


Article

Hybrid Dilated and Recursive Recurrent Convolution Network for Time-Domain Speech Enhancement

Zhendong Song¹, Yupeng Ma^{2,*}, Fang Tan¹  and Xiaoyi Feng¹

¹ School of Electronics and Information, Northwestern Polytechnical University, Xi'an 710129, China; secszd@163.com (Z.S.); bjtanfang@163.com (F.T.); fengxiao@nwpu.edu.cn (X.F.)

² College of Computer and Cyber Security, Hebei Normal University, Shijiazhuang 050024, China

* Correspondence: mayupenga@mail.nwpu.edu.cn

Abstract: In this paper, we propose a fully convolutional neural network based on recursive recurrent convolution for monaural speech enhancement in the time domain. The proposed network is an encoder-decoder structure using a series of hybrid dilated modules (HDM). The encoder creates low-dimensional features of a noisy input frame. In the HDM, the dilated convolution is used to expand the receptive field of the network model. In contrast, the standard convolution is used to make up for the under-utilized local information of the dilated convolution. The decoder is used to reconstruct enhanced frames. The recursive recurrent convolutional network uses GRU to solve the problem of multiple training parameters and complex structures. State-of-the-art results are achieved on two commonly used speech datasets.

Keywords: speech enhancement; time domain; hybrid dilated convolution; recurrent convolution



Citation: Song, Z.; Ma, Y.; Tan, F.; Feng, X. Hybrid Dilated and Recursive Recurrent Convolution Network for Time-Domain Speech Enhancement. *Appl. Sci.* **2022**, *12*, 3461. <https://doi.org/10.3390/app12073461>

Academic Editor: Lijiang Chen

Received: 21 February 2022

Accepted: 27 March 2022

Published: 29 March 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Speech enhancement refers to the technology of removing or attenuating noise from noisy speech signal and extracting useful speech signal. Speech enhancement technology is widely used in automatic speech recognition, speech communication system, and hearing aids. Traditional monaural speech enhancement methods include spectral subtraction [1], Wiener filtering [2], and subspace algorithm [3].

In the past few years, supervised methods based on deep learning have become the mainstream for speech enhancement. In these supervised methods, time-frequency (T-F) features of noisy speech are extracted first, and the T-F features of clean speech are extracted to represent the target. Training targets can be divided into two types; one is the masking-based, such as the ideal binary mask (IBM) [4] and the ideal ratio mask (IRM) [5]. The other is the spectral mapping-based, such as the log power spectrum feature used in [6]. Methods [5,7,8] based on T-F domain have certain limitations. Firstly, these methods require preprocessing, which increases the complexity. Secondly, these methods usually ignore the phase information of clean speech and use the noisy signal phase to reconstruct the time domain signal. Some previous studies have proved that phase is very important to improve speech quality, especially in low SNR [9].

For the above reasons, researchers have proposed a variety of speech enhancement networks based on time-domain [10–12]. Fu et al. [13] proposed a fully convolutional network and proved that the network is more suitable for speech enhancement in the time-domain than a full connection network. Then, a text speech model named WaveNet [14] directly synthesizes the original waveform. Rethage et al. [15] proposed an improved WaveNet model for speech enhancement based on WaveNet, which uses residual connection and one-dimensional dilated convolution. After that, Pandey et al. [16] combined a time convolution module and codec for speech enhancement in the time-domain, in which the time convolution module also uses one-dimensional dilated convolution.

One-dimensional dilated convolution can improve the receptive field of the network model. However, when one-dimensional dilated convolution is used for time-domain speech enhancement tasks, there is a problem as local information cannot be fully utilized. The reason is that when the dilation rate is greater than one during the convolution, holes in dilated convolutions lead to local information loss. Therefore, a hybrid dilated convolution module is proposed to combine dilated convolution with conventional convolution.

The end-to-end speech enhancement algorithm directly processes the original waveform of the speech, avoiding the low calculation efficiency and “phase inconsistency” problems based on the time-frequency domain-speech enhancement algorithm and also achieves a better enhancement effect. However, whether based on end-to-end or non-end-to-end speech enhancement algorithms, these models have a large number of trainable parameters. Recently, recursive learning mechanisms have been applied to a variety of tasks, such as single-image de-rain [17] and image super-resolution [18]. The principle of recursive learning is to train the network recursively using the same network parameters, similar to a mathematical iterative process in which the entire process of mapping the network model is handled in several stages. Thus, through recursive learning, the network parameters can be reused at each stage, and we can explore the network at a deeper level without using additional parameters. Inspired by recursive learning, this paper proposes a speech enhancement algorithm based on combining a hybrid dilated convolution module and recursive learning. The contributions of this article can be summarized as:

1. Proposed hybrid dilated convolution module (HDM), which consists of dilated convolution and conventional convolution;
2. Proposed recursive recurrent speech enhancement network (RRSENet), which uses a GRU module to solve the problem of multiple training parameters and complex structures;
3. Extensive experiments are performed on dataset synthesized by TIMIT corpus and NOISEX92, and the proposed model achieves excellent results.

The remainder of this paper is structured as follows. Section 2 describes the related work on speech enhancement, RNN, and dilated convolution. Section 3 formulates the problem and proposes the architecture. Section 4.3 presents the experiment settings, results, and analysis. Some conclusions are drawn in Section 5.

2. Related Work

2.1. Speech Enhancement

Spectral subtraction, Wiener filtering, and subspace algorithm are the three most classic traditional monaural speech enhancement methods. Spectral subtraction methods [1,19–22] firstly obtains the noise spectrum to be processed by estimating and updating the noise spectrum operation in the non-speech segment. After that, the enhanced speech spectrum will be estimated through the subtraction operation. Finally, the speech spectrum is converted into a speech waveform. Although the spectral subtraction method is relatively simple, there will be problems such as voice distortion and music noise, and this type of method is suitable for the case of stable noise. The effect of suppressing non-stationary noise is relatively poor.

The Wiener filtering algorithms [2,23–26] originated during the Second World War. It was proposed by the mathematician Norbert Wiener to solve the military air shooting control problem. It is mainly used to extract the required clean signal from the noisy observation signal. The Wiener filtering algorithm has a history of nearly 80 years, and its ideas have undergone many variations after decades of development. The essence of the Wiener filtering algorithm is to extract the signal from the noise and use the minimum mean square value of the error between the estimated result and the true value of the signal as the best criterion. Therefore, the Wiener filter is the optimal filter in the statistical sense or the optimal linear estimator of the waveform. However, the Wiener filter method has a general ability to handle non-stationary noise, which will cause voice distortion.

