DOI: 10.21917/ijct.2026.0566

MULTI-INPUT DEEP COMPLEX CONVOLUTION RECURRENT NETWORK BASED JOINT ACOUSTIC ECHO CANCELLATION AND BACKGROUND NOISE SUPPRESSION

Kum-Song Pak[1], Chol-I Om[2], Kwon Kim[3], Chol-Ui Ri[4] and Chol-Nam Om[5]

[1,3,4,5]Institute of Information Technology, High-Tech Research and Development Center, Kim Il Sung University, Democratic People's Republic of Korea

[2]Department of Chemistry, University of Sciences, Democratic People's Republic of Korea

Abstract

The most challenging problem of video conferencing systems is the degradation of sound quality due to various noise sources. Speech enhancement includes the reduction of background, acoustic echo cancellation, and dereverberation. A number of studies have been carried out to remove acoustic echo and background noise in video conferencing systems, and recently, DNN approaches have been applied to speech processing based on classical digital signal processing techniques, leading to great progress. We first propose a multi-input deep complex recurrent network (MIDCCRN) for noise suppression. Then, we propose a model for joint acoustic echo cancellation and background noise suppression in online voice communication systems, including video conferencing systems, using this network. The best performance of the proposed method is demonstrated by experiments with objective metrics including echo return loss enhancement (ERLE), signal-to-artifacts-ratio (SAR) and scale-invariant source-to-noise ratio (SI-SNR), mean opinion score (MOS) as a subjective metric, and AECMOS, real time factor (RTF), network size, and final score.

Keywords:

Acoustic Echo Cancellation (AEC), Background Noise Suppression (BNS), Multi-Input Deep Complex Convolution Recurrent Network (MIDCCRN), Speech Enhancement (SE)

1. INTRODUCTION

With the rapid development of information communication technology (ICT) worldwide, multimedia communication systems, including tele-video conferencing through information communication networks, are now widely used. In particular, the COVID-19 pandemic has led to the development of a number of online communication tools, which have been used to connect social members. However, the online communication systems suffer from noise, including background noise, acoustic echo, and other speaker interference noise. In this section, we analyze previous studies for background noise suppression (BNS) and acoustic echo cancellation (AEC) that are being the most challenging problem in online speech communication systems, including video conferencing systems.

1.1 REVIEW ON BNS

The BNS methods can be broadly classified as traditional digital signal processing and deep learning based methods.

1.1.1 Traditional Digital Signal Processing based BNS:

All BNS methods apply spectral suppression gain or filter to the noisy signals in a time-frequency domain. The larger the noise component reduction ability, the larger the distortion. So, in BNS, there exists a trade-off between noise attenuation and distortion. The performance of BNS methods is evaluated by the quality and intelligibility of the processed signal.

The traditional digital signal processing based BNS methods can be classified according to the number of channels (single/multi), signal processing domain (time/frequency), and type of algorithm (adaptive/non-adaptive).

In single-channel BNS mode, we usually assume that the probability distributions of speech and noise are different. Traditional digital signal processing methods have been studied, such as Wiener filter, Kalman filter, wavelet filter, spectral subtraction, cepstral mean normalization (CMN), and stochastic method [1]. The advantages of using a single microphone are its ease of installation, small size and low cost compared to the microphone array, while having the common disadvantage of distorting the desired signal. The reason is that there is no other signal to be used as a reference for noise due to the mixing of target signal and noise signal through one channel, so only the statistical characteristics of the speech signal and noise for speech enhancement (SE) is used. Moreover, single-channel mode assumes that noise is stationary in the speech interval, however the noise present in the real environment is non-stationary, with varying spectral characteristics and difficult to predict. Thus, this mode suffers from performance degradation when they are encountered in real-life nonstationary noise, especially in low signal to noise ratio (SNR) environment. In real-time applications such as speaker recognition, speech recognition, and mobile communication, BNS is very difficult due to the processing of single-channel speech.

In multi-channel BNS mode, the speech is enhanced by considering the spatial information of the signal and noise sources using a microphone array. Examples of speech recognition projects using this mode are CHIME CHALLENGE, AMI, REVERBE CHALLENGE, ASplRE, DIRHA, DARPA, etc.

Multi-channel BNS mode can be divided into two approaches. The first approach aims to separate the target speech from the multi-channel signals. Various methods such as independent component analysis (ICA), sparse component analysis (SCA), nonnegative natrix factorization (NMF), nonnegative tensor factorization (NTF), and hidden Markov model (HMM) have been studied [2]. The second approach uses beamforming techniques such as minimum variance distortionless response (MVDR). Beamforming techniques are used for speech enhancement by beamforming the target direction and removing signals incoming from the other directions [3]. The beamforming techniques include delay-sum (DS), multiple signal classification (MUSIC), MVDR, and maximum SNR (Max-SNR). Using beamforming techniques, the frequency-dependent steering matrices is obtained, and then applied to the array signal to get the BNS signal.

Multi-channel BNS mode uses the spatial information about the noise sources to overcome the drawbacks encountered in the single-channel methods and has a significant improvement under the nonstationary noise environment. While there is always a trade-off between noise attenuation and speech distortion in the single-channel mode, the multi-channel mode can significantly attenuate the noise without distorting the desired signal. These advantages of the multi-channel mode make it possible to effectively solve several problems such as source separation, source number estimation, high-quality recording, source location estimation, and time difference of arrival (TDOA) estimation, that could not be solved or were difficult in single-channel mode.

1.1.2 Deep Learning based BNS:

Recently, supervised learning using deep neural networks (DNNs) has been introduced to BNS. The DNNs estimate the time-varying gain function from features of noisy speech.

