FTDCN: Full Two-Dimensional Convolution Network for Speech Enhancement in Time-Frequency Domain

Maoqing Liu[1(✉)], Hongqing Liu[1], Yi Zhou[1], and Lu Gan[2]

[1] School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China
s200131275@stu.cqupt.edu.cn

[2] College of Engineering, Design and Physical Science, Brunel University, London UB8 3PH, UK

Abstract. The dual-path structure achieves superior performance in monaural speech enhancement (SE), demonstrating the importance of modeling the long-range spectral patterns of a single frame. In this paper, two novel causal temporal convolutional network (TCN) modules, inter-frame complex-valued two-dimensional TCN (Inter-CTTCN) and intra-frame complex-valued two-dimensional TCN (Intra-CTTCN), are proposed to capture the long-range spectral dependence within a single frame and the long-term dependence between frames, respectively. These two lightweight TCN components, which are composed entirely of two-dimensional convolutions, maintain a high dimension feature representation that facilitates the distinction between speech and noise. We join the Inter-CTTCN and Intra-CTTCN with a gated complex-valued convolutional encoder and decoder structure to design a full two-dimensional convolutional network (FTDCN) for SE in the time-frequency (T-F) domain. Using noisy speech as input, the proposed model was experimentally evaluated on the datasets of Interspeech 2020 Deep Noise Suppression Challenge (DNS Challenge 2020). The NB-PESQ of our proposed model exceeds the DNS Challenge 2020 first-placed model by 0.19 and our model requires only 0.8 M parameters.

Keywords: Speech enhancement · Dual-path · End-to-End

1 Introduction

Speech enhancement (SE) can substantially improve speech quality in applications such as hearing aids, pickup devices, and audio-video conferencing systems, enhancing users' experiences. With the developments of deep learning techniques in recent years, the performances of deep learning SE methods surpass those of traditional methods.

The temporal dependency of speech is what deep learning methods mainly model. In that case, long short-term memory (LSTM) [1,2] and temporal convolutional network (TCN) [3,4] are successively proposed to extract long-term

F. Gao et al. (Eds.): ChinaCom 2022, LNICST 500, pp. 97–108, 2023.
https://doi.org/10.1007/978-3-031-34790-0_8

temporal relationships of speech. Compared to LSTM, TCN has the advantage of a flexible receptive field and can be parallelized to improve the model running speed computationally. The TCN was first proposed for sequence modeling, proving that it is superior to LSTM. The speech separation model Conv-TasNet [5] used the TCN structure as a backbone and achieved excellent results. Subsequently, TCNN [6] applied TCN with a convolutional encoder-decoder (CED) structure for the time domain SE task and demonstrated the effectiveness of TCN. For long-range spectral patterns, the recent performance of PHASEN [7], DPCRN [8], and DCCRN+ [9] also demonstrates the importance of TCN structure.

In light of this, we design two causal TCN modules, Intra-CTTCN and Inter-CTTCN, to model the long-range spectral dependence within a single frame and the long-term dependence between frames, respectively. Compared with the original TCN structure, our proposed TCN module makes several optimisations. The two modules consist entirely of two-dimensional convolutions to maintain the advantage of high-dimensional features over low-dimensional ones and reduce the model's size. Adding complex-valued operations make the number of parameters reduced by half. The dilation factor is set to a power of 3 to avoid overlapping operations, thus allowing a larger receptive field of the exact computation. Maintain an appropriate Spectral resolution, as it affects the modelling of the intra-frame harmonic relations. The intra-frame module has frame group convolution kernels in each convolution layer used to model the harmonic relations of a single frame separately.

Finally, a gated complex-valued CED structure that joins Intra-CTTCN and Inter-CTTCN is developed to obtain a full two-dimensional convolution network termed FTDCN. Since the Spectral resolution is large enough, using LayerNorm to normalize each frame in the whole model can achieve better results while maintaining causality. The experiments show that FTDCN achieves better objective metrics than the top model in DNS Challenge 2020, and with a smaller model size.