The principle of the subspace algorithms is to decompose the vector space of the observed signal into a signal subspace, a noise subspace, and estimate the clean speech by eliminating the noise subspace and retaining the signal subspace. The process of subspace decomposition is to perform KLT transformation on the noisy speech signal, then set the KLT coefficient of the noise to 0, and finally obtain the enhanced speech through the inverse KLT transformation. The subspace method generally does not cause the problem of voice distortion and can maintain the quality of the enhanced voice. Its disadvantage is that it removes less noise and the enhancement effect is also not very good.

Compared with traditional speech enhancement algorithms, the algorithms [27–31] based on deep learning have obtained a relatively obvious improvement in performance and effect. With the development of deep neural network (DNN), which has promoted the rapid development of related research in the field of speech enhancement, researchers have proposed many DNN models to solve the problem of speech enhancement [28]. Computational inefficiencies and phase inconsistencies exist in non-end-to-end algorithms, so the researchers performed speech enhancement directly in the time domain. Most of the end-to-end algorithms use one-dimensional dilated convolution to improve the network's extraction of contextual information from the original speech waveform, and to a certain extent, the enhancement effect of the network model is improved.

2.2. Recurrent Neural Network

Recurrent neural networks (RNNs) [32] are excellent for sequence information processing. The core is to use the current information of the sequence and the current output information to infer the information of the next output, which can improve the prediction of the output. In Deep neural networks (DNNs) [29–31] the assumption of independence between input and output features is a very poor assumption for most tasks. In the RNN, the neural unit operates on the elements of the input sequence in the same way, that is, the current input and the previous output of the neuron are combined as the current output. RNN can reduce the complexity of the network model and facilitate training by using the states of the current neuron and the states of the previous neurons. Given the characteristics of RNN, it is an indispensable tool when solving natural language processing tasks, such as speech recognition, speech modeling, and machine translation. Short-term memory affects the performance of RNN when dealing with longer text or speech, and RNN loses information at the beginning of each sequence. Long short-term memory (LSTM) [33] and GRU [34] with a simpler structure were created to solve the short-term memory problem of RNN, and these two network structures are more commonly used when dealing with sequence tasks. Both of these network structures have a gate mechanism, also known as a memory mechanism, which can regulate the flow of information and control whether information is ignored in the process of neural unit transmission [35].

2.3. Dilated Convolution

In the traditional convolutional neural networks, the context information is usually augmented by extending the receptive field. One way to achieve this goal is to increase the depth of the network model and use a deeper network. Another method is to enlarge the size of convolution kernel. They will both raise the computational burden and the training time of the network model. Therefore, Yu and Koltun first proposed a multi-scale context aggregation dilated convolution model [36]. The original proposal was to solve the problem of image semantic segmentation because the context information between images is very helpful for object segmentation. Ye et al. [37] and S Pirhosseinloo et al. introduced dilated convolution into their algorithms.

3. Model Description

3.1. Problem Formulation

Given a single-microphone noisy signal $y(t)$, the target of single-channel speech enhancement is to estimate the target speech $x(t)$. This paper focuses on the additive

relationship between the target speech and noise. Therefore, the noisy signal $y(t)$ can be defined as

$$y(t) = x(t) + n(t) \tag{1}$$

where $y(t)$ is the time-domain noisy signal at time t , $x(t)$ is the time-domain clean signal at time t , and $n(t)$ is the time-domain noise signal at time t .

3.2. Hybrid Dilated Convolution Module (HDM)

In traditional convolutional neural networks, the context information is usually augmented by extending the receptive field. One way to achieve this goal is to increase the depth of the network model and use a deeper network. Another method is to enlarge the size of the convolution kernel. They will both raise the computational burden and the training time of the network model. Therefore, Yu and Koltun [36] first proposed a multi-scale context aggregation dilated convolution model. The original proposal was to solve the problem of image semantic segmentation, because the context information between images is very helpful for object segmentation.

Figure 1a is a one-dimensional convolutional neural network with three conventional convolution layers. The expansion rate r of each layer is 1, Figure 1b is a one-dimensional convolutional neural network with three dilated convolution layers. The expansion rates r of each layer are 1, 2, and 4, respectively. The top blue unit is regarded as the unit of interest, and the other blue units represent its receptive field in each layer. Compared with the conventional kernel convolution, the dilated convolution expands the receptive field of the convolution kernel without increasing the parameters.

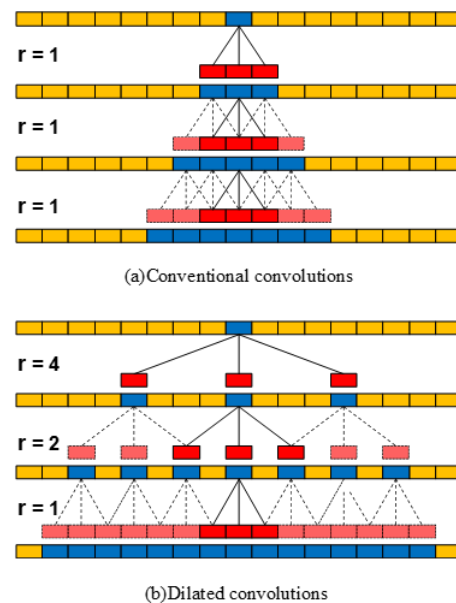


Figure 1. Conventional convolution and dilated convolution.

As shown in Figure 2, the hybrid dilated convolution module(HDM) consists of three parts: input 1×1 convolution, feature fusion convolution, and output 1×1 convolution. The input 1×1 convolution scrolling reduces the number of channels to half, reducing the number of model parameters. In feature fusion convolution, dilated and conventional convolution outputs are directly added one by one according to the elements. In addition, both dilated convolution and conventional convolution keep the number of channels unchanged. The output 1×1 convolution doubles the number of channels, so that the input and output of HDM channel number are consistent. Finally, the output is combined with the input of the module to form a residual connection [38], which makes the network training easier.

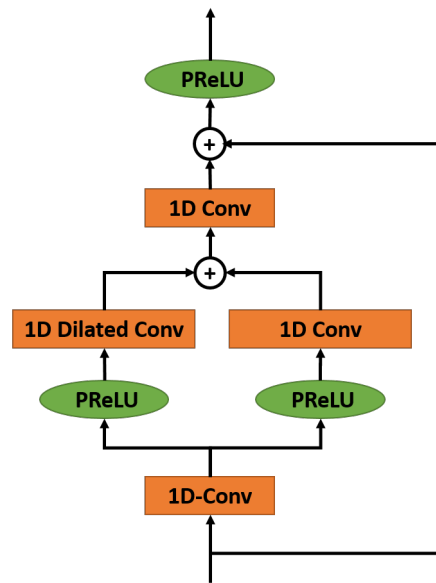


Figure 2. Hybrid dilated and conventional convolution module.

