was modified into a multi-stage one in [9] and a multi-frame MCWF (MFMCWF) is used to provide the filtering result.

The framework proposed in [8] has been migrated to single-channel scenarios. In [22], a similar model is proposed for single-channel speech dereverberation and speaker separation. Since the filters used in [8] are originally designed for multi-channel speech signals, the model proposed in [22] uses a linear-prediction module to estimate a dereverberation filter based on the first DNN's output. The predicted filter then provides the dereverberated signal as extra features to help train the second DNN.

3. Method

3.1. Framework and Network Architecture

The proposed system is illustrated in Figure 1, with the objective of removing background noise while minimizing distortion, a more complex problem than simple noise removal. To tackle this challenge, we adopted a problem decomposition approach and integrated two neural networks, DNN_1 and DNN_2. DNN_1 aims to suppress noise components coarsely, while DNN_2 is responsible for further noise removal and minimizing distortion introduced during enhancement.

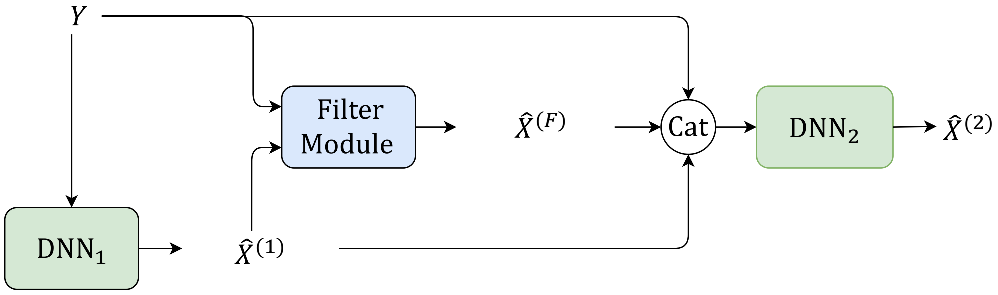

Figure 1. The framework of the proposed system.

To this end, we adopt the following two measures. Firstly, we introduced a filter module that calculates a low-distortion filter based on the enhancement result of DNN_1 ($\hat{X}^{(1)}$), and applies it to the noisy spectrum (Y). The filtered output ($\hat{X}^{(F)}$) is then concatenated with $\hat{X}^{(1)}$ and fed into DNN_2 as extra features. The purpose of this approach is two-fold. First, since the single-channel signal has no spatial information that can be exploited, we hope the result of multi-frame filters can convey the interframe correlation to DNN_2. Second, we wish the low-distortion filtered result can provide information that is complementary to the result of DNN_1, thus helping DNN_2 to produce less distorted speech, which is benifical to downstream tasks such as ASR.

Secondly, we concatenate Y with $\hat{X}^{(1)}$ and $\hat{X}^{(F)}$ as the input for DNN_2, inspired by [6]. Through theoretical analysis and experiments in [6], it was shown that adding a scaled version of the noisy signal to the enhanced signal can monotonically increase the signal-to-artifact ratio under mild conditions and improve ASR performance. Thus, we believe that concatenating Y with $\hat{X}^{(1)}$ and $\hat{X}^{(F)}$ can provide essential information for DNN_2 to reduce distortion during the training process.

We employ the TCN-DenseUNet described in [8] and modify it into a single-channel version for DNN_1 and DNN_2. Although both DNN_1 and DNN_2 adopt the TCN-DenseUNet structure, it should be noted that they do not share parameters. The main reason is that DNN_1 and DNN_2 focus on different tasks, while DNN_1 is concerned with removing background noise as much as possible, DNN_2 aims to make the most of the filter output and noisy speech to minimize distortion. We believe that not sharing parameters between the two networks can lead to better performance, which is supported to some extent by the experimental results in Section 5.4.

TCN-DenseUNet is a variant of U-Net, with a temporal convolutional network (TCN) network inserted between the encoder and decoder. The DenseNet blocks are also inserted between different layers of the encoder and decoder of the U-Net. Figure 2 shows the diagram of the TCN-DenseUNet. The encoder contains a 2D convolution layer and seven convolutional blocks, while the decoder contains seven deconvolutional blocks and a 2D deconvolution layer. Skip connections are added between the encoder and decoder. Each convolutional block consists of a 2D convolution layer, an exponential linear units (ELU) nonlinearity layer, and an instance normalization (IN) layer. The deconvolutional block has the same structure as the convolutional block, except that the 2D convolution layer is replaced with the 2D deconvolution layer. The DenseNet blocks consist of five convolutional blocks and the TCN network contains four layers, each with seven dilated convolutional blocks.