2 System Overview

In this section, we will show details of the proposed architecture. Figure 1 depicts the general structure of FTDCN, which consists of STFT, encoder, dual-path complex-valued TCN module, decoder, and iSTFT. The output shapes along the channel and frequency axis are 32 and 128 for encoder1 and encoder2, respectively, and 64 and 64 for the other encoders. The output shape of the decoder is symmetrical to the encoder.

2.1 Dual-Path Complex-Valued TCN Module

Inspired by DPCRN [8] and TCNN [6], we propose a dual-path complex-valued TCN module (DPCTM), a causal lightweight module in the T-F domain. It consists of an intra-frame module and an inter-frame module for modeling the

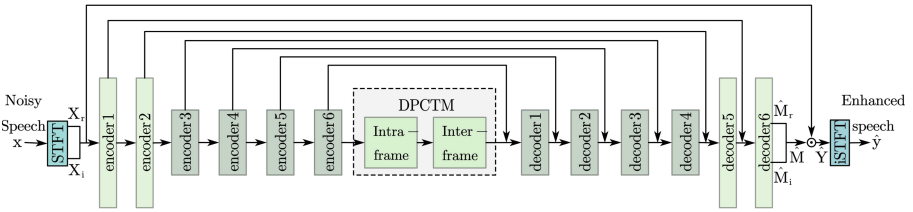

Fig. 1. FTDCN network.

long-range spectral dependence within a single frame and the long-term dependence between frames, respectively. The backbone of the inter-frame module is composed of multiple Inter-CTTCN modules. There is a pointwise convolution (1×1-Conv) at the front and back of the Inter-CTTCNs to perform a fully connected operation on the input channel axes. The intra-frame module is similar to the inter-frame module, but it consists of three parts: 1×1-Conv, Intra-CTTCN modules, and 1×1-Conv, where the number of groups of 1×1-Conv is set to T, which is used to equalize the scale of each frame.

2.2 Inter-frame Complex-Valued Two-Dimensional TCN Module (Inter-CTTCN)

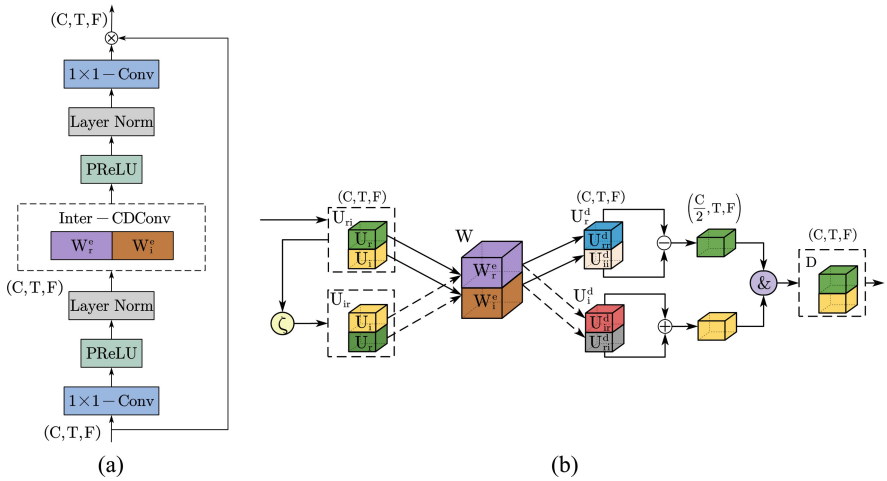

Fig. 2. (a)Proposed Inter-CTTCN. (b)Proposed Inter-CDConv, ζ denotes swapping the order of the two parts on the C axis

We propose Inter-CTTCN for extracting long-term inter-frame dependencies of high-level features. Traditionally, to extract long-term context dependencies with modules like TCN, the encoder output is usually flattened to two dimensions, C

$\times F$ and T. This approach does not fully exploit the relationship between noise and speech in the T-F domain for high-dimensional features, failing to separate them. Therefore, the proposed Inter-CTTCN replaces all the 1D convolutions in TCN with 2D convolutions to exploit the high-dimensional features in the T-F domain. The purpose of this is to fully exploit the relationship between noise and speech in the T-F domain and to reduce the model size greatly.