3.3. Recursive Learning Speech Enhancement Network (RLSENet)

In order to improve the performance of speech enhancement, deeper and more complex networks are generally required, which usually leads to more training parameters. Therefore, multiple stages can deal with the speech enhancement problem. A conventional multi-stage solution is the addition of multiple sub-networks, but it will inevitably increase network parameters and cause overfitting. In contrast, sharing network parameters through multiple stages and using inter-stage recursive learning allows exploring the deeper network for speech enhancement without increasing parameters.

The recursive mechanism works by combining the estimated output of the previous stage of the network model with the original noisy signal on the channel as the input of the next network. The output of each stage can be compared to a state between different stages, similar to the recurrent neural network, which can train the network model cyclically. In this way, the feature dependencies of different stages can be fully utilized, and the estimation of the network can be gradually refined. Figure 3 is the proposed recursive learning speech enhancement network. The output of each stage t in RLSENet can be defined as:

$$x^{t-0.5} = f_{en}(x^{t-1}, y) \tag{2}$$

$$x^t = f_{de}(f_{hdm}(f_{en}(x^{t-0.5}))) \tag{3}$$

where, f_{en} , f_{hdm} , and f_{de} represent the fully convolutional encoder, hybrid dilated convolution module, and fully convolutional decoder in RLBlock, respectively. They are stage-invariant, and the parameters of network are reused in different stages. f_{en} combines the estimated x^{t-1} of the current stage and the noisy signal y as input. $x^{t-0.5}$ represents the input of the fully convolutional decoder.

RLBlock consists of three parts: encoder, decoder, and 6 HDMs. The input of RLBlock is the output of the previous stage and the original noisy signal. The full convolution encoder is used to extract the low-dimensional features of the input speech frames. The HDM combines dilated and conventional convolution to fully use contextual information without losing local information. The full convolution decoder is used to reconstruct the enhanced speech frames. The structure of RLBlock is shown in Figure 4b. The encoder consists of four one-dimensional convolution layers. The input size is 1×2048 , in which one represents the number of channels and 2048 represents the frame length of speech. The convolution of each layer of the encoder will halve the frame length of the speech; that is, the step size is 2, and the number of output channels of each layer is 16, 32, 64, and 128. So, the output of the last layer is 128×128 , the activation function of each layer in the encoder is

PReLU [39], and the filter size is 11. The expansion rate of the 6 HDMs grows exponentially and is set to 1, 2, 4, 8, 16, and 32, respectively. The decoder is a mirror-image of the encoder, with four layers of one-dimensional deconvolution [40], in which the hopping connection is used to connect each coding layer with its corresponding decoding layer, making up for the loss of features in the coding process. In the decoder, the activation function used in the first three layers is PReLU, and the activation function used in the last layer is Tanh. The filter size of each layer is 11.

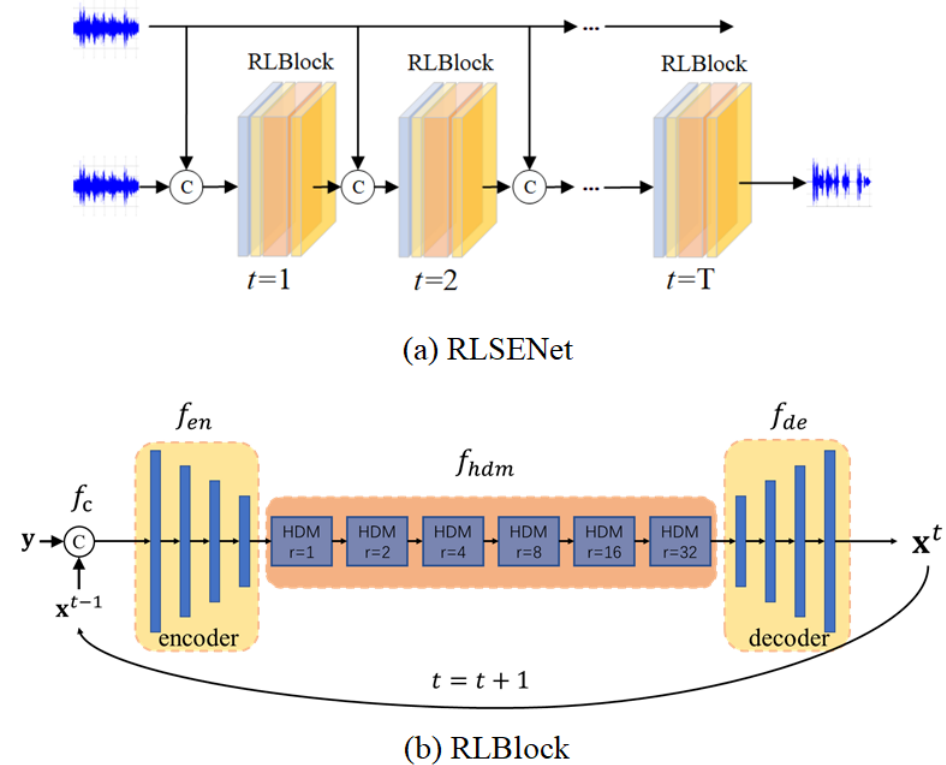


Figure 3. (a) RLSNet: recursive learning speech enhancement network, (b) RLBlock: recursive learning block.

3.4. Recursive Recurrent Speech Enhancement Network (RRSENet)

The learning process of noisy speech to clean speech can be considered as a sequence learning, where each stage represents the intermediate output of a stage. Therefore, the network can be trained in the same way as a RNN. A GRU module is introduced in RLBlock to form recursive recurrent Block (RRBlock), as shown in Figure 4b. Through the RRBlock, the feature dependencies of different stages can be propagated to facilitate the noise removal.

Compared with RLBlock, RRBlock adds a GRU module before the encoder. The GRU module has a convolution layer, the frame length of input speech is 2048 points, the kernel size of this layer is 11, and the stride is 2. The convolution layer increases the number of channels from 2 to 16, and reduces the input speech frame to 1024 points. The input and output of the first convolution layer in the encoder remains the same in the number of channels and speech frame length, the output is 16×1024 , and the others are consistent with RLBlock. The network trained recursively by using the RRBlock module is named the recursive recurrent speech enhancement network (RRSENet), as shown in Figure 4a.