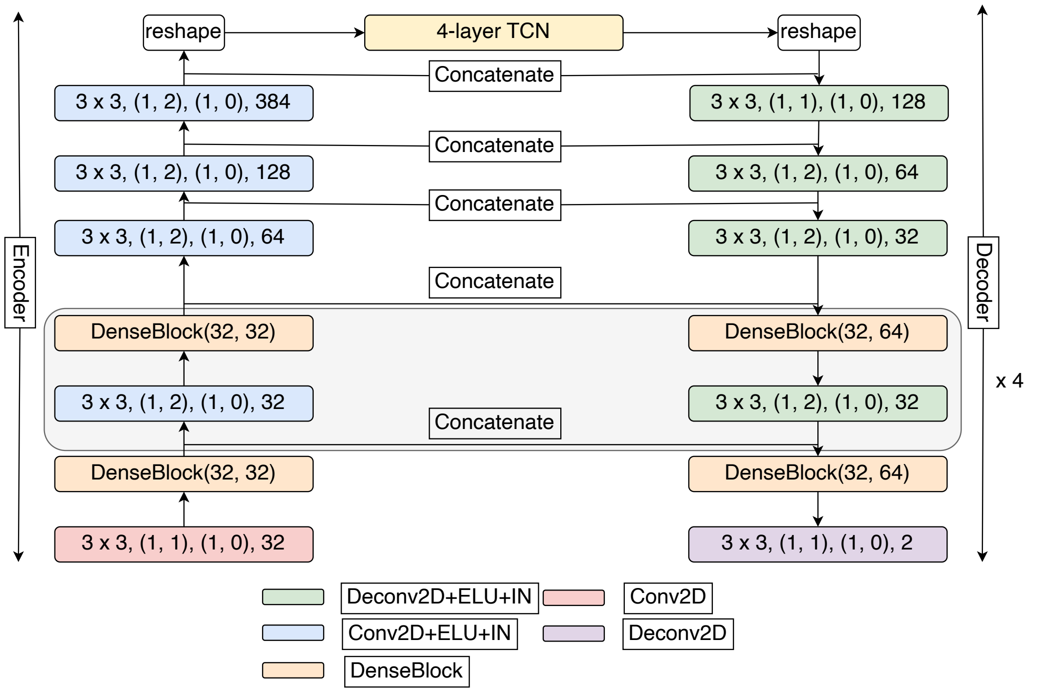

Figure 2. The diagram of TCN-DenseUNet.

The detailed setup for the TCN-DenseUNet is also shown in Figure 2. DenseNet block is represented by DenseBlock(g_1, g_2), where g_1 and g_2 are the growth rates for the first four and the last convolutional block, respectively. Other convolutional blocks are represented in the form of $(k,\ s,\ p,\ o)$, where k, s, p, o are the kernel size, stride, padding, and output channels, respectively.

3.2. Two-Stage Training Strategy

A two-stage training strategy is employed to train the system. To be specific, in the first stage, only DNN$_1$ is trained. The input is the noisy spectrum $\mathbf{Y}$, while DNN$_1$ estimates the spectrum of the clean speech. In the second stage, the well-trained DNN$_1$ is fixed, and its output $\hat{\mathbf{X}}^{(1)}$ is first fed into the Filter Module to calculate a multi-frame filter $\mathbf{h}$. $\mathbf{h}$ is then applied to $\mathbf{Y}$ to get the enhanced spectrum $\hat{\mathbf{X}}^{(F)}$. Finally, $\mathbf{Y}$, $\hat{\mathbf{X}}^{(1)}$, and $\hat{\mathbf{X}}^{(F)}$ are concatenated and fed into DNN$_2$. DNN$_2$ then outputs the final estimated spectrum of the clean speech. In a nutshell, the training stage can be formulated as:
Stage 1:

$$\hat{\mathbf{X}}^{(1)} = \mathrm{DNN}_1(\mathbf{Y}) \tag{1}$$

Stage 2:

$$\mathbf{h} = \mathrm{FilterModule}(\hat{\mathbf{X}}^{(1)}) \tag{2}$$

$$\hat{\mathbf{X}}^{(F)} = \mathbf{h}^H \mathbf{Y} \tag{3}$$

$$\hat{\mathbf{X}}^{(2)} = \mathrm{DNN}_2(\mathrm{Cat}(\mathbf{Y}, \hat{\mathbf{X}}^{(1)}, \hat{\mathbf{X}}^{(F)})) \tag{4}$$

3.3. Multi-Frame MVDR

The MFMVDR was first proposed in [12]. It considers the correlation between consecutive time-frames to obtain better enhancement performance.

The signal model for MFMVDR is as follows:

$$y(t) = x(t) + n(t), \tag{5}$$

where $y(t)$, $x(t)$, and $n(t)$ are the noisy speech, clean speech, and the additive noise, respectively. Using STFT, (5) can be rewritten as:

$$Y(t,f) = X(t,f) + N(t,f). \tag{6}$$

In order to model the interframe correlation, a L-dimensional noisy speech vector $\mathbf{y}(t,f)$ is defined as:

$$\mathbf{y}(t,f) = \begin{bmatrix} Y(t,f) & \cdots & Y(t-L+1,f) \end{bmatrix}^T, \tag{7}$$

where L is the number of frames used to calculate the MFMVDR filter. $\mathbf{x}(t,f)$ and $\mathbf{n}(t,f)$ can be defined similarly.

