Deep Learning Based Real-Time Speech Enhancement for Dual-Microphone Mobile Phones

Ke Tan, Xueliang Zhang, and DeLiang Wang, *Fellow, IEEE*

***Abstract*—In mobile speech communication, speech signals can be severely corrupted by background noise when the far-end talker is in a noisy acoustic environment. To suppress background noise, speech enhancement systems are typically integrated into mobile phones, in which one or more microphones are deployed. In this study, we propose a novel deep learning based approach to real-time speech enhancement for dual-microphone mobile phones. The proposed approach employs a new densely-connected convolutional recurrent network to perform dual-channel complex spectral mapping. We utilize a structured pruning technique to compress the model without significantly degrading the enhancement performance, which yields a low-latency and memory-efficient enhancement system for real-time processing. Experimental results suggest that the proposed approach consistently outperforms an earlier approach to dual-channel speech enhancement for mobile phone communication, as well as a deep learning based beamformer.**

***Index Terms*—Real-time speech enhancement, complex spectral mapping, densely-connected convolutional recurrent network, dual-microphone mobile phones.**

I. Introduction

In MOBILE communication, speech signals are corrupted by background noise when the far-end talker is in a noisy acoustic environment. In order to attenuate background noise, speech enhancement algorithms have been integrated into most mobile phones, where one or more microphones are deployed. More microphones typically yield better enhancement results. However, the number of the microphones is subject to practical limitations such as the size, power consumption, and expense of the array. Therefore, a dual-microphone configuration is a common choice. In a typical dual-microphone setup, a primary microphone is placed on the bottom of a mobile phone and a secondary microphone on the top, as illustrated in Fig. 1.

In the past decade, a variety of algorithms have been developed for dual-channel speech enhancement. Yousefian *et al.* [43] developed a Wiener filter that exploits the power level difference (PLD) between the signals received by two microphones. The experimental results show that their approach improves speech quality. Jeub *et al.* [16] designed a PLD-based noise estimator, which uses the normalized inter-channel PLD as speech presence probability (SPP). The estimated noise spectrum is used to compute a spectral gain, which is subsequently applied to the noisy spectrum to derive the enhanced spectrum. The results show that their approach outperforms the approach in [43], in terms of objective intelligibility. A similar method was proposed in [44], in which the power level ratio of the dual-channel signals is used to calculate a spectral gain. This method produces comparable results to the PLD-based method in [16], while more efficient computationally. More recently, Fu *et al.* [8] developed a SPP-based noise correlation matrix estimator, where the inter-channel posteriori signal-to-noise ratio difference (PSNRD) is utilized to estimate SPP. The estimated noise correlation matrix is subsequently used to derive a minimum variance distortionless response (MVDR) spatial filter for noise reduction. Their results show that the PSNRD method is more robust than the PLD method in [16] against different sensitivities of two microphones. Other related studies include [22], [19] and [3].

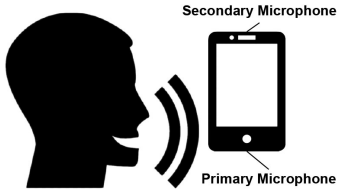


Fig. 1. Illustration of a dual-microphone configuration of mobile phones.

Speech enhancement has been recently formulated as supervised learning, inspired by the concept of time-frequency (T-F) masking in computational auditory scene analysis (CASA) [36]. Thanks to the use of deep learning, the performance of supervised speech enhancement has been dramatically improved in the past decade [37]. Compared to the dual-channel setup, speech enhancement for mobile phones needs to consider short speaker-microphone distances and head shadow effects. To our knowledge, the first deep learning based enhancement method for dual-microphone mobile phones was designed by López-Espejo *et al.* [18], where a deep neural network (DNN) is trained to produce a binary mask from the log-mel features of the noisy array signals. A truncated-Gaussian based imputation algorithm is used to produce the enhanced spectrum from the estimated mask. In a subsequent study [20], they trained a DNN

Manuscript received October 25, 2020; revised March 10, 2021 and May 2, 2021; accepted May 14, 2021. Date of publication May 20, 2021; date of current version June 4, 2021. This work was supported in part by an NIDCD under Grant R01 DC012048, and in part by Ohio Supercomputer Center. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yu Tsao. *(Corresponding author: Ke Tan.)*

Ke Tan is with the Department of Computer Science and Engineering, The Ohio State University, Columbus, OH 43210-1277, USA (e-mail: tan.650@osu.edu).