The BNS methods using DNN model include regression and mask approach [4]. The regression approach predicts a clean speech signal or its time-frequency representation from a noisy speech signal. The mask approach estimates the time-frequency mask from the noisy input signal and then produces a clean signal by element-wise multiplying the estimated mask with the input signal.

Recently, convolutional recurrent networks (CRNs) have been widely used for speech signal processing. Unlike the original convolutional recursive network with magnitude mapping, an improved neural network structure consisting of one encoder and two decoders is proposed to model the real and imaginary parts of the spectrogram that is computed by a short-time Fourier transform (STFT) from the input mixture to the clean signal [5]. The deep complex U-net (DCUNET) is a combination of U-net and deep complex networks, which is considered to be effective in BNS. DCUNET is trained to estimate complex ratio mask (CRM) and optimizes the scale-invariant source-to-noise ratio (SI-SNR) loss calculated in the time domain [6]. In [7], they introduced the deep complex recurrent networks (DCCRN), which was a combination of DCUNET and CRN with complex long-short term memory (CLSTM) layers in the bottleneck layer and complex batch normalization (CBN). This model is significantly faster and has less parameters compared to DCUNET. The DCCRN achieves great performance in Interspeech 2020 Deep Noise Suppression (DNS) [8].

1.2 REVIEW ON AEC

In an online voice communication system, including video conferencing systems, the main factor of sound quality degradation is the acoustic echo that is reflected back to the source. AEC aims to obtain a clean near-end speech signal by cancelling echo from microphone signals and minimizing the speech distortion of the near-end speaker. The AEC methods can be broadly classified as adaptive filtering, a deep neural network-based methods.

1.2.1 Adaptive filtering based AEC

There are many algorithms for adaptive filtering, such as least mean square (LMS), normalized least mean square (NLMS), linear least mean square (LLMS), recursive least squares (RLS), average adaptive filter (AAF), Affine projection algorithm (APA), etc [15]. Adaptive filtering algorithms have a slow convergence and low performance in real environments due to problems such as changes in acoustic path, background noise, and nonlinear distortion.

1.2.2 Deep Learning based AEC:

Recently, deep learning based AEC algorithms have been studied to improve AEC performance. Using DNN with sufficient training data, AEC can achieve better performance than traditional methods. Deep learning has shown great potential in AEC due to its strong nonlinear modeling ability. Usually, we use the log-magnitude spectrum of the time-frequency representation of the far-end signal and microphone signal, as input to the neural network. The output of the neural network and microphone signal phase are used to estimate the time-frequency representation of the near-end speech signal, and then the inverse time-frequency transform is used to estimate the near-end speech signal.

A number of methods have been studied using deep learning techniques, including autoencoder, restricted Boltzmann machine (RBM), convolutional neural network (CNN), recurrent neural network (RNN), LSTM-RNN, and generative adversarial network (GAN) [10], [11]. These neural networks replace adaptive filters and therefore, another digital signal processing method should be used to remove background noise or residual echo.

In online speech communication systems, there have been many approaches tor joint AEC and BNS by combining DNN with traditional signal processing methods. The linear components of echo are estimated by an adaptive filter, and the nonlinear components and background noise are modeled by DNN. With the ICASSP AEC challenge game held since 2020 in response to the global COVID-19 pandemic, DNNs combining BNS and AEC have been extensively studied [12]. A method of suppressing nonlinear echo by incorporating a LSTM structure into an adaptive filter and a method of combining frequency-domain NLMS and RNN to adaptively estimate the near-end speech features, are proposed [13]. In [14], three-stage framework of echo cancellation, residual echo plus noise suppression is proposed. The first stage cancels the linear echo by estimating time delay using the partitioned block frequency domain least mean square algorithm. The second stage suppresses residual echo and noise using a deep complex U-Net. The last stage trains a tiny U-Net to further suppress the noise. In [15], a nonlinear residual echo suppression framework based on multi-stream Conv-TasNet is proposed. Also, several methods such as dual-signal transformation LSTM, multi-task learning framework for real-time echo cancellation, multi-task learning framework to simultaneous estimate echo and near-end speech, and Wave-U-Net-based attention method are proposed.

The Complex network can simultaneously consider magnitude and phase to enhance the performance of speech signal processing [17]. Compared with real-valued networks, complex networks can achieve better performance with much smaller size parameters. A method for joint AEC and BNS by combining frequency-time-LSTM (F-T-LSTM) and a complex encoder-decoder architecture is proposed [18]. Authors demonstrated that their method outperformed [19] (four bidirectional LSTM (BLSTM) layers with 300 hidden units) and [20] (2021 AEC-challenge baseline, 2 GRU layers with 322 hidden units)) in all conditions through simulations.

Although many research groups presented AEC models in the AEC challenge contest held since 2020 year, there are still ongoing performance enhancements under various noise and echo environments [14] [20] - [27].

In this paper, we propose a model for joint AEC and BNS usimg a multi-input deep complex recurrent network (MIDCCRN).

The rest of this paper is organized as follows. Section 2 proposes a MIDCCRN, and Section 3 proposes a SE method for joint AEC and BNS tasks using the proposed network. Section 4 presents the performance evaluation through simulation. Section 5 concludes this paper.

2. MIDCCRN

In this section, we propose a MIDCCRN. As mentioned in Section 1, DCCRN are known to be suitable for noise suppression [7, 8]. In the DCCRN, the original CRN is replaced by complex CNN and the complex batch normalization layer in the encoder-decoder, and the original LSTM is replaced by CLSTM. Complex networks perform well in signal analysis by modeling the correlation between magnitude and phase with complex multiplication. The input of the Original DCCRN is only one as a noisy signal. This noisy signal is converted a complex spectrum by short-time Fourier transform (STFT), which is split into real and imaginary parts, input to the complex encoder, output to a clean signal via CLSTM, fully connected (FC) layers, complex decoder, and inverse STFT (ISTFT).