The formula of MFMVDR is as follows:

$$\mathbf{h}_{\mathrm{MFMVDR}}(t,f) = \frac{\mathbf{\Phi}_{\mathbf{n}}^{-1}(t,f)\mathbf{\Phi}_{\mathbf{y}}(t,f) - \mathbf{I}_{L\times L}}{\mathrm{tr}\left[\mathbf{\Phi}_{\mathbf{n}}^{-1}(t,f)\mathbf{\Phi}_{\mathbf{y}}(t,f)\right] - L}\mathbf{i}_1, \tag{8}$$

where $\mathbf{I}_{L\times L}$ is an $L \times L$ identity matrix, $\mathbf{i}_1$ is the first column of $\mathbf{I}_{L\times L}$, $\mathrm{tr}[\cdot]$ is the trace operation, and

$$\mathbf{\Phi}_{\mathbf{y}} = E[\mathbf{y}(t,f)\mathbf{y}^H(t,f)], \tag{9}$$

$$\mathbf{\Phi}_{\mathbf{n}} = E[\mathbf{n}(t,f)\mathbf{n}^H(t,f)]. \tag{10}$$

By viewing $\hat{\mathbf{X}}^{(1)}$ as $\mathbf{X}$, we can calculate $\mathbf{N} = \mathbf{Y} - \hat{\mathbf{X}}^{(1)}$. Then, we can calculate $\mathbf{\Phi}_{\mathbf{n}}$ using Equation (10), and finally, we get the MFMVDR filter.

It should be noted that calculating $\mathbf{\Phi_n}$ using Equation (10) yields a matrix of rank 1, i.e., a singular matrix, but in Equation (8) the inverse matrix of $\mathbf{\Phi_n}$ is required. To address this issue, we use the following methodology in this study. First, we calculate $\mathbf{\Phi_y}$ and $\mathbf{\Phi_n}$ using the following equations, instead of Equations (9) and (10):

$$\mathbf{\Phi_y}(t,f) = \lambda_y \mathbf{\Phi_y}(t,f-1) + (1-\lambda_y)\mathbf{y}(t,f)\mathbf{y}^H(t,f), \tag{11}$$

$$\mathbf{\Phi_v}(t,f) = \lambda_n \mathbf{\Phi_n}(t,f-1) + (1-\lambda_n)\mathbf{n}(t,f)\mathbf{n}^H(t,f), \tag{12}$$

where $0 < \lambda_y < 1$ and $0 < \lambda_n < 1$ are the forgetting factors. Second, we apply diagonal loading [23] to $\mathbf{\Phi_y}$ and $\mathbf{\Phi_n}$ to improve the robustness of training stage.

4. Experiment

4.1. Dataset and Evaluation Metrics

Experiments are performed on the WHAM! [24] dataset. WHAM! was originally designed for speech separation in noisy environments. It pairs each two-speaker mixture in the wsj0-2mix [25] dataset with a real-world noise. The noise was recorded in urban environments, such as coffee shops, restaurants, bars, office buildings, parks, etc. [24]. Meanwhile, WHAM! also provides a version for speech enhancement, where only the speech of the first speaker is mixed with the noise at SNRs randomly sampled between −6 and +3 dB. The training set, development set, and test set contain 20,000, 5000, and 3000 utterances, respectively. The training and development sets share common speakers, but the test set speakers are different. In order to evaluate the generalizability of the proposed model, we further adopt the one-channel test set from CHiME-4 [26] for evaluation. Both the simulated and the real-world noisy utterances are used.

Four widely used objective metrics are used to evaluate the enhanced speech, namely narrow-band Perceptual Evaluation of Speech Quality (PESQ-NB) [27], Short-Time Objective Intelligibility (STOI) [28], scale-invariant Source-to-Noise Ratio (SI-SNR) [29] and Signal-to-Distortion Ratio (SDR) [30]. All of these metrics are the larger the better. The Word Error Rate (WER) is used to indicate the performance of the enhanced speech on the ASR task.

4.2. Baselines and Experimental Settings

The proposed model is compared with several popular baseline models, including LSTM, Conv-TasNet, and DPRNN. Figure 3 shows the diagrams of these baseline models, and detailed descriptions of all the models are provided below.

LSTM: Three-layers of bi-directional LSTM, where each layer has 512 hidden units, followed by a linear layer with 257 output units and tanh activation function. A Dropout layer with dropout probability equal to 0.4 is introduced on the outputs of each LSTM layer except the last layer.

Conv-TasNet: The encoder and decoder are symmetric 1D convolution layers. The mask estimator comprises a layer normalization and a 1D convolution layer with a kernel size of 1 (1 × 1-conv block) and 256 output channels. It is then followed by 8 convolutional blocks with a kernel size of 3, output channels of 512, and dilation factors ranging from 1 to 2^7, which are repeated 4 times. Ultimately, a 1 × 1-conv block with a ReLU activation function is employed to estimate the mask.

DPRNN: The encoder and decoder are symmetric 1D convolution layers. Six DPRNN blocks described in [21] are used to predict the mask. The DPRNN blocks utilize intra-block and inter-block RNNs, both of which adopt residual connections. These RNNs consist of bidirectional LSTMs with 128 hidden units, linear layers with 64 output units, and layer normalization. A dropout layer with a 0.1 dropout rate is inserted between the LSTM and linear layer.