Xueliang Zhang is with the Department of Computer Science, Inner Mongolia University, Hohhot 010021, China (e-mail: cszxl@imu.edu.cn).

DeLiang Wang is with the Department of Computer Science and Engineering and the Center for Cognitive and Brain Sciences, The Ohio State University, Columbus, OH 43210-1277 USA (e-mail: dwang@cse.ohio-state.edu).

Digital Object Identifier 10.1109/TASLP.2021.3082318

to estimate the noise spectrum from the log-mel features of dual-channel noisy speech. The noise estimate, along with the primary-channel noisy signal, is used to produce the primary-channel enhanced spectrum by a vector Taylor series feature compensation method. The enhanced spectrum is subsequently passed into a speech recognizer for evaluation. Their results show that the DNN-based approach yields significantly higher word accuracy than several conventional approaches.

Real-time speech enhancement is needed for mobile communication, and it poses several requirements on model design. First, the model should use no or few future time frames. For example, causal DNNs for speech enhancement have been recently developed [23], [31]. Second, the model should not have a high computational cost for the sake of processing latency and power consumption. Third, memory consumption should fit the given capacity of mobile phones. It should be noted that memory consumption has two main aspects, i.e. to store trainable parameters and intermediate results (e.g. the activations from lower DNN layers).

In a preliminary study [33], we recently proposed a convolutional recurrent network (CRN) for real-time dual-microphone speech enhancement, motivated by an earlier study on CRN [31]. The proposed method produces a phase-sensitive mask (PSM) [6], [39] from magnitude-domain intra- and inter-channel features. The present study extends the CRN-based method to improve its robustness. The present work differs from the preliminary study in the following main aspects. First, we extend the CRN architecture into a densely-connected CRN (DC-CRN). Specifically, each convolutional or deconvolutional layer is replaced by a densely-connected block. In addition, each skip connection between the encoder and the decoder is replaced by a densely-connected block. Second, we train the DC-CRN to learn a mapping from the real and imaginary spectrograms of the dual-channel noisy mixture to those of the primary-channel clean speech signal, inspired by recent advances in complex-domain speech enhancement [7], [32], [42]. Third, we propose a structured pruning technique to compress the DC-CRN, which significantly reduces the model size without significantly affecting the enhancement performance. Fourth, we simulate array signals by spatializing speech and noise signals by covering a reasonable range of source-array distances and including the head shadow effect. Such a data simulation method accounts for various ways of holding a mobile phone, more robust than using close-talk inter-channel relative transfer functions [33].

The rest of this paper is organized as follows. In Section II, we formulate dual-channel speech enhancement for mobile phones. In Section III, we describe our proposed approach in detail. Experimental setup is provided in Section IV. In Section V, we present experimental and comparison results. Section VI concludes this paper.

II. Dual-Channel Speech Enhancement for Mobile Phone Communication

Given a dual-channel signal recorded in a noisy and reverberant environment, the signal model can be formulated as

$$y^{(q)}[k] = s[k] * h_s^{(q)}[k] + \sum_j n_j[k] * h_{n_j}^{(q)}[k], \tag{1}$$

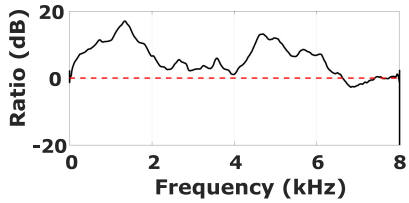
where s and n_j denote the speech source and the j-th noise source, respectively, and h_s and h_{n_j} the room impulse responses (RIRs) corresponding to the speech source and the j-th noise source, respectively. Symbol $*$ represents the convolution operation, k the time sample index, and $q \in \{1, 2\}$ the microphone index. In the short-time Fourier transform (STFT) domain, the signal model can be written as

$$\begin{aligned} \mathbf{Y}(m,f) &= \mathbf{c}(f)S_1(m,f) + \mathbf{R}(m,f) + \mathbf{N}(m,f) \\ &= [Y_1(m,f), Y_2(m,f)]^T \in \mathbb{C}^{2\times 1}, \end{aligned} \tag{2}$$