In online speech communication systems, the echo canceller is designed to input the far-end signal with the noisy microphone signal and output the clean near-end speech signal. Hence, it is possible that the DCCRN can input more than two signals, thus cancelling echo and suppressing near-end noise. Based on this idea, we propose a multi-input DCCRN called MIDCCRN by modifying the input number of the DCCRN as more than two.

Figure 1 shows the structure of the MIDCCRN. MIDCCRN consists of three blocks; input signal transform block, encoder-decoder block, and output block.

Now we will look at each block in detail.

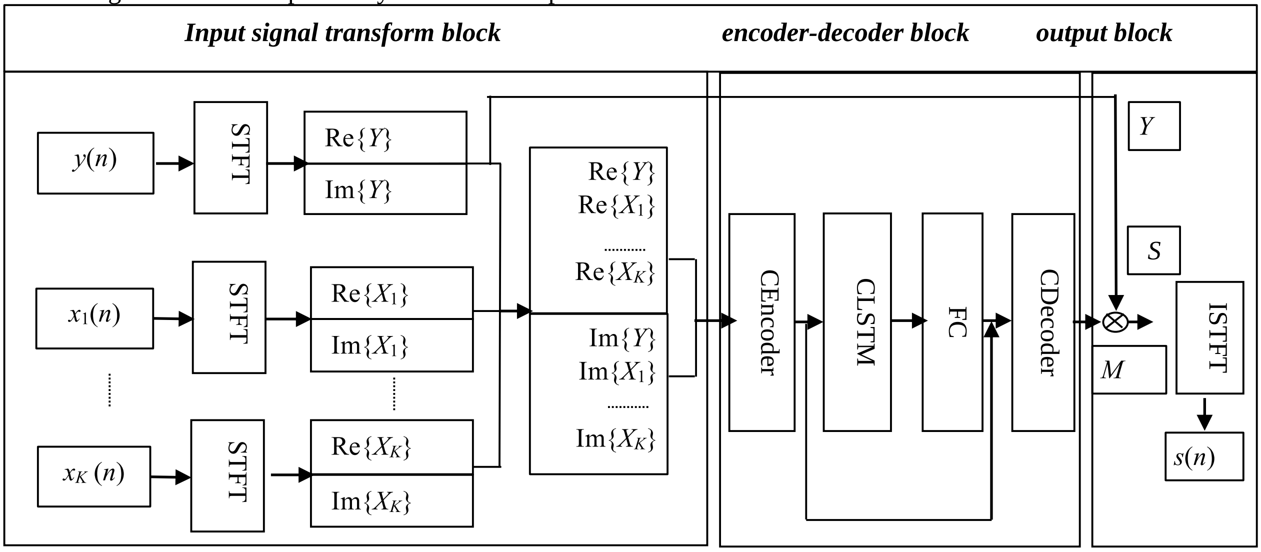

Fig.1. Structure of MIDCCRN

2.1 INPUT SIGNAL TRANSFORM BLOCK

The input of the original DCCRN was the only noisy signal $y(n)$ [7]. We change the input of the DCCRN to more than two signals. Consider $y(n)$ as noisy signal as in DCCRN. Let us consider the new K signals $x_k(n)$, (k=1,...,K) to be added, except for $y(n)$. Let $Y(t,f)$ and $X_k(t,f)$ be STFT of by $y(n)$, $x_k(n)$, respectively.

As shown in Fig.1, we rearrange the time-frequency representation of the signals by separating the real and imaginary parts of time-frequency representation of each signal, connecting the real parts to form a real sequence, connecting the imaginary parts to form a imaginary sequence.

$$R_o = [Re\{Y\}, Re\{X_1\}, \ldots, Re\{X_K\}],$$

$$I_o = [Im\{Y\}, Im\{X_1\}, \ldots, Im\{X_K\}] \qquad (1),$$

where $Re\{\cdot\}$ and $Im\{\cdot\}$ denote the real and imaginary part of the complex number, respectively. The time-frequency parameter (t,f) is omitted. For the real sequence R_o and imaginary sequence I_o, the complex spectrum $U=R_o+j\cdot I_o$, where $j=\sqrt{-1}$ is the output of the input signal transform block. Finally, multiple signals are input to the MIDCCRN, however as in the original DCCRN, one complex spectrum is input to the complex encoder.

Let us denote MIDCCRN(K) according to the number of additional incoming signals K. MIDCCRN (0) is the original DCCRN.

We set the sampling frequency of the input signal to 16 kHz. The window is Hamming window function, and the window length is set to 25 ms and the step size is set to 6.25 ms. The FFT length is set to 320. Since the spectrum corresponding to 320 frequency bins is conjugate symmetrical with respect to the center, only the previous 161 frequency spectra are selected as features. To reflect the context characteristics, we input 16 consecutive frames. Thus, the feature for one input signal is a complex matrix of size 161×16. Finally, the size of the output U of the input signal transform block of MIDCCRN(K) is T×F, where T=16(K+1) is the time dimension and F=161 is the frequency dimension.

2.2 ENCODER-DECODER BLOCK

The Encoder-Decoder block follows the form of the U-net (Fig.2).