Our proposed Inter-CTTCN has a TCN-like topology, as shown in Fig. 2(a), where 1×1-Conv is a 2D convolutional layer with a convolutional kernel size of one in both T and F axes. The 1×1-Conv, PReLU, and LayerNorm are used to increase the nonlinearity of the network and act first on the module input $U \in \mathbb{R}^{C\times T\times F}$. In the middle, the inter-frame complex-valued dilated convolution (Inter-CDConv) extends the receptive field of the model to extract the high-level signal's long-term frame-to-frame dependence. The Inter-CDConv is also followed by PReLU and LayerNorm. After that, a layer of 1×1-Conv is employed, which is a fully connected operation on the channel axis of the signal. Finally, the residual path is used to add the module input to the output of the final layer. In a nutshell, the output U_e of the Inter-CTTCN is calculated by

$$\tilde{U}^1 = g_1(U; \psi_1), \tag{1}$$

$$\tilde{U}^{cd} = g_{cd}(\beta(\delta(\tilde{U}^1; \psi_c d))), \tag{2}$$

$$\tilde{U}_e = g_2(\beta(\delta(\tilde{U}^{cd}; \psi_2))) + U, \tag{3}$$

where δ and β represent PReLU and LayerNorm, respectively, $(\tilde{\cdot})^1$ and $(\tilde{\cdot})^{cd}$ denotes the outputs of 1×1-Conv and Inter-CDConv, respectively, g_1, g_{cd} and g_2 represent the functions of the corresponding three modules with the parameter set $\psi_{(\cdot)}$,

Our proposed Inter-CDConv computes the real and imaginary parts in a complex-valued manner, achieving a great performance with half of the model parameters for ordinary dilated convolution. The structure of Inter-CDConv is shown in Fig. 2(b). Let the complex-valued input features be $U_{ri} = U_r \& U_i$, where $U_{ri} \in \mathbb{R}^{C\times T\times F}$, and & denotes the splicing in the first axis. Swapping the order of Ur and Ui to produces $U_{ir} = U_i \& U_r$. The Inter-CDConv has two sets of convolution kernels of $W = W_r^e \& W_i^e$, where $W \in \mathbb{R}^{C\times C\times K_t\times K_f}$ in which K_t and K_f denote the size of the convolution kernels in T and F axis, respectively.

By applying Inter-CDConv to U_{ri} and U_{ri}, we obtain $U_r^d = U_{rr}^d \& U_{ir}^d$ and $U_i^d = U_{ir}^d \& U_{ri}^d$, respectively, in which $U_{rr}^d = U_r \times W_r^e$, $U_{ii}^d = U_i^d \times W_i^e$, $U_{ir}^d = U_i^d \times W_r^e$ and $U_{ri}^d = U_r^d \times W_i^e$. Finally, the output $D \in \mathbb{R}^{C\times T\times F}$ of Inter-CDConv is calculated as $D = (U_{rr}^d - U_{ir}^d) \& (U_{ir}^d - U_{ri}^d)$.

2.3 Intra-frame Complex-Valued Two-Dimensional TCN Module (Intra-CTTCN)

Recent studies [7–9] have demonstrated the importance of capturing the long-range dependence of spectrograms within a single frame. The reason is that the fundamental frequencies of speech and its harmonics are strongly correlated and