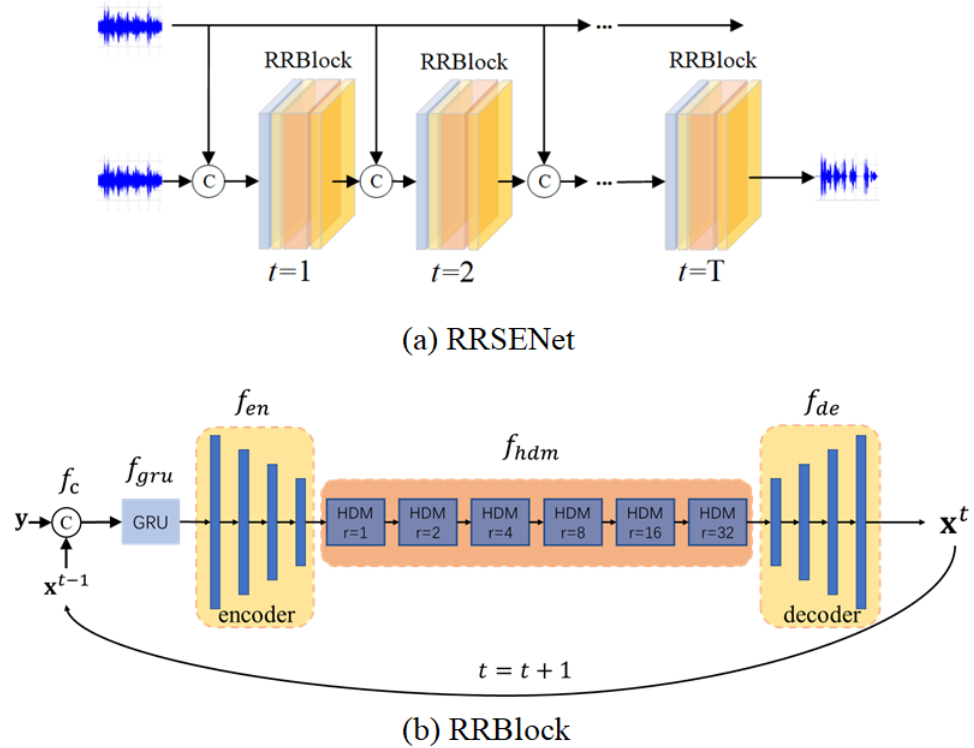


Figure 4. (a) RRSENet: recursive recurrent speech enhancement network, (b) RRBlock: recursive recurrent block.

The output of each stage t in RRSENet can be defined as:

$$x^{t-0.5} = f_c(x^{t-1}, y) \tag{4}$$

$$s^t = f_{recurrent}(s^{t-1}, x^{t-0.5}) \tag{5}$$

$$x^t = f_{de}(f_{hdm}(f_{en}(s^t))) \tag{6}$$

where, f_c concatenates the output of the previous stage to the original noise signal on the channel, $f_{recurrent}$ is the loop layer, which takes the intermediate state s^{t-1} and $x^{t-0.5}$ as input to the stage $t - 1$, and the loop layer is able to regulate the information flow. Then the intermediate state s^t is input to the encoder f_{en} for feature extraction of speech information, and then the enhanced speech frames are output after f_{hdm} and f_{de} .

4. Experiments

4.1. Datasets

In the experiment, the clean corpus used comes from the TIMIT corpus [41], which includes 630 speakers of 8 major dialects of American English with each reading 10 utterances. All sentences in the TIMIT corpus are sampled at 16 KHZ, with 4620 utterances in the training set and 1680 utterances in the test set, resulting in a total of 6300 utterances. Then, 1000, 200, and 100 clean utterances are randomly selected for training, validation, and testing, respectively. Training and validation dataset are mixed under different SNR levels ranging from -5 dB to 10 dB with the interval 1 dB while the testing datasets are mixed under -5 dB and -2 dB conditions.

For training and validation, we used two noisy datasets. One dataset is a noise library recorded in the laboratory of Prof. Wang at Ohio State University, which has 100 sounds with different durations and a sampling rate of 16 kHz. A total of 100 kinds of noise, which includes machine, water, wind, crying noise, etc, were used. The other is NOISEX92 [42], with 15 noises, a duration of 235 s, and a sampling frequency of 19.98 kHz. A total of 15 kinds of noise, which includes truck, machine gun, factory, etc, were used. Another five

types of noises from NOISEX92, including babble, f16, factory2, m109, and white, were chosen to test the network generalization capacity.

The dataset is constructed using the following steps. First, the noise is downsampled to 16 kHz and stitched into a long noise. Second, a clean speech is randomly selected, and a noise of the same length is selected. Finally, during each mixed process, the cutting point is randomly generated, which is subsequently mixed with a clean utterance under one SNR condition. As a result, totally 10,000, 2000, and 400 noisy-clean utterance pairs are created for training, validation, and testing, respectively.

4.2. Experimental Settings

All the speech enhancement network models are written in Python, and the models are configured and trained using the PyTorch.

For model training, the synthesized noisy speech and clean speech are framed, both at a sampling rate of 16 kHz, with each frame having a size of 2048 sample points, i.e., a frame length of 128 ms, and an offset of 512 sample points between adjacent frames, i.e., a frame shift of 32 ms. All the compared modules, the compared speech enhancement network models, and the speech enhancement algorithm proposed in this chapter are trained using mean absolute error (MAE). This loss function is used to calculate the error between the estimated and actual values, and Adam [43] is used to speed up the convergence of the model. We set the hyper-parameter learning rate to 0.0002 at the beginning, and when there are three consecutive increases in the validation loss, the learning rate is halved. The training process is stopped early when there are 10 increases in the validation loss. For the training set and validation set, a 5-fold cross-validation method method is used to conduct the experiments.