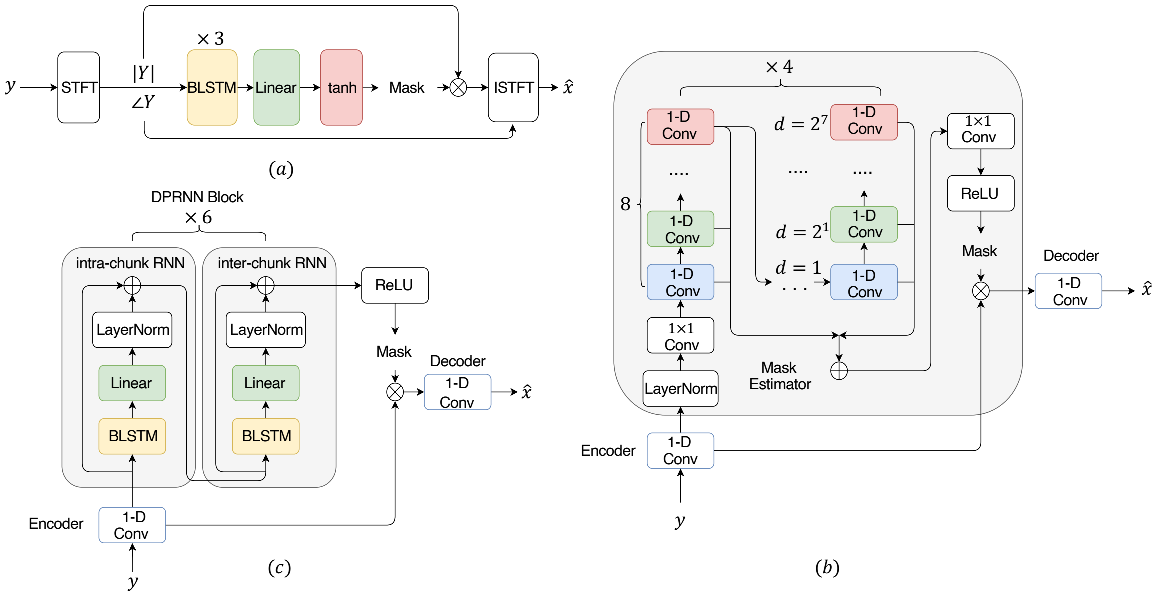

Figure 3. The diagrams of baseline models. (**a**) The diagram of LSTM model. (**b**) The diagram of the Conv-TasNet. (**c**) The diagram of the DPRNN.

All utterances are resampled to 8 kHz. For T-F domain models used in the experiments, STFT with frame size of 64 ms, frame shift of 16 ms, and a 64 ms Hanning window is employed to extract the features. For Conv-TasNet the encoder is a 1D convolution layer with kernel size 40, stride 20, and 256 output channels. For DPRNN, the kernel size and stride for the convolutional encoder are 2 and 1, respectively, and the number of output channels is 64.

The LSTM takes the magnitude of the speech as input and estimates the magnitude mask. The batch size is set to 16. The Conv-TasNet and DPRNN accept raw waves as input and directly output enhanced waves. The batch size for Conv-TasNet and DPRNN are 8 and 4, respectively. The input for the proposed model is the stacked RI components of the noisy spectrum and the output is the estimated RI components. The batch size is set to 16.

All the models are optimized using Adam [31] with a learning rate of 1.0×10^{-3}. The negative Source-to-Noise Ratio (SNR) is used as the loss function. The maximum number of training epoch is 100. For T-F domain models, the training will stop if the loss is not decreasing on the development set for 10 consecutive epochs, and for time domain models, this number is 4. We set the forgetting factors λ_y and λ_n to 0.6 in this study.

5. Results and Discussion

5.1. Effect of the Padding Mode on Multi-Frame Filters

The enhancement performance of the multi-frame filters produced by the Filter Module has an impact on the proposed system. To improve the performance of the whole system, we need to set appropriate hyperparameters for the filters. Since the MFMCWF filter achieves better performance than the MVDR filter in [9], in this study, in addition to the MFMVDR filter, we also tried to let the Filter Module output the multi-frame Wiener filter (MFWF) filter, hoping to explore the performance gap between the two under single-channel conditions.

There are two hyperparameters for the filters: one is the padding mode and the other is the total number of frames. Here, the padding mode means the number of frames on the left of the current frame when the total number of frames is determined.

In order to find the appropriate padding mode for multi-frame filters, we set the total number of frames to 17 and feed the enhanced results from stage 1 and the noisy speeches to the Filter Module. The calculated filter is used to enhance the noisy speeches.

Table 1 shows the effect of the padding mode on MFMVDR, the impact on MFWF has the same trend, for the sake of brevity, the corresponding table is not given here. In the first row, all the frames are padded on the right side of the target frame, which means the system only uses future information, similarly, the last row means the system only uses history information. All the other rows mean the system uses both history and feature information. From Table 1, we can tell that using both history and future information can make the filter have better enhancement performance. Since the results in Table 1 are obtained when the total number of frames is set to 17, without loss of generality, we decided to pad the same number of frames on both sides of the target frames in subsequent experiments.

Table 1. The effect of the padding mode on MFMVDR.

Padding Mode		**PESQ-NB**	**STOI**	**SI-SNR (dB)**	**SDR (dB)**
Left Frames	**Right Frames**				
0	16	2.33	0.90	9.55	10.08
1	15	2.35	0.91	9.86	10.44
2	14	2.36	0.91	9.99	10.59
3	13	2.36	0.91	10.05	10.67
4	12	2.36	0.91	10.09	10.72
5	11	2.36	0.91	10.11	10.75
6	10	2.36	0.91	10.13	10.77
7	9	2.36	0.91	10.13	10.79
8	8	2.36	0.91	10.14	10.80
9	7	2.37	0.91	10.14	10.80
10	6	2.37	0.91	10.14	10.81
11	5	2.38	0.91	10.12	10.81
12	4	2.39	0.91	10.11	10.80
13	3	2.40	0.91	10.08	10.78
14	2	2.41	0.91	10.04	10.73
15	1	2.41	0.91	9.93	10.63
16	0	2.38	0.91	9.62	10.01

5.2. Effect of the Total Number of Frames on Multi-Frame Filters

After determining the padding mode, we further explore the effect of the total number of frames on multi-frame filters, since the total number of frames used to calculate the filters represents the context information. If the total number of frames is too small, the available information will be limited, which will affect the enhancement performance. However, when the total number of frames increases to a certain size, the computing overhead will exceed the performance gain.