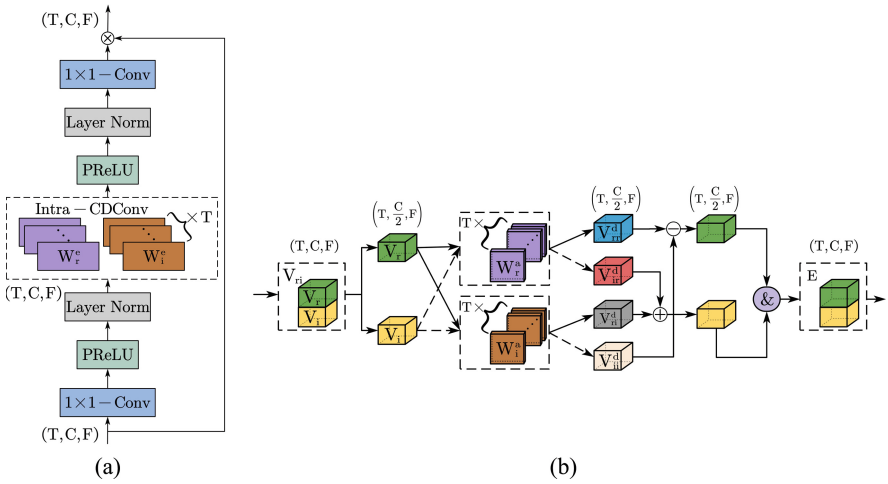

Fig. 3. (a)Proposed Intra-CTTCN. (b)Proposed Intra-CDConv

often distributed in the long-range spectrum within a given frame. Therefore, we propose Intra-CTTCN to capture long-range spectrogram dependencies within a single frame.

The topology of Intra-CTTCN is similar to Inter-CTTCN, shown in Fig. 3(a). The input $V \in \mathbb{R}^{T\times C\times F}$ is fed into the first layer of the module 1×1-Conv, which has T sets of independent convolution kernels of size 1 in both C and F axes, which equalizes the size of each frame in the time scale. The 1×1-Conv is followed by PReLU and LayerNorm, where LayerNorm is normalized to the [C, F] dimension of each frame during training and evaluation. The intermediate Intra-CDConv is shown in Fig. 3(b), which consists of T groups of convolution kernels to perform convolution operations on the spectral map of each frame separately to extract the spectral map long-range dependence. Each group has two convolution kernels for real and imaginary parts, similar to the complex-valued operation equation, the computation process of Intra-CDConv is

$$\begin{aligned} V_{rr}^{d} &= V_r \times W_r^{a}, \quad V_{ii}^{d} = V_i \times W_i^{a}, \\ V_{ri}^{d} &= V_r \times W_i^{a}, \quad V_{ir}^{d} = V_i \times W_r^{a}, \end{aligned} \tag{4}$$

$$E = (V_{rr}^{d} - V_{ii}^{d})\&(V_{ri}^{d} + V_{ir}^{d}). \tag{5}$$

The Intra-CDConv is followed by PReLU, LayerNorm, and 1×1 conv, and the residual path is also added.

2.4 Gated Complex-Valued Convolutional Encoder and Decoder

The convolutional encoder and decoder (CED) structure has been widely adopted in many applications [10–12]. A complex-valued CED architecture consists of six complex-valued encoder layers and corresponding complex-valued

decoders into mirrors, connecting each decoder's input to the corresponding encoder output with a skip connection to reduce information loss, is developed in DCCRN [13]. Each encoder layer consists of a complex-valued convolutional layer, a complex-valued BatchNorm, and a real-valued PPeLU. Each complex-valued convolution layer consists of two 2D convolutions with stride size 2 along the frequency axis, gradually and exponentially reducing the spectrum's resolution and exponentially increasing the channels to extract high-level features. Each decoder comprises a complex-valued transposed convolutional layer, a complex-valued BatchNorm, and a real-valued PPeLU. Among them, the complex-valued transpose convolution layer consists of a 2D transpose convolution used to reconstruct the target spectral map.

Recent studies also have shown that the gating mechanism is effective [14–16]. The gating mechanism reduces gradient disappearance in deep networks by providing linear paths for gradients while preserving nonlinear capabilities. Therefore, we propose a gated complex-valued CED by combining the complex-valued CED and gating mechanism, and the computation process is

$$\begin{aligned} \tilde{U}^{rr} &= g_r(I_r;\psi_r), \quad \tilde{U}^{ii} = g_i(I_i;\psi_i), \\ \tilde{U}^{ri} &= g_i(I_r;\psi_i), \quad \tilde{U}^{ir} = g_r(I_i;\psi_r), \end{aligned} \tag{6}$$

$$\tilde{U}^{cc} = (\tilde{U}^{rr} - \tilde{U}^{ii})\&(\tilde{U}^{ri} + \tilde{U}^{ir}), \tag{7}$$