4.3. Experimental Results

4.3.1. Compared with Typical Algorithms

In order to verify the effectiveness of the proposed speech enhancement network model RRSENet, we compared three typical speech enhancement algorithms, namely LogMMSE [44], TCNN [16], and AECNN [45]. Among them, LogMMSE is the minimum mean square error logarithmic spectrum amplitude estimation, which is a speech enhancement algorithm based on statistical models. TCNN is a speech enhancement model based on a temporal convolution neural network. The overall framework of the model consists of an encoder and a decoder, and 18 temporal convolution modules are embedded between the encoder and decoder. Compared with TCNN, RLSENet replaces the temporal convolution module used in TCNN with a hybrid dilated convolution module. While AECNN is a typical 1-D convolution encoder-decoder framework, it still needs to train a large number of parameters. The frame length of model input and output is 2048, and the number of channels in successive layers is 1, 64, 64, 64, 128, 128, 128, 256, 256, 256, 512, 512, 256, 256, 128, 128, 128, and 1, the size of the convolution kernel of each layer is 11, and the activation function uses PReLU. RRSENet adds the GRU module on the basis of RLSENet. The experiment was tested on the matched test set and the unmatched test set.

Table 1 is the comparison result of the three typical algorithms of LogMMSE [44], TCNN [16], and AECN [45] under noise matching at -5 dB, 0 dB, and 5 dB. The evaluation index is the average PESQ [46] and STOI [47] values. The “PESQ” value is the mean of speech quality under different signal-to-noise ratios in case of matched noise. The “STOI” value is the mean of speech intelligibility under different signal-to-noise ratios. The average PESQ value is the average speech quality evaluation value under different signal-to-noise ratios. The average STOI value represents the evaluation value of the average speech intelligibility under different signal-to-noise ratios, and the “Avg.” represents the three types of signals under different evaluation indicators. HDMNet is a network structure that does not use GRU modules; RLSENet is a recursive learning network using the HDM modules; and RRSENet is a network structure that uses HDM modules and GRU modules.

The results in the Table 1 show that the RLSENet and RRSENet using recursive learning network outperform the other three speech enhancement algorithms on PESQ and

STOI evaluation metrics. Among them, “HDMNet” is the speech enhancement network that uses only the HDM module. The traditional method LogMMSE enhancement is the least effective, indicating that it is difficult to handle non-smooth noise. In contrast to the temporal convolution module used in TCNN, which only considers the historical information and uses a one-dimensional dilated convolution with its own defects, the hybrid dilated module proposed in this paper is better than the traditional convolution block, making full use of the information of the neighboring points of the speech waveform without losing local information and improving the enhancement performance of the model. Compared with RLSNet, RRSENet adds a GRU module, which makes the results of RRSENet better than RLSNet. In general, comparing with other network models, the RRSENet network model proposed in this paper has the best performance. For example, in a low SNR-5dB noise environment, the enhanced speech with RRSENet network model achieves the best enhancement performance compared to the unprocessed noisy speech with PESQ and STOI by 0.693 and 16.92%, respectively. The AECNN and the TCNN using the temporal convolution module are slightly inferior, which proves that the RRSENet network model is more suitable for end-to-end speech enhancement tasks.

Table 1. Experimental results of different network models under seen noise conditions for PESQ and STOI. **BOLD** indicates the best result for each case.

Metrics	PESQ				STOI				
	SNR	−5 dB	0 dB	5 dB	Avg.	−5 dB	0 dB	5 dB	Avg.
Noisy		1.537	1.843	2.083	1.821	66.40	77.56	84.92	76.30
LogMMSE		1.579	1.906	2.176	1.887	66.77	78.25	85.52	76.84
Wiener		1.710	2.180	2.610	2.166	65.15	77.84	85.90	76.29
TCNN		1.969	2.510	2.830	2.436	80.87	89.11	92.71	87.56
AECNN		2.013	2.557	2.945	2.505	81.33	89.45	92.99	87.92
HDMNet		2.122	2.654	3.013	2.596	82.03	89.52	93.09	88.22
RLSNet		2.151	2.692	3.039	2.627	83.06	90.18	93.40	88.80
RRSENet		2.220	2.750	3.051	2.674	83.17	90.19	93.30	88.89

From the results in Table 2, it can be seen that at a low signal-to-noise ratio of −5 dB, the enhanced speech of RRSENet has increased by 4.5% and 4.98% compared with the enhanced speech of TCNN and AECNN, respectively. Because under the background of low signal-to-noise ratio, people pay more attention to the intelligibility of speech, that is, the STOI evaluation index. This shows that RRSENet can improve the intelligibility of speech under low signal-to-noise ratio. In addition, the “Avg.” of RRSENet under the PESQ and STOI evaluation indicators is higher than the other four comparative experiments. Therefore, the enhanced speech of RRSENet obtains the best speech quality and intelligibility, which also shows that RRSENet is good at the generalization ability on the mismatched noise test set is better [48].

Table 2. Experimental results of different network models under unseen noise conditions for PESQ and STOI. **BOLD** indicates the best result for each case.

Metrics	PESQ				STOI				
	SNR	−5 dB	0 dB	5 dB	Avg.	−5 dB	0 dB	5 dB	Avg.
Noisy		1.419	1.603	1.868	1.630	65.05	72.99	82.10	73.38
LogMMSE		1.439	1.654	1.941	1.678	65.23	73.66	82.67	73.85
Wiener		1.617	2.030	2.387	2.011	64.50	76.20	84.90	75.20
TCNN		1.785	2.054	2.359	2.066	74.76	83.71	89.88	82.79
AECNN		1.771	2.066	2.437	2.091	74.28	83.50	89.79	82.52
HDMNet		1.937	2.234	2.611	2.261	77.99	85.06	90.25	84.43
RLSNet		1.899	2.204	2.586	2.230	77.10	85.62	90.97	84.56
RRSENet		1.959	2.276	2.649	2.295	78.87	85.99	91.03	85.29

4.3.2. Module Comparisons

We then attempt to verify the effectiveness of the proposed hybrid dilated convolution module, that is, to combine conventional convolution to solve the problem of under-utilization of the local information of dilated convolution. The HDM module is compared with the full-dilated convolution module and the full-conventional convolution module for experiments to analyze the effects of different modules on speech enhancement performance. As shown in Figure 5, these two modules are very similar to HDM, as both include 1×1 convolution, feature fusion convolution and output 1×1 convolution, and residual connection, and the number of parameters of the two modules is consistent with HDM. The difference lies in the feature fusion convolution layer, where the full-dilated convolution uses two dilated convolutions and the full-conventional convolution uses two conventional convolutions.