Table 2 shows the effect of the total number of frames on MFMVDR and MFWF. The first two rows are the enhancement metrics scores of the noisy speech and the speech enhanced by DNN_1. Both filters are calculated using the enhanced speech from the first stage and are used to process the original noisy speech. From the results, we can observe that both filters perform better as the total number of frames grows. It should be pointed out that due to the limitation of computing resources, we only explored the case of up to 13 frames. When the total number of frames is further increased, whether the enhancement performance will be improved remains to be verified. Another observation is that MFWF performs much better than MFMVDR for the same total number of frames. The table also shows that the enhancement results of DNN 1 are significantly better than the two filters, which is reasonable because the input used to calculate the required parameters for both filters is the enhanced results of DNN1 rather than clean speech, which leads to an accumulation of errors.

Table 2. The effect of the number of frames on different filters.

Filter	# Frames	PESQ-NB	STOI	SI-SNR (dB)	SDR (dB)
noisy	-	1.63	0.77	−2.76	−2.67
DNN$_1$	-	2.79	0.94	13.24	13.69
MFMVDR	$2 \times 2 + 1$	2.10	0.86	3.86	3.94
	$2 \times 4 + 1$	2.34	0.89	7.19	7.37
	$2 \times 6 + 1$	2.39	0.90	9.20	9.60
MFWF	$2 \times 2 + 1$	2.35	0.90	10.62	10.98
	$2 \times 4 + 1$	2.47	0.91	11.15	11.63
	$2 \times 6 + 1$	2.46	0.91	11.35	12.04

5.3. Effect of Filter Type on System Performance

Although the results in Table 2 show that MFWF outperforms MFMVDR when used alone, the effect of both on the overall system performance needs to be further verified. For this purpose, we trained two models on WHAM!, both of which have the same configuration except for the different types of filters used. Both filters set the total number of frames to 13, with 6 frames on both sides of the target frame.

The performance of the two models on the WHAM! test set is shown in Table 3. The WER metric in the table is obtained by feeding the speech enhanced by the model to an ASR system. The ASR system used in this study is a joint Connectionist Temporal Classification-Attention (CTC-Attention) model trained on wsj0 corpus resampled to 8 kHz. In the decoding stage, a word-level language model is used to improve the decoding results.

It is observed that the system using MFMVDR as the filtering module performs slightly better than the one using MFWF, which is different from the conclusion in [9]. The scores on the enhancement metrics are the same for both, but the system using MFMVDR scores better on the WER metric than the one using MFWF. We suppose this may be related to the non-distortion property of MFMVDR; the model may learn this property from MFMVDR in the second training stage so that the enhancement results of the system contain less distortion. As the system using MFMVDR performs better, all models in subsequent experiments will use the MFMVDR filter.

Table 3. The performance of the proposed system with different filtering modules.

Filtering Module	PESQ-NB	STOI	SI-SNR (dB)	SDR (dB)	WER (%)
MFMVDR	2.84	0.94	13.50	13.92	19.60
MFWF	2.83	0.94	13.49	13.92	20.00

5.4. Effect of the Framework on System Performance

The framework shown in Figure 1 can integrate MFMVDR information into DNN$_2$, but incorporating two neural networks in the framework results in a relatively large number of parameters for the model. A possible modification could be to use a single neural network instead, reducing the total number of parameters in the entire system by 50%; the framework is shown in Figure 4. It is worth noting that we have depicted two neural networks in Figure 4 to better illustrate the training process; however, both of these networks are, in fact, identical.

The training of this framework still adopts a two-stage strategy. In the first stage, the noisy spectrum Y are concatenated with two zero matrices of the same shape and fed to the DNN to estimate the clean spectrum. In the second stage, two iterations are required to produce the final output. In the first iteration, similar to the first stage, Y, concatenated with two all-zero matrices, is fed into the DNN, and the initial estimate of the clean spectrum, $\hat{X}^{(1)}$, is obtained. $\hat{X}^{(1)}$ and Y are used to calculate the filtered spectrum $\hat{X}^{(F)}$. In the second iteration, the noisy spectrum Y, the result of the previous iteration $\hat{X}^{(1)}$, and the filtered spectrum $\hat{X}^{(F)}$ are concatenated and fed into the neural network to obtain the final output $\hat{X}^{(2)}$.