$$\tilde{U}^{G} = \tilde{U}^{cc_2} + g_t(\tilde{U}^{cc_2};\psi_t), \tag{8}$$

$$\tilde{U}_o = \beta(\delta(\tilde{U}^{G})), \tag{9}$$

where $(\tilde{\cdot})^{rr}$, $(\tilde{\cdot})^{ii}$, $(\tilde{\cdot})^{ri}$, $(\tilde{\cdot})^{ir}$ represents real part input through real part convolution kernel, imaginary part input through imaginary part convolution kernel, real part input through imaginary part convolution kernel, and imaginary part input through real part convolution kernel, respecitively, $(\tilde{\cdot})^{cc_1}$ and $(\tilde{\cdot})^{cc_2}$ represent the output of two complex-valued convolutions that do not share weights, g_r, g_i and g_t represent the functions of the corresponding three modules with the parameter set $\psi_{(\cdot)}$.

2.5 The Final Proposed Framework

DPCTM. We place a 1×1-Conv at the head and tail of the Inter-frame module and stack eight Inter-CTTCNs in the middle. In the Inter-CTTCN, the Inter-CDConv has T sets of convolutional kernels, and the size of each convolutional kernel in the time and frequency axes is set to [3,3]. The dilation factor d on the time axis is set to [1,3,9,27,1,3,9,27], and the dilatation factor on the frequency axis is kept as 1. To maintain the causal constraint, we complement the signal with $(3-1)\times d$ zeros at the top of the time axis before Inter-CDConv. The normalized shape of LayerNorm is set to F to maintain the causal constraint.

Similarly, we place a 1×1-Conv at the head and tail of the Intra-frame module, with six Intra-CTTCNs stacking in between. In Intra-CTTCN, Intra-CDConv has only one set of conv kernels, each of size [3,3] conducting convolution

on the channel and frequency axes of the signal. Since there is no causal constraint in the channel and frequency axes, the expansion factors of both axes are set to [1,3,9,1,3,9] and the signal is complemented by $\frac{(3-1)\times d}{2}$ zeros in front and behind each of the two axes before Intra-CDConv. To obtain a better normalization performance, we normalize the C and F axes of the signal together, so we set the normalized shape to [C,F], which is still under the causal constraint.

Gated Complex-Valued Convolutional Encoder and Decoder. Within each encoder block, the size of the convolution kernel on the time and frequency axes is set to (2,5). The stride for the cross-correlation of the first and third encoder is set to (1,2) and the others are set to (1,1). This is because we empirically found that the resolution of the input spectrum of DPCTM would reduce the SE effect if it is too small, but it would also increase the computation cost if it is too high. Considering all these factors, we change the frequency in the encoder to a quarter of the original one. Similarly, we set the number of channels in the final output of encoder to 64, compared to the 256 channels of DCCRN, since we want a smaller model size. As with Inter-CTTCN, we set the normalized shape of the LayerNorm after Gated Complex-Valued Convolutional to F to maintain the causal constraint. Note that we are looking forward to one frame in each encoder-decoder pair, which results in a 37.5ms look ahead.

Detailed Parameter Settings. The detailed parameter settings of our proposed model are shown in Table 1. We split the real and imaginary parts of the STFT output into two channels and delete the first point, fill a zero in front of iSTFT and merge the real and imaginary parts of the input. The size of the input and output in the frequency domain is *channels* × *frames* × *frequency*, except for Intra-CTTCN, which is *frames* × *channels* × *frequency*. For encoder and decoder, the parameters represent *kernel size alone height* × *kernel size alone width, (stride along height, stride along width), input channels, output channels*. For Intra-CTTCN, the parameters represent *groups of input, kernel size alone height* × *kernel size alone width, (stride along height, stride along width), channels*. Inter-CTTCN has one less paramete groups of input than Intra-CTTCN.