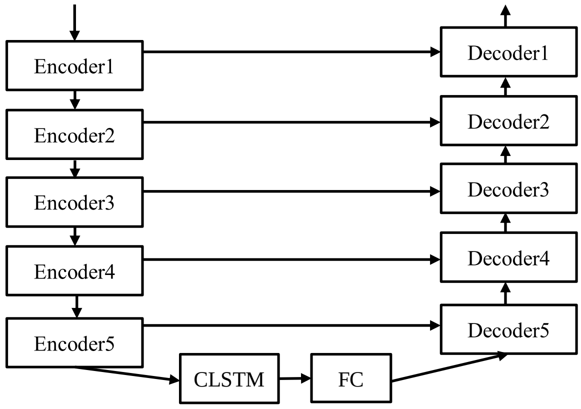

Fig.2. Encoder-Decoder block

Table.1. Hyper-parameters of Encoder-Decoder block

layer name	input size	hyper-parameters	output size
Encoder1	2 × T × 161	1 × 3, (1, 2), 16	16 ×T × 80
Encoder2	16 × T × 80	1 × 3, (1, 2), 32	32 ×T × 39
Encoder3	32 × T × 39	1 × 3, (1, 2), 64	64 ×T × 19
Encoder4	64 ×T × 19	1 × 3, (1, 2), 128	128 ×T × 9
Encoder5	128 × T × 9	1 × 3, (1, 2), 256	256 ×T × 4
reshape 1	256 × T × 4		T × 1024
CLSTM	T × 1024	1024	T × 1024
FC	T × 1024	1024	T × 1024
reshape 2	T × 1024		256 ×T × 4
Decoder5 (× 2)	512 ×T × 4	1 × 3, (1, 2), 128	128 ×T × 9
Decoder4 (× 2)	256 ×T × 9	1 × 3, (1, 2), 64	64 ×T × 19
Decoder3 (× 2)	128 ×T × 19	1 × 3, (1, 2), 32	32 ×T × 39
Decoder2 (× 2)	64 ×T × 39	1 × 3, (1, 2), 16	16 ×T × 80
Decoder1(× 2)	32 ×T × 80	1 × 3, (1, 2), 1	1 ×T × 161
concat	1× T ×161 (× 2)		2 ×T × 161

This block consists of 5 complex encoders (Encoder1, Encoder2, Encoder3, Encoder4, Encoder5), a CLSTM, a fully-connection (FC) layer, and 5 complex-decoders (Decoder1, Decoder2, Decoder3, Decoder4, and Decoder5). This block inputs the output U of the input signal conversion block of size $T \times F$ and outputs the mask Ma of size $T \times F$. The blue arrow indicates skip connection. Reshape means dimensional change and concat means that the dimension is doubled by the skip connections.

The Table.1 shows the hyper parameters of the encoder-decoder block. In the table, T=16(M+1) is the time dimension, and the frequency dimension F decreases by half from 161 to 9 in the encode, then increases again in the decoder to 161. The number of channels increases to 2, 16, 32, 64, 128, and 256 in the encoder, and then decreases again to 2 in the decoder. The input size and output size of each layer are given in the form of (number of channels × time dimension × frequency dimension). The hyper-parameters of the layers are given in the form of (kernel size, step size, and number of channels).

2.2.1 *Complex Encoder:*

The complex encoder consists of a complex 2D convolution (CConv2d), a complex batch normalization (CBN), and a parametric rectified linear unit (PReLU).

CConv2d consists of four 2D convolution (Conv2d) operations. Let W=W_r+jW_i be a complex-valued convolution filter. Here, the real-valued matrices W_r and W_i represent the real and imaginary parts of the filters, respectively. If the input complex matrix is X=X_r+jX_i, the output of the complex convolution operation is as follows;

$$Y = X \# W = (X_r * W_r - X_i * W_i) + j(X_r * W_i + X_i * W_r) \quad (2)$$

The four operations in the above equation mean the Conv2d respectively. CBN is normalized by finding the mean and variance for complex-valued training data as in classical batch normalization (BN) [17]. PReLU is a generalization of the traditional rectified linear unit (PReLU), which improves training performance by reducing overfitting risk with very small extra computational cost [28].

2.2.2 *CLSTM:*

In [7], CLSTM was implemented using four LSTM. When the complex input matrix is X=X_r+jX_i, and the complex weight matrix is W=W_r+jW_i, output Y of CLSTM is obtained as follows:

$$Y = (Y_{rr} - Y_{ii}) + j(Y_{ri} + Y_{ir}) \quad (3)$$

$$Y_{rr} = LSTM(X_r, W_r),\ Y_{ir} = LSTM(X_i, W_r),$$

$$Y_{ri} = LSTM(X_r, W_i),\ Y_{ii} = LSTM(X_i, W_i)$$

When the input real vector at time t is x_t and the real weight is W., LSTM(x_t,W) is defined as follows:

$$i_t = \sigma(W_{xi} x_t + W_{hi} h_{t-1} + b_t) \quad (4)$$

$$f_t = \sigma(W_{xf} x_t + W_{hf} h_{t-1} + b_f) \quad (5)$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \tanh(W_{xc} x_t + W_{hc} h_{t-1} + b_c) \quad (6)$$

$$o_t = \sigma(W_{xo} x_t + W_{ho} h_{t-1} + b_o) \quad (7)$$

$$h_t = o_t \circ \tanh(c_t) \quad (8)$$

where, σ denotes the sigmoid activation function, tanh the hyperbolic tangent function, ○ the element-wise multiplication, and * the real convolution operation. i_t, o_t and f_t denote the vector notation of the input, output and forget gates respectively. c_t and h_t the vector notation of the cell and hidden states respectively. For each of the gates (g=i,f,c,o), W_{xg} and W_{hg} are the input and the hidden weights. b_g is the corresponding deviation. All variables are real numbers.