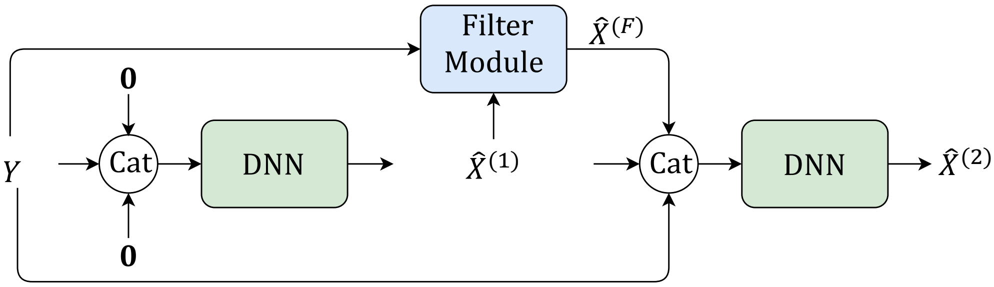

Figure 4. The framework of the system with a single DNN.

Table 4 displays the performance of systems utilizing the two different frameworks on the WHAM! test set. The results indicate that the framework using two DNNs outperforms the one using only one DNN. We hypothesize that there are two possible reasons for this observation. Firstly, it may be challenging for a single neural network to learn both noise reduction and the utilization of MFMVDR results to reduce distortion in the final output. Secondly, it is more probable that when using only a single neural network, the network parameters keep changing, which cannot ensure the accuracy of the input provided to the filter module. In contrast, a dual neural network-based framework can ensure the accuracy of the input provided to the filter module, resulting in superior filtered results and ultimately improving the overall performance.

Given the results shown in Tables 3 and 4, the system proposed in this study is based on the framework utilizing two neural networks and employs MFMVDR as the filter module.

Table 4. Performance of systems using different frameworks.

Framework	#Para (M)	PESQ-NB	STOI	SI-SNR (dB)	SDR (dB)	WER (%)
Two DNNs	15.44	2.84	0.94	13.50	13.92	19.60
Single DNN	7.72	2.75	0.94	12.69	13.11	23.90

5.5. Comparison with Baselines on WHAM!

Table 5 shows the performance of the proposed model and baseline models on the test set of WHAM!. From the table, we can see that compared to noisy speech, speech after enhancement achieves significant improvement in WER, which to some extent illustrates the help of speech enhancement for back-end ASR tasks. The proposed model performs significantly better than the LSTM model in all the metrics, even though both of them are time-frequency domain models and the number of parameters of them is close to each other.

The proposed model has a larger number of parameters than Conv-TasNet, which is reasonable as the latter is a time domain model and usually has a smaller parameter size. Though Conv-TasNet contains half as many parameters as the proposed model, there is still a large gap between its performance and the proposed one, especially on the WER score.

Thanks to its dual-path structure, DPRNN can deliver high-quality enhancement results with a limited number of parameters and performs well in terms of WER metrics. Despite having a larger number of parameters than DPRNN, the proposed model requires less memory and time to train a single epoch during the training phase. Moreover, it outperforms DPRNN in all metrics.

Table 5. Comparison with other methods on WHAM!.

Model	#Para (M)	PESQ-NB	STOI	SI-SNR (dB)	SDR (dB)	WER (%)
noisy	-	1.63	0.77	−2.76	−2.67	73.20
LSTM [17]	16.41	2.50	0.89	9.77	10.43	30.60
Conv-TasNet [20]	8.66	2.27	0.92	11.44	11.94	40.10
DPRNN [21]	2.59	2.68	0.94	13.09	13.65	20.10
proposed	15.44	2.84	0.94	13.50	13.92	19.60

5.6. Ablation Study

We also perform an ablation test to evaluate the effectiveness of the two-stage training strategy and the incorporation of MFMVDR results. The results are shown in Table 6. The first row corresponds to a single TCN-DenseUNet model. The second row uses the MFMVDR as a post-processing module for the TCN-DenseUNet. The model in the third row only takes the noisy input and the enhanced speech from the first stage as inputs in the second training stage. The fourth row is the results of the proposed model.

By comparing the first and fourth rows, we can observe that our chosen base network, TCN-DenseUNet, achieves strong performance, but there is still room for noticeable improvement by training it in two stages and fusing the MFMVDR information. When we compare the first and second rows, it can be seen that post-processing the output of the neural network with the MFMVDR filter alone does not work and even leads to a significant drop in all metrics of the processed speech. However, by adopting the two-stage training approach as shown in the first and third rows, we can achieve improvements in both enhancement and WER metrics. Furthermore, comparing the third and fourth rows, it becomes apparent that combining the information from the MFMVDR in the second training phase can further improve the performance of the system. This finding suggests that the results of the MFMVDR can also contribute to the overall system's performance improvement.

Table 6. Ablation study of two-stage training strategy and MFMVDR.

2 Stage Training	MFMVDR	PESQ-NB	STOI	SI-SNR (dB)	SDR (dB)	WER (%)
✗	✗	2.79	0.94	13.24	13.69	21.80
✗	✓	2.39	0.90	9.20	9.60	50.80
✓	✗	2.79	0.94	13.47	13.92	19.90
✓	✓	2.84	0.94	13.50	13.92	19.60

5.7. Generalizability of the Proposed System

To evaluate the generalization of the proposed system, we used the system trained on the WHAM! dataset to directly enhance the noisy speech from the CHiME-4 dataset; the results are shown in Table 7. The proposed system outperforms the other methods significantly on all the metrics for both simulated data and real-world data, indicating its superior generalizability.

Table 7. Comparison with other methods on CHiME-4.