2.6 Loss Function

We use complex ratio mask (CRM) [17] as the learning target of FDTCN. This network is trained to estimate the mask $\hat{M} = \hat{M}_R + j\hat{M}_I$, where $\hat{M} \in \mathbb{C}^{C\times T\times F}$. Combining $\hat{M}$ with the input $X = X_R + jX_I$, where $X \in \mathbb{C}^{C\times T\times F}$, yields the output $\hat{Y}$, given by

$$\hat{Y} = \hat{M} \times X = (\hat{M}_R + j\hat{M}_I)(X_R + jX_I), \tag{10}$$

Considering the loss functions in the frequency and time domains can better utilize information from both domains, so we propose the following the joint loss function, given by

$$\hat{Y}_{mag} = \|\hat{Y}_r + \hat{Y}_i\|_F, \tag{11}$$

Table 1. Detailed parameter settings of the proposed FTDCN. The meaning of the values represented is described in the text.

layer name	input size	hyperparameters	output size
STFT	1×16000	–	514×T
encoder1	2×T×256	2×5,(1,2),2,32	32×T×128
encoder2	32×T×128	2×5, (1,1), 32, 32	32×T×128
encoder3	32×T×128	2×5, (1,2), 32, 64	64×T×64
encoder4	64×T×64	2×5,(1,1),64,64	64×T×64
encoder5	64×T×64	2×5,(1,1),64,64	64×T×64
encoder6	64×T×64	2×5,(1,1),64,64	64×T×64
reshape	64×T×64	-	T×64×64
Intra-CTTCN	T×64×64	T,1×1,(1,1),T $\left.\begin{pmatrix} T,1\times1,(1,1),\mathrm{T} \\ T,3\times3,(1,1),\mathrm{T} \\ T,3\times3,(1,1),\mathrm{T} \\ T,1\times1,(1,1),\mathrm{T} \end{pmatrix} \begin{pmatrix} T,1\times1,(1,1),\mathrm{T} \\ T,3\times3,(3,3),\mathrm{T} \\ T,3\times3,(3,3),\mathrm{T} \\ T,1\times1,(1,1),\mathrm{T} \end{pmatrix} \begin{pmatrix} T,1\times1,(1,1),\mathrm{T} \\ T,3\times3,(9,9),\mathrm{T} \\ T,3\times3,(9,9),\mathrm{T} \\ T,1\times1,(1,1),\mathrm{T} \end{pmatrix} \right\} \times 2$ T,1×1,(1,1),T	T×64×64
reshape	T,64,64	-	64 ×T×64
Inter-CTTCN	64×T×64	1×1,(1,1),64 $\left.\begin{pmatrix} 1\times1,(1,1),64 \\ 3\times3,(1,1),64 \\ 1\times1,(1,1),64 \end{pmatrix} \begin{pmatrix} 1\times1,(1,1),64 \\ 3\times3,(3,1),64 \\ 1\times1,(1,1),64 \end{pmatrix} \begin{pmatrix} 1\times1,(1,1),64 \\ 3\times3,(9,1),64 \\ 1\times1,(1,1),64 \end{pmatrix} \begin{pmatrix} 1\times1,(1,1),64 \\ 3\times3,(27,1),64 \\ 1\times1,(1,1),64 \end{pmatrix} \right\} \times 2$ 1×1,(1,1),64	T×64×64
decoder1	128×T×64	2×5,(1,1),64	64×T×64
decoder2	128×T×64	2×5,(1,1),64	64×T×64
decoder3	128×T×64	2×5,(1,1),64	64×T×64
decoder4	128×T×64	2×5,(1,2),64	32×T×128
decoder5	64×T×64	2×5,(1,1),64	32×T×128
decoder6	64×T×64	2×5,(1,2),64	2×T×256
iSTFT	514×T	-	1×16000

$$Y_{mag} = \|Y_r + Y_i\|_F, \tag{12}$$

$$L_F = 10log_{10}(\|\hat{Y}_r - Y_r\|_F^2 + \|\hat{Y}_i - Y_i\|_F^2 + \|\hat{Y}_{mag} - Y_{mag}\|_F^2), \tag{13}$$

$$L_T = 10log_{10}\frac{\|\frac{\langle\hat{y},y\rangle\hat{y}}{\|Y\|_F^2}\|_F^2}{\|\hat{y}-y\|_F^2}, \tag{14}$$

$$L = L_T + L_F, \tag{15}$$

where $\langle\cdot,\cdot\rangle$ is the inner product of two matrices, and $\|\cdot\|_F$ denotes Frobenius norm.