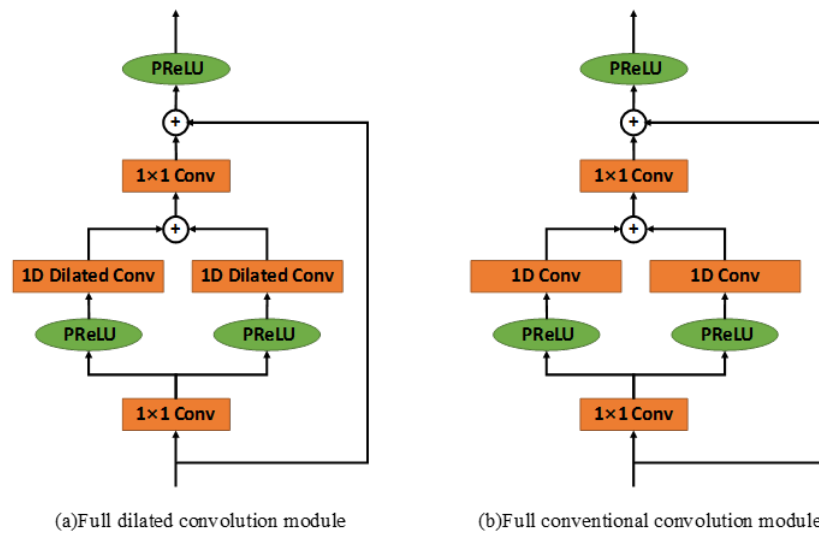


Figure 5. The full-dilated convolution module and full-conventional convolution module.

The network architecture of this experiment keeps the whole structure of the encoder and decoder unchanged. In the comparison of the modules, the GRU module of recursive learning is not used. Therefore, the network that uses HDM is defined as HDMNet. The network that used full-dilated convolution modules is defined as full-dilated convolution module network (FDMNet), and the dilation rate of dilated convolution in FDMNet is consistent with that in HDMNet. The network used the full-conventional convolution module network (FCMNet).

Table 3 shows the experimental results of PESQ and STOI for three different modules under seen noise conditions. Table 4 shows the experimental results of PESQ and STOI for three different modules under unseen noise conditions. The average PESQ value is the evaluation value of the average speech quality under different signal-to-noise ratios, the average STOI value represents the evaluation value of the average speech intelligibility under different signal-to-noise ratios, and “Avg.” represents the three types of signals under different evaluation indicators.

Table 3. Experimental results of different modules under seen noise conditions for PESQ and STOI. BOLD indicates the best result for each case.

Metrics	PESQ				STOI				
	SNR	−5 dB	0 dB	5 dB	Avg.	−5 dB	0 dB	5 dB	Avg.
Noisy		1.537	1.843	2.083	1.821	66.40	77.56	84.92	76.30
FCMNet		1.832	2.268	2.528	2.209	76.45	86.02	90.50	84.32
FDMNet		1.959	2.441	2.755	2.385	78.46	87.31	91.56	85.77
HDMNet		2.122	2.654	3.013	2.596	82.03	89.52	93.09	88.22

Table 4. Experimental results of different modules under unseen noise conditions for PESQ and STOI. **BOLD** indicates the best result for each case.

Metrics	PESQ				STOI				
	SNR	−5 dB	0 dB	5 dB	Avg.	−5 dB	0 dB	5 dB	Avg.
Noisy		1.419	1.603	1.868	1.630	65.05	72.99	82.10	73.38
FCMNet		1.714	1.976	2.279	1.990	74.24	82.19	88.17	81.53
FDMNet		1.903	2.183	2.499	2.195	76.79	83.66	89.18	83.20
HDMNet		1.937	2.234	2.611	2.261	77.99	85.06	90.25	84.43

From the results in Tables 3 and 4, it can be seen that HDMNet obtained the best results, followed by FDMNet, and FCMNet was the worst. The experiment proves: (1) that the dilated convolution is very effective in end-to-end speech enhancement tasks, greatly improving the enhancement effect of the network model. (2) The use of the hybrid dilated convolution model improves the evaluation index compared with the full dilated convolution model, which shows that the HDM makes full use of the information of the adjacent points of the speech waveform without losing the local feature information of the speech, and improves the enhancement effect of the model.

4.3.3. GRU Module and Recursive Times

In order to explore the effect of the time of recursive on RLSENet and RRSENet for speech enhancement, the time of recursive is taken from 1 to 5. Objective evaluations are performed on the seen noise test set and unseen noise test set, respectively.

Table 5 shows the PESQ and STOI values of RLSENet and RRSENet under seen noise conditions. As the number of recursion increases, RRSENet uses the memory mechanism to further learn the feature information between different stages, thereby promoting noise removal, improving the quality of speech and improving the intelligibility of speech.

Table 5. Experimental results of RLSENet and RRSENet under seen noise conditions. **BOLD** indicates the best result for each case.

Metrics	PESQ		STOI		
	Model	RLSENet	RRSENet	RLSENet	RRSENet
t1		2.351	2.326	84.94	83.45
t2		2.502	2.581	86.96	87.11
t3		2.591	2.623	88.64	88.50
t4		2.627	2.688	88.78	88.82
t5		2.625	2.704	88.75	88.22

Table 6 shows the PESQ and STOI values of the speech enhanced by the RLSENet and RRSENet modles. The PESQ value is the mean of speech quality under different signal-to-noise ratios in the case of unmatched noise. The STOI value is the mean of speech intelligibility under different signal-to-noise ratios. The results in Table 6 show that when $t = 4$, RRSENet achieves the best results under the two indicators of PESQ and STOI. As the number of recursion increases, the results of RRSENet are better than RLSENet, mainly due to the addition of the GRU module. Combining the two experiments, the higher the number of recursion, the better. In RRSENet, $t = 4$ can obtain better results.

Table 6. Experimental results of RLSENet and RRSENet under unseen noise conditions. **BOLD** indicates the best result for each case.

Metrics	PESQ		STOI		
	Model	RLSENet	RRSENet	RLSENet	RRSENet
t1		2.151	2.135	81.92	81.54
t2		2.170	2.238	81.89	83.13
t3		2.189	2.244	89.37	84.26
t4		2.230	2.268	84.56	84.59
t5		2.262	2.245	84.56	83.48

4.3.4. Speech Spectrogram

Figure 6 is the spectrogram of the synthesized speech enhanced by the RRSENet model, “t” is the number of recursion of RRSENet. From Figure 6, noisy speech is more disturbed than clean speech. In the spectrogram of the enhanced speech, as the number of recursion increases, the noise reduction effect of the speech is better. The enhanced speech spectrogram works very well, and speech quality is preserved intact. Through the analysis of the spectrogram, the effectiveness of RRSENet is proved.