We use the complex convolution operation ⊛ instead of the real convolution operation * and the complex element-wise multiplication ⊙ instead of the real element-wise multiplication ○ to implement the CLMTM operation as follows.

$$i_t = \sigma(W_{xi} \# x_t + W_{hi} \# h_{t-1} + b_t) \quad (9)$$

$$f_t = \sigma(W_{xf} \# x_t + W_{hf} \# h_{t-1} + b_f) \quad (10)$$

$$c_t = f_t \square\ c_{t-1} + i_t \square\ \tanh(W_{xc} \# x_t + W_{hc} \# h_{t-1} + b_c) \quad (11)$$

$$o_t = \sigma(W_{xo} \# x_t + W_{ho} \# h_{t-1} + b_o) \quad (12)$$

$$h_t = o_t \odot \tanh(c_t) \tag{13}$$

Here all variables are complex matrices or complex vectors. For complex input values, both sigmoid σ and hyperbolic tangent $tanh$ apply separately for real and imaginary parts. As can be seen from the formulus, the proposed method uses only one LSTM operation

Now let us compare the computational cost of our proposed method with that of the previous method. ⊛ is implemented with four * operations. Since complex multiplication is implemented with four real multiplications, the computational amount of ⊙ is four times the computational amount of ○. Therefore, the computation amount of three ⊙ in the previous method and 12 ○ in the proposed method are the same. Both the previous and the proposed methods perform * 4 × 8 = 32 times. The call numbers of σ and tanh are 12×N and 8×N respectively in the previous method, while 6×N and 4×N respectively in the proposed method. Here, *N* is the dimension of the input vector. That is, the call numbers of σ and tanh are twice that of the proposed method. As a result, we can conclude that the proposed method has half the call numbers of activation functions σ and tanh than the previous method.

2.3 OUTPUT BLOCK

The Encoder-Decoder block outputs a complex rate mask (CRM) *M*. Multiplying this complex mask *M* by the complex spectrum *Y* of the noisy signal yields the spectrum of the clean signal.

$$\hat{S} = M \cdot Y \tag{14}$$

The polar coordinate representation (M_m,M_φ) of the Cartesian representation of the complex number $M = M_r + jM_i$, ($M_r, M_i \in \mathrm{R},\ j = \sqrt{-1}$) is as follows;

$$M_m = \sqrt{M_r^2 + M_i^2},\ M_\phi = \arctan 2(M_i, M_r) \tag{15}$$

where *arctan*2 is a quarter-square tangent function with values in the interval (-π,π]. The spectrum of the clean signal is calculated as follows;

$$\hat{S} = Y_m \cdot M_m \exp\{Y_\phi + M_\phi\} \tag{16}$$

where ((Y_m,Y_φ) is the polar coordinate representation of the spectrum *Y* of the incoming noisy signal and *exp*$\{\cdot\}$ is an exponential function. The time-domain clean signal $\hat{s} = ISTFT\{\hat{S}\}$ is estimated.

3. PERFORMANCE OF AEC AND BNS

In this section, we propose a method for joint AEC and BNS in online speech communication systems using MIDCCRNs.

Let *y*(*n*) be the microphone signal and *x*(*n*) the far-end signal. Then, the microphone signal can be modeled as follows.

$$y(n) = d(n) + z(n) = d(n) + s(n) + v(n) \tag{17}$$

where *d*(*n*) is the echo signal, *s*(*n*) is the near-end speech, and *v*(*n*) is the noise. *z*(*n*)=*s*(*n*)+*v*(*n*) is the near-end signal. If *H*(·) denotes a echo path function, the echo signal can be expressed as *d*(*n*)=*H*(*x*(*n*)).

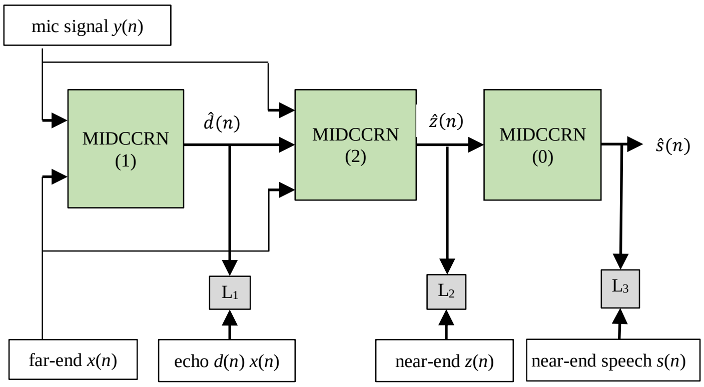

Fig.3. Framework for AEC and BNS

The framework for AEC and BNS using MIDCCRNs is shown in Fig.3. As shown in the figure, the proposed network consists of three blocks. The first is the echo estimation block, the second is the AEC block, and the third is the BNS block. L1, L2, and L3 are loss functions for echo signal, near-end signal, and near-end signals, respectively. $\hat{s}(n)$ denotes the estimate of *s*(*n*).

The first block inputs the microphone signal *y*(*n*) and the far-end signal *x*(*n*) into MDCCRN(1) and estimates the echo signal $\hat{d}(n)$. The second block inputs the microphone signal *y*(*n*), the far-end signal *x*(*n*), and the estimated echo signal $\hat{d}(n)$ into MDCCRN(2), removes the echo and estimates the near-end signal $\hat{z}(n)$. The third block inputs the estimated near-end signal $\hat{z}(n)$ into MDCCRN(0), removes noise and estimates the target near-end speech signal $\hat{s}(n)$.