3 Experimental Setup

3.1 Dataset

In this paper, we use the DNS Challenge 2020 dataset [18] to evaluate FTDCN with the sampling rate at 16 kHz. The clean speech consists of 500 h of speech from 2150 speakers, and noise contains 150 types of noise for 180 h. We made two training dataset to verify the effectiveness of the model, with and without reverberations. For 500 h of the training set with reverberation, the clean speech was first convoluted with random room impulse responses mixed with a reverberation time of 0.3 to 0.6 s and then synthesized randomly with the noise of signal-to-noise ratio (SNR) of –5 to 20 dB. For the training set of 500 h without reverberation, we synthesize clean speech and noise with SNRs in the range of –5 to 20 dB randomly. We use the test set provided by DNS Challenge 2020 containing 150 voices of ten seconds in length to compare with other methods.

3.2 Training Setup

We set the window length to 25 ms, the frameshift to 6.25 ms, and a 512-point FFT. We set the initial learning rate to 0.001 and the weight decay to 0.00001. When we observe that the decreasing trend of the loss curve tends to level off, we usually set it to half of the previous one, and finally, we trained 51 epochs with a learning rate of 0.000025.

3.3 Ablation Studies

Two baseline models are trained for the purposes of ablation studies. First, we trained a model, called FTDCN-E, which differs from FTDCN only in that it removes the intra-frame module from FTDCN. Second, we trained a model that doubles C and halves F of the DPCTM input, calling it FTDCN-D, where the (*stride along height,stride along width*) and *channels* of the six encoders of FTDCN-D are set to (*(1,2), (1,1), (1,2), (1,1), (1,2), (1,1)), (1,2), (1,1)*), (*32,32,64,64,128,128*), and Decoder and DPCTM are changed correspondingly.

4 Results

In Table 2, we compare FTDCN with DNS Challenge 2020 top-ranked methods and baselines together, where NSNet [19] is the DNS Challenge 2020's official baseline network, DTLN [20] and DCCRN [13] are for the real-time track, and Conv-TasNet [5] is for the non-real-time track. Under the condition of the same test set, we directly use the objective scores provided by these authors, where "-"

Table 2. NB-PESQ, WB-PESQ, STOI, SI-SDR on DNS Challenge 2020 test set.

Method	Para.(M)	no Reverb				Reverb			
		NB-PESQ	WB-PESQ	STOI	SI-SDR	NB-PESQ	WB-PESQ	STOI	SI-SDR
Noisy	–	2.45	1.58	91.52	9.07	2.75	1.82	86.62	9.03
NSNet	5.1	2.87	2.15	90.47	15.61	3.08	2.37	90.43	14.72
DTLN	1.0	3.04	–	94.76	16.34	2.70	–	84.68	10.53
Conv-TasNet	5.08	–	2.73	–	–	–	2.75	–	–
DCCRN	3.7	3.27	–	–	–	3.08	–	–	–
TRU-Net	**0.38**	3.36	2.86	96.32	17.55	3.35	2.74	91.29	14.87
FTDCN-E	0.69	3.34	2.81	96.52	18.53	3.29	2.57	90.58	14.57
FTDCN-D	1.43	3.36	2.80	96.41	18.65	3.31	2.62	90.37	14.40
FTDCN	0.82	**3.46**	**2.96**	**96.91**	**19.23**	**3.40**	**2.79**	**91.70**	**15.51**

means this score is not provided, "Para" indicates the parameters of the model, "no Reverb" and "Reverb" represent the test set without and with reverberations, respectively.

The following conclusions can be drawn from Table 2:

1) Comparing FTDCN with FTDCN-E, it shows that the proposed intra-frame module greatly enhances the SE capability with only a slightly increased model size. It illustrates the importance of capturing the long-range spectral dependence and the effectiveness of the intra-frame module.
2) Comparing FTDCN with FTDCN-D, it illustrates the importance of proper frequency resolution, showing that minor F may attenuate the model's ability to model long-range dependencies of speech.
3) Comparing FTDCN with the top-ranked models from DNS Challenge 2020, it demonstrates the excellent noise reduction capability is obtained by our proposed FTDCN, even with a small model size. FTDCN-D also illustrates the importance of maintaining the high dimensionality of the features compared to other methods.