Model	CHiME-4 SIMU					CHiME-4 REAL
	PESQ-NB	STOI	SI-SNR (dB)	SDR (dB)	WER (%)	WER (%)
noisy	1.74	0.81	5.07	5.18	76.60	81.80
LSTM [17]	2.45	0.89	12.21	12.84	45.80	61.90
Conv-TasNet [20]	2.24	0.89	10.86	12.56	47.50	62.80
DPRNN [21]	2.47	0.90	10.17	13.19	38.30	45.40
proposed	2.69	0.93	14.23	14.87	30.50	41.40

6. Conclusions

In this study, a two-stage model for single-channel speech enhancement is proposed. There are two DNNs in the proposed model—in the first stage, one of the two DNNs is trained first. In the second stage, the other DNN uses the enhanced speech from the trained DNN as extra input features. To further improve the enhancement performance and reduce the distortion introduced by neural networks, the result of a single-channel filter is also used in the second stage to guide the training of the model. Two different single-channel filters are investigated in this study, namely, MFMVDR and MFWF. We investigate the influence of the number of frames used to calculate the filter and the padding mode on the performance of the two filters. We also compare the impact of MFMVDR and MFWF on the final model performance and find that MFMVDR delivers more improvement.

Our main contribution is proposing a two-stage training approach in the single-channel scenario, which utilizes the information from MFMVDR filters to assist in neural network training. As a result, our method achieves improved speech enhancement performance and higher speech recognition accuracy. Experiments on two datasets containing both synthetic and real-world noisy speech show that the proposed model has better enhancement performance and generalization ability. On the WHAM! test set, our model exhibited a relative improvement of 3% in SI-SNR and 2% in WER compared to the best-performing baseline model, DPRNN. On the synthetic test set of CHiME-4, our model demonstrated substantial relative improvements of 40% and 20% in SI-SNR and WER, respectively. Additionally, our model exhibited a noteworthy relative improvement of 9% in WER on the more challenging real test set of CHiME-4. The ablation study demonstrates the effectiveness of the two-stage training strategy and the incorporation of MFMVDR results in training.

Possible directions for future improvements include:

- Smaller model parameters. In this study, we employ TCN-DenseUNet, a time-frequency domain model, for both DNN_1 and DNN_2. Our future research will explore the use of time-domain models, such as DPRNN, as the structure for DNN_1 and DNN_2 to reduce the number of model parameters.
- Improved training strategies. For example, using the parameters of DNN_1 to initialize DNN_2 in the second stage, followed by fine-tuning DNN_2, or jointly training DNN_1 and DNN_2 with DNN_1 using a smaller learning rate to speed up model convergence and further improve performance.
- More refined information fusion methods. In this study, the input of DNN_2 is a simple concatenation of $\hat{X}^{(1)}$, $\hat{X}^{(F)}$, and Y. In future research, more sophisticated methods can be explored to fuse the information of the three, such as attention mechanisms.
- Diversified model selection. We could make DNN1 or DNN2 a time-frequency domain model, and the other a time-domain model. It is hoped that these two models can complement each other to achieve better enhancement performance.