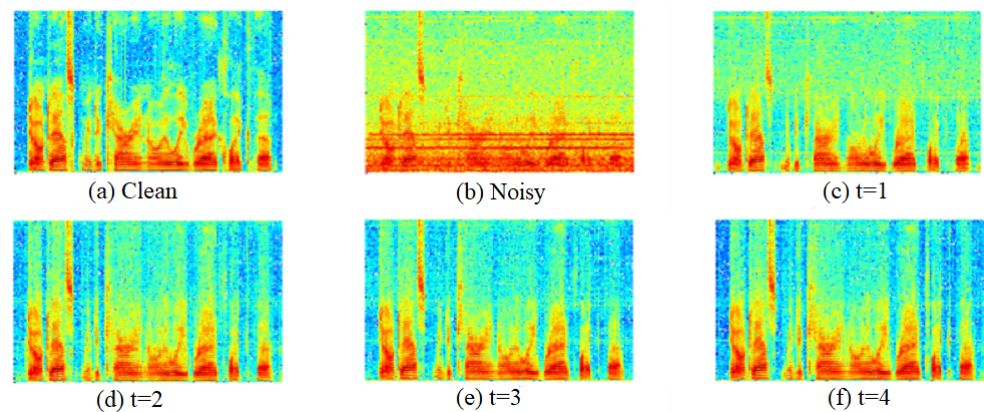


Figure 6. Spectrogram of synthesized speech enhanced by the RRSENet model.

To demonstrate the practicality of the RRSENet algorithm proposed in this chapter, five speech segments with ambient background noise were recorded in a real-life scenario with a sampling rate of 8 kHz. when using the RRSENet algorithm for speech enhancement, firstly, the adoption rate of the recorded speech was resampled operationally so that the adoption rate of the speech changed to 16 kHz, because the sampling rate of the speech during model training was 16 kHz. Secondly, after the enhancement of the speech, the sampling rate of the speech is downsampled to 8 kHz. Finally, the speech is visualized as shown in Figure 7.

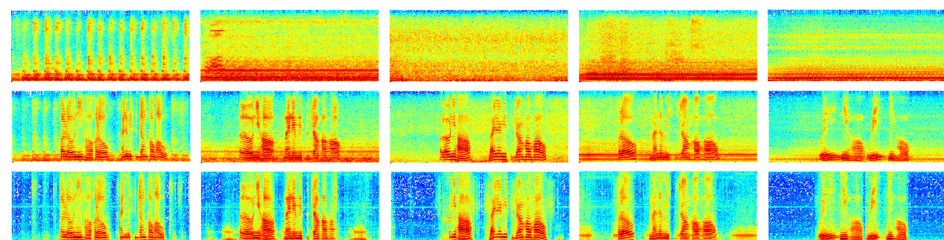


Figure 7. Spectrogram of real-world speech enhancement. The first row is the spectrogram of real-world noise, the second row is the spectrogram of real-world speech, the third row is the spectrogram of the enhancement speech

5. Conclusions

In the paper, a speech enhancement algorithm based on recursive learning with a hybrid dilated convolution model was proposed. A hybrid dilated convolution module (HDM) is proposed to solve the problem of insufficient utilization of local information in one-dimensional dilated convolution. Through HDM, the receptive field can be enhanced, the context information can be fully utilized, and the speech enhancement performance of the model can be improved. A recursive recurrent network training model is proposed, which solves the problems of a conventional network with many training parameters and a complex network structure. We improved the speech enhancement quality while reducing the training parameters. The experimental results showed that RRSENet performs better than the other two baseline time-domain models. Future research includes exploring the RRSENet model for other speech processing tasks such as de-reverberation and speaker separation.

Author Contributions: Methodology, Z.S. and Y.M.; software, Y.M.; validation, F.T.; investigation, X.F.; writing—original draft preparation, Y.M.; writing—review and editing, Z.S.; visualization, F.T.; supervision, X.F.; project administration, X.F.; funding acquisition, X.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work is partly supported by the Key Research and Development Program of Shaanxi (Program Nos. 2021ZDLGY15-01, 2021ZDLGY09-04, 2021GY-004 and 2020GY-050), the International Science and Technology Cooperation Research Project of Shenzhen (GJHZ20200731095204013), the National Natural Science Foundation of China (Grant No. 61772419).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are openly available in [41,42].

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. Ephraim, Y.; Malah, D. Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. *IEEE Trans. Acoust. Speech Signal Process.* **1984**, *32*, 1109–1121. [\[CrossRef\]](#)
3. Ephraim, Y.; Van Trees, H.L. A signal subspace approach for speech enhancement. *IEEE Trans. Speech Audio Process.* **1995**, *3*, 251–266. [\[CrossRef\]](#)
4. Wang, D. On ideal binary mask as the computational goal of auditory scene analysis. In *Speech Separation by Humans and Machines*; Springer: Berlin, Germany, 2005; pp. 181–197.
5. Wang, Y.; Narayanan, A.; Wang, D. On training targets for supervised speech separation. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2014**, *22*, 1849–1858. [\[CrossRef\]](#)
6. Xu, Y.; Du, J.; Dai, L.R.; Lee, C.H. A regression approach to speech enhancement based on deep neural networks. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2014**, *23*, 7–19. [\[CrossRef\]](#)
7. Tan, K.; Wang, D. A Convolutional Recurrent Neural Network for Real-Time Speech Enhancement. In Proceedings of the Interspeech, Hyderabad, India, 2–6 September 2018; pp. 3229–3233.
8. Zhao, H.; Zarar, S.; Tashev, I.; Lee, C.H. Convolutional-recurrent neural networks for speech enhancement. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 2401–2405.
9. Paliwal, K.; Wójcicki, K.; Shannon, B. The importance of phase in speech enhancement. *Speech Commun.* **2011**, *53*, 465–494. [\[CrossRef\]](#)
10. Pascual, S.; Bonafonte, A.; Serra, J. SEGAN: Speech enhancement generative adversarial network. *arXiv* **2017**, arXiv:1703.09452.
11. Kolbæk, M.; Tan, Z.H.; Jensen, S.H.; Jensen, J. On loss functions for supervised monaural time-domain speech enhancement. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2020**, *28*, 825–838. [\[CrossRef\]](#)
12. Abdulbaqi, J.; Gu, Y.; Chen, S.; Marsic, I. Residual Recurrent Neural Network for Speech Enhancement. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 6659–6663.