5 Conclusion

In this work, we propose an end-to-end semi-causal lightweight speech enhancement method. To model the intra- and inter-frame dependencies of the speech, we propose two lightweight modules, termed Intra-CTTCN and Inter-CTTCN, by using only convolutions. To utilize the Intra-CTTCN and Inter-CTTCN, we develop a gated complex CED structure to model the real and imaginary parts of the input. The results show that the performance of the proposed model is superior to the top-ranked model in DNS Challenge 2020, even with smaller model size. Future research will explore the possibility of deploying our model to edge devices.

References

1. Tan, K., Wang, D.L.: A convolutional recurrent neural network for real-time speech enhancement. Interspeech **2018**, 3229–3233 (2018)
2. Luo, Y., Mesgarani, N.: Tasnet: time-domain audio separation network for real-time, single-channel speech separation. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 696–700. IEEE (2018)
3. Bai, S., Kolter, J.Z., Koltun, V.: An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint arXiv:1803.01271 (2018)
4. Kishore, V., Tiwari, N., Paramasivam, P.: Improved speech enhancement using tcn with multiple encoder-decoder layers. In: Interspeech, pp. 4531–4535 (2020)
5. Luo, Y., Mesgarani, N.: Conv-tasnet: Surpassing ideal time-frequency magnitude masking for speech separation. IEEE/ACM Trans. Audio Speech Lang. Proc. **27**(8), 1256–1266 (2019)
6. Pandey. A., Wang, D.: Tcnn: Temporal convolutional neural network for real-time speech enhancement in the time domain. In: ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6875–6879. IEEE (2019)
7. Yin, D., Luo, C., Xiong, Z., Zeng, W.: Phasen: A phase-and-harmonics-aware speech enhancement network. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 9458–9465 (2020)
8. Le, X., Chen, H., Chen, K., Lu, J.: Dpcrn: Dual-path convolution recurrent network for single channel speech enhancement. arXiv preprint arXiv:2107.05429 (2021)
9. Lv, S., Hu, Y., Zhang, S., Xie, L.: Dccrn+: Channel-wise subband dccrn with snr estimation for speech enhancement. arXiv preprint arXiv:2106.08672 (2021)
10. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. arXiv preprint arXiv:1505.04597 (2015)
11. Zhao, S., Nguyen, T.H., Ma, B.: Monaural speech enhancement with complex convolutional block attention module and joint time frequency losses. In: ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6648–6652. IEEE (2021)
12. Choi, H.-S., Kim, J.-H., Huh, J., Kim, A., Ha, J.-W., Lee, K.: Phase-aware speech enhancement with deep complex u-net. In: International Conference on Learning Representations (2018)
13. Hu, Y., et al.: Dccrn: Deep complex convolution recurrent network for phase-aware speech enhancement. arXiv preprint arXiv:2008.00264 (2020)
14. Van Oord, A., Kalchbrenner, N., Kavukcuoglu, K.: Pixel recurrent neural networks. In: International Conference on Machine Learning, pp. 1747–1756. PMLR (2016)
15. Dauphin, Y.N., Fan, A., Auli, M., Grangier, D.: Language modeling with gated convolutional networks. In: International Conference on Machine Learning, pp. 933–941. PMLR (2017)
16. Van den Oord, A., Kalchbrenner, N., Espeholt, L., Vinyals, O., Graves, A., et al.: Conditional image generation with pixelcnn decoders. In: Advances in Neural Information Processing Systems 29 (2016)
17. Williamson, D.S., Wang, Y., Wang, D.: Complex ratio masking for monaural speech separation. IEEE/ACM Trans. Audio Speech Lang. Proc. **24**(3), 483–492 (2015)
18. Reddy, C.K.A., et al.: The interspeech 2020 deep noise suppression challenge: Datasets, subjective speech quality and testing framework. arXiv preprint arXiv:2001.08662 (2020)

19. Xia, Y., Braun, S., Reddy, C.K.A., Dubey, H., Cutler, R., Tashev, I.: Weighted speech distortion losses for neural-network-based real-time speech enhancement. In: ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 871–875. IEEE (2020)
20. Westhausen, N.L., Meyer, B.T.: Dual-signal transformation lstm network for real-time noise suppression. arXiv preprint arXiv:2005.07551 (2020)