Author Contributions: Conceptualization, W.Z. and S.L.; methodology, W.Z. and S.L.; software, W.Z. and S.L.; validation, S.L., W.Z. and Y.Q.; investigation, S.L.; resources, Y.Q.; data curation, S.L.; writing—original draft preparation, S.L.; writing—review and editing, W.Z. and Y.Q.; visualization, S.L.; supervision, Y.Q.; project administration, Y.Q.; funding acquisition, Y.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by China NSFC projects under Grants 62122050 and 62071288, and in part by Shanghai Municipal Science and Technology Major Project under Grant 2021SHZDZX0102 and in part by Jiangsu Technology Project (No.BE2022059-4).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Boll, S. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Trans. Acoust. Speech Signal Process.* **1979**, *27*, 113–120. [CrossRef]
2. Wiener, N.; Wiener, N.; Mathematician, C.; Wiener, N.; Wiener, N.; Mathématicien, C. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series: With Engineering Applications*; MIT Press: Cambridge, MA, USA, 1949; Volume 113.
3. Lee, D.D.; Seung, H.S. Learning the parts of objects by non-negative matrix factorization. *Nature* **1999**, *401*, 788–791. [CrossRef] [PubMed]
4. Bedoui, R.A.; Mnasri, Z.; Benzarti, F. On the Use of Spectrogram Inversion for Speech Enhancement. In Proceedings of the 2021 18th International Multi-Conference on Systems, Signals & Devices (SSD), Monastir, Tunisia, 22–25 March 2021; pp. 852–857. [CrossRef]
5. Chang, S.; Kwon, Y.; Yang, S.i.; Kim, I.j. Speech enhancement for non-stationary noise environment by adaptive wavelet packet. In Proceedings of the 2002 IEEE International Conference on Acoustics, Speech, and Signal Processing, Orlando, FL, USA, 13–17 May 2002; Volume 1, pp. I-561–I-564. [CrossRef]
6. Iwamoto, K.; Ochiai, T.; Delcroix, M.; Ikeshita, R.; Sato, H.; Araki, S.; Katagiri, S. How bad are artifacts?: Analyzing the impact of speech enhancement errors on ASR. *arXiv* **2022**, arXiv:2201.06685. [CrossRef]
7. Ochiai, T.; Delcroix, M.; Ikeshita, R.; Kinoshita, K.; Nakatani, T.; Araki, S. Beam-TasNet: Time-domain Audio Separation Network Meets Frequency-domain Beamformer. In Proceedings of the ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Virtual, 4–9 May 2020; pp. 6384–6388. [CrossRef]
8. Wang, Z.Q.; Wichern, G.; Roux, J.L. Leveraging low-distortion target estimates for improved speech enhancement. *arXiv* **2021**, arXiv:2110.00570. [CrossRef]
9. Lu, Y.J.; Cornell, S.; Chang, X.; Zhang, W.; Li, C.; Ni, Z.; Wang, Z.Q.; Watanabe, S. Towards Low-Distortion Multi-Channel Speech Enhancement: The ESPNET-SE Submission to the L3DAS22 Challenge. In Proceedings of the ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 22–27 May 2022; pp. 9201–9205. [CrossRef]
10. Van Veen, B.; Buckley, K. Beamforming: A versatile approach to spatial filtering. *IEEE ASSP Mag.* **1988**, *5*, 4–24. [CrossRef] [PubMed]
11. Van Trees, H.L. *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*; John Wiley & Sons: New York, NY, USA, 2002; pp. 480–510, ISBN 9780471221104. [CrossRef]
12. Benesty, J.; Huang, Y. A single-channel noise reduction MVDR filter. In Proceedings of the ICASSP 2011–2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 22–27 May 2011; pp. 273–276. [CrossRef]
13. Huang, Y.A.; Benesty, J. A Multi-Frame Approach to the Frequency-Domain Single-Channel Noise Reduction Problem. *IEEE Trans. Audio Speech Lang. Process.* **2012**, *20*, 1256–1269. [CrossRef]
14. Srinivasan, S.; Roman, N.; Wang, D. Binary and ratio time-frequency masks for robust speech recognition. *Speech Commun.* **2006**, *48*, 1486–1501. [CrossRef]
15. Erdogan, H.; Hershey, J.R.; Watanabe, S.; Le Roux, J. Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks. In Proceedings of the ICASSP 2015–2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Queensland, Australia, 19–24 April 2015; pp. 708–712. [CrossRef]
16. Williamson, D.S.; Wang, Y.; Wang, D. Complex ratio masking for monaural speech separation. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2015**, *24*, 483–492. [CrossRef] [PubMed]
17. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
18. Tan, K.; Wang, D. A Convolutional Recurrent Neural Network for Real-Time Speech Enhancement. In Proceedings of the Interspeech 2018, Hyderabad, India, 2–6 September 2018; pp. 3229–3233. [CrossRef]
19. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Singapore, 18–22 September 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241. [CrossRef]
20. Luo, Y.; Mesgarani, N. Conv-TasNet: Surpassing Ideal Time–Frequency Magnitude Masking for Speech Separation. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2019**, *27*, 1256–1266. [CrossRef] [PubMed]
21. Luo, Y.; Chen, Z.; Yoshioka, T. Dual-Path RNN: Efficient Long Sequence Modeling for Time-Domain Single-Channel Speech Separation. In Proceedings of the ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 46–50. [CrossRef]
22. Wang, Z.Q.; Wichern, G.; Le Roux, J. Convolutive prediction for monaural speech dereverberation and noisy-reverberant speaker separation. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2021**, *29*, 3476–3490. [CrossRef]
23. Carlson, B.D. Covariance matrix estimation errors and diagonal loading in adaptive arrays. *IEEE Trans. Aerosp. Electron. Syst.* **1988**, *24*, 397–401. [CrossRef]
24. Wichern, G.; Antognini, J.; Flynn, M.; Zhu, L.R.; McQuinn, E.; Crow, D.; Manilow, E.; Roux, J.L. WHAM!: Extending Speech Separation to Noisy Environments. *arXiv* **2019**, arXiv:1907.01160. [CrossRef]

25. Hershey, J.R.; Chen, Z.; Le Roux, J.; Watanabe, S. Deep clustering: Discriminative embeddings for segmentation and separation. In Proceedings of the ICASSP 2016–2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 31–35. [CrossRef]
26. Vincent, E.; Watanabe, S.; Nugraha, A.A.; Barker, J.; Marxer, R. An analysis of environment, microphone and data simulation mismatches in robust speech recognition. *Comput. Speech Lang.* **2017**, *46*, 535–557. [CrossRef]
27. Rix, A.; Beerends, J.; Hollier, M.; Hekstra, A. Perceptual evaluation of speech quality (PESQ)—A new method for speech quality assessment of telephone networks and codecs. In Proceedings of the ICASSP 2001–2001 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Salt Lake City, UT, USA, 7–11 May 2001; Volume 2, pp. 749–752. [CrossRef]
28. Taal, C.H.; Hendriks, R.C.; Heusdens, R.; Jensen, J. A short-time objective intelligibility measure for time-frequency weighted noisy speech. In Proceedings of the ICASSP 2010–2010 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Dallas, TX, USA, 14–19 March 2010; pp. 4214–4217. [CrossRef]
29. Roux, J.L.; Wisdom, S.; Erdogan, H.; Hershey, J.R. SDR—Half-baked or Well Done? In Proceedings of the ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 626–630. [CrossRef]
30. Vincent, E.; Gribonval, R.; Févotte, C. Performance measurement in blind audio source separation. *IEEE Trans. Audio Speech Lang. Process.* **2006**, *14*, 1462–1469. [CrossRef]
31. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2015**, arXiv:1412.6980. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.