In the previous work, we usually use the weighted-signal-to-distortion ration (wSDR) as a loss function. Let $y, s, \hat{s}$ be noisy signal (mixed signal), clean signal, and estimated signal in the time-domain, respectively.

$$wSDR(y, s, \hat{s}) = -\alpha\phi(s, \hat{s}) - (1-\alpha)\phi(n, \hat{n}) \tag{18}$$

where $\varphi(v,w)$ represents the cross-correlation between two vectors *v*,*w*, i.e., the cosine of the angle between the two vectors.

$$\phi(v, w) = \frac{v^T w}{\|v\|\|w\|} \tag{19}$$

where ||·|| is the Euclidean norm. And $\alpha = \frac{\|s\|^2}{\|y\|^2}$ is the energy ratio of the clean signal to the noisy signal. Also, $n = y - s$ and $\hat{n} = y - \hat{s}$ are the noise and the estimated noise, respectively.

The wSDR calculates the loss function for the total length of the signal. Applying wSDR to segment the whole signal into short time intervals can better reflect the local characteristics of the signal. So, we use the short-time-wSDR (STwSDR) as a loss function.

The sequence of signals segmented by *N* short time intervals of $y,\ s, \hat{s}$ can be written as $y = \{y_0, \ldots, y_N\}$, $s = \{s_0, \ldots, s_N\}$, $\hat{s} = \{\hat{s}_0, \ldots, \hat{s}_N\}$. Then STwSD is defined as follows;

$$STwSDR(y,s,\hat{s}) = \frac{\sum_{i=0}^{N-1} wSDR(y_i, s_i, \hat{s}_i)}{N} \qquad (20)$$

The number of segments N is usually set to 10. In Fig.3, all loss functions L_1, L_2 and L_3 are STwSDR. We train the model using multi-task learning. The final loss function is as follows;

$$\begin{aligned} Loss = &\beta_1 \cdot STwSDR_1(y(n), d(n), \hat{d}(n)) \\ &+ \beta_2 \cdot STwSDR_2(y(n), z(n), \hat{z}(n)) \\ &+ \beta_3 \cdot STwSDR_3(y(n), s(n), \hat{s}(n)) \end{aligned} \qquad (21)$$

where, β_1, β_2 and β_3 are respectively set to values between 0 and 1 as constants determining the weight of each loss term.

$$\beta_1, \beta_2, \beta_3 \in [0,1] \qquad (22)$$

The network proposed in this section can effectively remove echo and noise.

4. EXPERIMENTS AND DISCUSSION

In this section, we present a performance evaluation of our proposed method for BNS and AEC through experiments

4.1 EXPERIMENTAL DESIGN

To evaluate the performance of various AECs, we describe dataset, evaluation metrics and comparison methods.

Two datasets are presented in the AEC challenge [20,21]. One is synthetic data and the other is real recordings.

In the synthetic dataset, there are 10,000 synthesized data of 10s length, including single talk, double talk, far-end noise, near-end noise, and nonlinear distortions. The dataset is randomly divided into 80% for training, 10% for validation and 10% for testing.

The length of each signal is 10s.is resampled at 16k Hz. The window length is set as 25ms, the hop size is set as 6.25ms and the FFT length is set as 512. The optimizer is selected as Adam. The initial learning rate is set to 0.001.

As objective evaluation metrics for acoustic echo cancellation, echo return loss enhancement (ERLE), signal-to-artifact-ratio (SAR), and scale-invariant source-to-noise ratio (SI-SNR) are used. The mean opinion score (MOS) is used as a subjective metric. We use an objective perceptual quality assessment scale called AECMOS [29].

First, we determine the parameters β_1, β_2, and β_3 in the proposed method through simulations. Next, we compare the proposed MIDCCRN with 2021 AEC challenge baseline [30], F-T-LSTM based complex network [18], and best methods in the 2022 acoustic echo cancellation challenge contest [23-27].

Final score for the AEC challenge performance evaluation is determined using the weighted average of the four subjective scores (M_1, M_2, M_3, M_4) and *Wacc*.

$$M = \sum_{i=1}^{4} \frac{1}{4} \alpha_i (M_i - 1) + \alpha_5 W_{acc} \qquad (23)$$

where M_1 is far end single talk, M_2 is near end single talk, M_3 is double talk echo, and M_4 is double talk other. The word accuracy rate (*WAcc*) is used as a metric in the speech recognition and AEC challenge; *WAcc* = 1-*WER*, where *WER* denotes word error rate. We all the weights are set equally α_i=1/5. We also use real-time factor (RTF) and network size for performance evaluation.

4.2 EXPERIMENTAL RESULTS

4.2.1 Parameter Determination in the Final Loss Function:

First, let us determine experimentally the best value of the parameters β_1, β_2, and β_3 in the final loss function formula (21) of our proposed method. The final BNS block MIDCCRN (0) must be and plays the most important role. So the loss function weight of the last BNS block is always set to 1 and the other parameters are changed. The Table.2 shows the results of SI-SNR comparison for different values of the parameters β_1, β_2, and β_3 of the final loss function for the proposed method. In the table, C is the clean signal, N-F is the noisy far-end signal, N-N is the noisy near-end signal, and N-FN is the noisy signal both the far-end signal and the near-end speech signal. As can be seen from the table, if

$$\beta_1 = 0.2, \qquad \beta_2 = 0.5, \qquad \beta_3 = 1, \qquad (24)$$

SI-SNR is the best. In the experiments with the proposed method, the parameters are set as Eq.(23).