13. Fu, S.W.; Tsao, Y.; Lu, X.; Kawai, H. Raw waveform-based speech enhancement by fully convolutional networks. In Proceedings of the 2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Kuala Lumpur, Malaysia, 12–15 December 2017; pp. 006–012.
14. Oord, A.V.D.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; Kavukcuoglu, K. Wavenet: A generative model for raw audio. *arXiv* **2016**, arXiv:1609.03499.
15. Rethage, D.; Pons, J.; Serra, X. A wavenet for speech denoising. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 5069–5073.
16. Pandey, A.; Wang, D. TCNN: Temporal convolutional neural network for real-time speech enhancement in the time domain. In Proceedings of the ICASSP 2019—2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 6875–6879.
17. Ren, D.; Zuo, W.; Hu, Q.; Zhu, P.; Meng, D. Progressive Image Deraining Networks: A Better and Simpler Baseline. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
18. Tai, Y.; Yang, J.; Liu, X. Image Super-Resolution via Deep Recursive Residual Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
19. Berouti, M.; Schwartz, R.; Makhoul, J. Enhancement of speech corrupted by acoustic noise. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'79), Washington, DC, USA, 2–4 April 1979.
20. Sim, B.L.; Tong, Y.C.; Chang, J.S.; Tan, C.T. A parametric formulation of the generalized spectral subtraction method. *IEEE Trans. Speech Audio Process.* **1998**, *6*, 328–337. [[CrossRef](#)]
21. Kamath, S.D.; Loizou, P.C. A multi-band spectral subtraction method for enhancing speech corrupted by colored noise. In Proceedings of the ICASSP international Conference on Acoustics Speech and Signal Processing, Orlando, FL, USA, 13–17 May 2002; Volume 4.
22. Sovka, P.; Pollak, P.; Kybic, J. Extended Spectral Subtraction. In Proceedings of the 8th European Signal Processing Conference, Trieste, Italy, 10–13 September 1996.
23. Cohen, I. Relaxed statistical model for speech enhancement and a priori SNR estimation. *IEEE Trans. Speech Audio Process.* **2005**, *13*, 870–881. [[CrossRef](#)]
24. Cohen, I.; Berdugo, B. Speech enhancement for non-stationary noise environments. *Signal Process.* **2001**, *81*, 2403–2418. [[CrossRef](#)]
25. Hasan, M.K.; Salahuddin, S.; Khan, M.R. A modified a priori SNR for speech enhancement using spectral subtraction rules. *IEEE Signal Process. Lett.* **2004**, *11*, 450–453. [[CrossRef](#)]
26. Cappe, O. Elimination of the musical noise phenomenon with the Ephraim and Malah noise suppressor. *IEEE Trans. Speech Audio Process.* **1994**, *2*, 345–349. [[CrossRef](#)]
27. Li, A.; Zheng, C.; Cheng, L.; Peng, R.; Li, X. A Time-domain Monaural Speech Enhancement with Feedback Learning. In Proceedings of the 2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Auckland, New Zealand, 7–10 December 2020; pp. 769–774.
28. Yuliani, A.R.; Amri, M.F.; Suryawati, E.; Ramdan, A.; Pardede, H.F. Speech Enhancement Using Deep Learning Methods: A Review. *J. Elektron. Dan Telekomun.* **2021**, *21*, 19–26. [[CrossRef](#)]
29. Yan, Z.; Xu, B.; Giri, R.; Tao, Z. Perceptually Guided Speech Enhancement Using Deep Neural Networks. In Proceedings of the ICASSP 2018—2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018.
30. Karjol, P.; Ajay Kumar, M.; Ghosh, P.K. Speech Enhancement Using Multiple Deep Neural Networks. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 5049–5052.
31. Xu, Y.; Du, J.; Huang, Z.; Dai, L.R.; Lee, C.H. Multi-objective learning and mask-based post-processing for deep neural network based speech enhancement. *arXiv* **2017**, arXiv:1703.07172.
32. Zaremba, W.; Sutskever, I.; Vinyals, O. Recurrent Neural Network Regularization. *arXiv* **2015**, arXiv:1409.2329.
33. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)]
34. Cho, K.; Merriënboer, B.V.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv* **2014**, arXiv:1406.1078.
35. Gao, T.; Du, J.; Dai, L.R.; Lee, C.H. Densely Connected Progressive Learning for LSTM-Based Speech Enhancement. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 5054–5058.
36. Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv* **2015**, arXiv:1511.07122.
37. Ye, S.; Hu, X.; Xu, X. TdGAN: Temporal Dilated Convolutional Generative Adversarial Network for End-to-end Speech Enhancement. *arXiv* **2020**, arXiv:2008.07787.
38. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
39. He, K.; Zhang, X.; Ren, S.; Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1026–1034.
40. Noh, H.; Hong, S.; Han, B. Learning deconvolution network for semantic Segmentation. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1520–1528.

41. Garofolo, J.S.; Lamel, L.F.; Fisher, W.M.; Fiscus, J.G.; Pallett, D.S. DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus CD-ROM. NIST Speech Disc 1-1.1. NASA STI/Recon Technical Report n. 1993. Volume 93, p. 27403. Available online: <https://nvlpubs.nist.gov/nistpubs/Legacy/IR/nistir4930.pdf> (accessed on 8 March 2022).
42. Varga, A.; Steeneken, H.J. 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.* **1993**, *12*, 247–251. [[CrossRef](#)]
43. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
44. Ephraim, Y.; Malah, D. Speech enhancement using a minimum mean-square error log-spectral amplitude estimator. *IEEE Trans. Acoust. Speech Signal Process.* **1985**, *33*, 443–445. [[CrossRef](#)]
45. Pandey, A.; Wang, D. A New Framework for CNN-Based Speech Enhancement in the Time Domain. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2019**, *27*, 1179–1188. [[CrossRef](#)]
46. Rix, A.W.; Beerends, J.G.; Hollier, M.P.; Hekstra, A.P. Perceptual Evaluation of Speech Quality (PESQ): A New Method for Speech Quality Assessment of Telephone Networks and Codecs. In Proceedings of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'01), Salt Lake City, UT, USA, 7–11 May 2001.
47. Taal, C.H.; Hendriks, R.C.; Heusdens, R.; Jensen, J. An Algorithm for Intelligibility Prediction of Time–Frequency Weighted Noisy Speech. *IEEE Trans. Audio Speech Lang. Process.* **2011**, *19*, 2125–2136. [[CrossRef](#)]
48. Demiar, J.; Schuurmans, D. Statistical Comparisons of Classifiers over Multiple Data Sets. *J. Mach. Learn. Res.* **2006**, *7*, 1–30.