Table.2. SI-SNR evaluation according to parameters β_1, β_2, β_3

β_1	β_2	β_3	*C*	*N-F*	*N-N*	*N-FN*
0.2	0.5	1	13.89	10.52	6.35	6.02
0.2	1	1	13.60	10.32	6.12	5.75
1	1	1	13.58	10.14	5.94	5.34
0.2	0	1	13.37	10.25	5.99	5.68
0	1	1	13.41	10.21	5.94	5.51
0	0	1	12.91	9.92	6.01	5.54
1	0	1	12.47	9.36	5.34	4.81

4.2.2 Objective Evaluation:

The Table.3 shows the results of the comparative evaluation of SAR in present of only near-end signal and ERLE in present of far-end signal alone without near-end signal. As can be seen from the table, the highest SAR and ERLE of the proposed method are found.

Table.3. SAR and ERLE evaluation [dB]

Method	SAR	ERLE
[30]	14.70	17.74
[18]	17.56	23.98
Proposed	**17.85**	**24.62**

We compare the SI-SNR of each method through experiments. Table 4 shows the comparison of the SI-SNR [dB]. In the method column, 0 represents the case where no processing is done. As can be seen from the table, the proposed method outperforms all other methods in SI-SNR performance under all conditions.

Table 4. SI-SNR evaluation under several conditions [dB]

Method	C	N-F	N-N	N-FN
0	7.60	2.97	-3.65	-3.82

[30]	12.82	8.05	1.89	1.62
[18]	13.51	10.35	6.20	5.78
Proposed	13.89	10.52	6.35	6.02

4.2.3 Subjective Evaluation and Performance Evaluation:

Employees of High-Tech Research and Development Center, Kim Il Sung University attended MOS evaluation. We prepared 20 far-end-microphone signal pairs $\{x_j,y_j\}$ (x_j is the j^{th} far-end signal and y_j is the j^{th} microphone signal) extracted during the video conference. The length of each signal is 10s. The Table.5 shows the MOS of each method about these 20 signal pairs. As can be seen from the table, the MOS of the proposed method is the highest.

Table 5. MOS evaluation

Method	**MOS**
[30]	3.89
[18]	4.14
Proposed	**4.15**

The Table.6 shows the results of comparing AECMOS, RTF, network size and final score *M* of the eight methods. The final score *M* of the proposed method is the highest, and compared to the same final score [23], the AECMOS value is lower, and the network size is larger, but the RTF is smaller.

Table.6. AECMOS, RTF, network size, final score M evaluation

No	**Method**	**AECMOS**	**RTF**	**network size[106]**	***M***
1	[30]	3.65	0.283	5.08	0.75
2	[18]	3.95	0.410	12.8	0.79
3	[23]	**4.00**	0.60	**1.5**	**0.85**
4	[24]	3.97	0.10	17.4	0.84
5	[25]	3.92	0.20	4.8	0.82
6	[26]	3.85	0.30	55.5	0.80
7	[27]	3.72	**0.02**	4.3	0.78
8	Proposed	3.97	0.302	4.28	**0.85**

5. CONCLUSIONS

The contributions of this paper are as follows.

- First, the MIDCCRN is proposed. Unlike the DCCRN with only one input, the MIDCCRN can increase the number of input signals as needed. And in CLSTM implementation, we reduced the number of calls to half of the activation function by using complex convolution operation $\circledast$ instead of real convolution operation *.
- Second, a metwork for joint AEC and BNS using MIDCCRNs is proposed. This network consists of echo estimation block, AEC block and BNS block using MIDCCRNs, and uses STwSDRs as losse functions.
- Third, the performance evaluation of the proposed AEC model is carried out in comparison with previous methods, and the proposed method is demonstrated to be the best.

In the future, we will study neural networks that simultaneously remove echo, interfering signals, background noises, and reverberation under various acoustic environments.

ACKNOWLEDGMENT

This work was supported by the Institute of Information Technology, High-Tech Research and Development Center, Kim Il Sung University. The authors would like to take the opportunity to express their hearty gratitude to all those who made a contribution to the completion of this article. Especially authors would like to thank employees of High-Tech Research and Development Center to attend our MOS evaluation.

REFERENCES

[1] Eberhard. Hansler and Gerhard Schmidt, "*Speech and Audio Processing in Adverse Environments*", 2008.

[2] Katerina Zmolıkova, Marc Delcroix, Keisuke Kinoshita, Takuya Higuchi, Atsunori Ogawa and Tomohiro Nakatani, "Speaker-Aware Neural Network based Beamformer for Speaker Extraction in Speech Mixtures", *Interspeech*, pp. 2655-2659, 2017.

[3] Jacob Benesty, Israel Cohen and Jingdong Chen, "*Fundamentals of Signal Enhancement and Array Signal Processing*", 2018.

[4] S. Donald Williamson, Yuxuan Wang and DeLiang Wang, "Complex Ratio Masking for Monaural Speech Separation", *IEEE Transactions on Audio, Speech and Language Processing*, Vol. 24, No. 3, pp. 483-492, 2016.

[5] Ke Tan and DeLiang Wang, "Complex Spectral Mapping with a Convolutional Recurrent Network for Monaural Speech Enhancement", *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, pp. 6865-6869, 2019.

[6] Yi Luo and Nima Mesgarani, "Conv-Tasnet: Surpassing Ideal Time-Frequency Magnitude Masking for Speech Separation", *IEEE Transactions on Audio, Speech and Language Processing*, Vol. 27, No. 8, pp. 1256-1266, 2019.

[7] Yanxin Hu, Yun Liu, Shubo Lv, Mengtao Xing, Shimin Zhang, Yihui Fu, Jian Wu, Bihong Zhang and Lei Xie, "DCCRN: Deep Complex Convolution Recurrent Network for Phase-Aware Speech Enhancement", *Proceedings of International Conference on Audio and Speech Processing*, pp. 1-6, 2020.

[8] C.K. Reddy, V. Gopal, R. Cutler, E. Beyrami, R. Cheng, H. Dubey, S. Matusevych, R. Aichner, A. Aazami, S. Braun, P. Rana, S. Srinivasan and J. Gehrke, "The Interspeech 2020 Deep Noise Suppression Challenge: Datasets, Subjective Testing Framework and Challenge Results", *Interspeech*, pp. 2492-2496, 2020.

[9] Eberhard Hansler and Gerhard Schmidt, "*Acoustic Echo and Noise Control: A Practical Approach*", 2004.

[10] Yao-Cheng Tsai, Kai-Wen Liang and Pao-Chi Chang, "Acoustic Echo Cancellation based on Recurrent Neural Network", *Proceedings of International Conference on Asia-Pacific Signal and Information Processing Association Annual Summit*, pp. 88-91, 2020.

[11] Yi Zhang, Chengyun Deng, Shiqian Ma, Yongtao Sha, Hui Song and Xiangang Li, "Generative Adversarial Network based Acoustic Echo Cancellation", *Interspeech*, pp. 3945-3949, 2020.

[12] Ross Cutler, Ando Saabas, Tanel Parnamaa, Markus Loide, Sten Sootla, Marju Purin, Hannes Gamper, Sebastian Braun, Karsten Sorensen, Robert Aichner and Sriram Srinivasan, "Interspeech 2021 Acoustic Echo Cancellation Challenge", *Interspeech*, pp. 4748-4752, 2021.

[13] L. Ma, H. Hua and Z. Pei, "Acoustic Echo Cancellation by Combining Adaptive Digital Fillter and Recurrent Neural Network", *Proceedings of International Conference on Audio and Speech Processing*, pp. 1-8, 2020.

[14] R. Peng, "Icassp 2021 Acoustic Echo Cancellation Challenge: Integrated Adaptive Echo Cancellation with Time Alignment and Deep Learning-based Residual Echo Plus Noise Suppression", *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, pp. 146-150, 2021.

[15] H. Chen, "Nonlinear Residual Echo Suppression based on Multi-Stream Conv-Tasnet", *Proceedings of International Conference on Audio and Speech Processing*, pp. 1-7, 2020.

[16] D. Stoller, "Wave-U-Net: A Multi-Scale Neural Network for End-to-End Audio Source Separation", *Proceedings of International Conference on Audio and Speech Processing*, pp. 1-7, 2018.

[17] C. Trabelsi, "Deep Complex Networks", *Proceedings of International Conference on Audio and Speech Processing*, 2017.

[18] S. Zhang, "F-T-LSTM based Complex Network for Joint Acoustic Echo Cancellation and Speech Enhancement", *Proceedings of International Conference on Audio and Speech Processing*, pp. 1-8, 2021.

[19] H. Zhang, "Deep Learning for Acoustic Echo Cancellation in Noisy and Double-Talk Scenarios", *Interspeech*, Vol. 161, No. 2, pp. 1-6, 2018.

[20] R. Cutler, A. Saabas, T. Parnamaa, M. Loide, S. Sootla, M. Purin, H. Gamper, S. Braun, K. Sorensen, R. Aichner and S. Srinivasan, "Interspeech 2021 Acoustic Echo Cancellation Challenge: Datasets and Testing Framework", *Interspeech*, pp. 1-13, 2021.

[21] R. Cutler, "ICASSP 2022 Acoustic Echo Cancellation Challenge", *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, pp. 1-10, 2022.

[22] "AEC Challenge", Available at https://github.com/microsoft/AEC-Challenge/tree/main/baseline/ icassp2022, Accessed in 2022.

[23] G. Zhang, L. Yu, C. Wang and J. Wei, "Multi-Scale Temporal Frequency Convolutional Network with Axial Attention for Speech Enhancement", *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, pp. 1-11, 2022

[24] H. Zhao, N. Li, R. Han, L. Chen, X. Zheng, C. Zhang, L. Guo and B. Yu, "A Deep Hierarchical Fusion Network for Fullband Acoustic Echo Cancellation", *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, pp. 1-6, 2022

[25] S. Zhang, Z. Wang, J. Sun, Y. Fu, B. Tian, Q. Fu and L. Xie, "Multi-Task Deep Residual Echo Suppression with Echo-Aware Loss", *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, pp. 1-9, 2022.

[26] X. Sun, C. Cao, Q. Li, L. Wang and F. Xiang, "Explore Relative and Context Information with Transformer for Joint Acoustic Echo Cancellation and Speech Enhancement", *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, pp. 1-5, 2022.

[27] F. Cui, L. Guo, W. Li, P. Gao and Y. Wang, "Multi-Scale Refinement Network based Acoustic Echo Cancellation", *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, pp. 1-12, 2022.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on Imagenet Classification", *Proceedings of International Conference on Computer Vision*, pp. 1026-1034, 2015.

[29] Marju Purin, Sten Sootla, Mateja Sponza, Ando Saabas and Ross Cutler, "AECMOS: A Speech Quality Assessment Metric for Echo Impairment", *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, pp. 1-7, 2022.

[30] Yangyang Xia, Sebastian Braun, K.A. Chandan Reddy, Harishchandra Dubey, Ross Cutler and Ivan Tashe,"Weighted Speech Distortion Losses for Neural-Network-based Real-Time Speech Enhancement", *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, pp. 871-875, 2020.