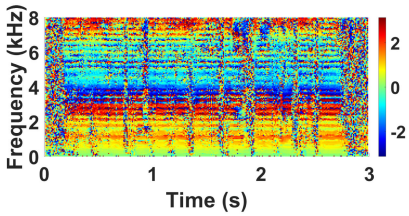
where $S_1 \in \mathbb{C}$ is the STFT of the target speech signal captured by the primary microphone (microphone 1 in this case), $\mathbf{c}(f) = [1, c(f)]^T \in \mathbb{C}^{2\times 1}$ is the relative transfer function between the two microphones, and $\mathbf{R}$ and $\mathbf{N}$ denote the STFTs of speech reverberation and reverberant noise, respectively. Y_1 and Y_2 are the STFTs of $y^{(1)}$ and $y^{(2)}$, respectively. Symbols m and f index the time frame and the frequency bin, respectively. In this study, we aim to extract the target speech signal captured by the primary microphone, i.e. $s_1 = \mathcal{F}^{-1}\{S_1\}$, where $\mathcal{F}^{-1}$ represents the inverse STFT (iSTFT). We focus on noise reduction and assume that reverberation energy is relatively weak, which is reasonable with relatively short speaker-phone distances in mobile communication.

There are broadly two kinds of mobile phone use scenarios: hand-held and hands-free. In a hand-held scenario, the primary microphone is typically close to the talker's mouth and the secondary microphone close to the ear. In a hands-free scenario, the mobile phone can be placed at some distance, e.g. on a desk in front of the talker. Note that the terms *hand-held* and *hands-free* in this paper should not be interpreted literally, but are meant to differentiate the locations of the two microphones relative to the head.

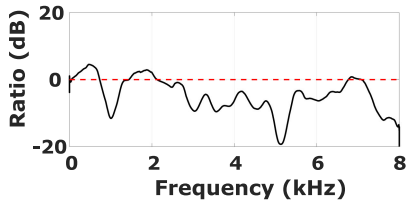
In the hand-held scenario, the sound level of the speech signal coming from the talker's mouth is reduced by the head obstruction, prior to reaching the secondary microphone near the ear. This head shadow results in a difference between the received speech levels at the two microphones. An example of the power spectral density (PSD) ratio of the primary channel to the secondary channel is shown in Fig. 2(a), where the dual-channel signals are recorded in a hand-held setup without background noise. It can be observed that the primary signal has a larger PSD than the secondary signal in almost all frequency bands. In the hands-free scenario without the head shadow effect, as illustrated in Fig. 2(c), the speech level at the primary channel is not always higher than that at the secondary channel. In both scenarios, the inter-channel intensity difference (IID) is a useful spatial cue for speech enhancement, corresponding to the magnitude difference between Y_1 and Y_2 (see Eq. (2)), which is leveraged by most studies for dual-channel speech enhancement. Another useful spatial cue is the inter-channel phase difference (IPD) or inter-channel time difference (ITD), which is highly correlated with the direction of arrival with respect to the dual-channel array. Specifically, the IPD can be calculated as $\theta_{y_1} - \theta_{y_2}$, where θ_{y_1} and θ_{y_2} are the phases of Y_1 and Y_2, respectively. Figs. 2(b) and 2(d) show the IPDs, wrapped into $[-\pi, \pi]$, for the corresponding hand-held and hands-free scenarios, respectively.



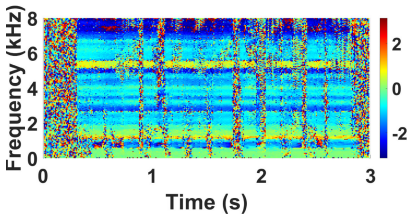
(a) Inter-channel PSD Ratio



(b) Inter-channel Phase Difference



(c) Inter-channel PSD Ratio



(d) Inter-channel Phase Difference

Fig. 2. (Color Online). Examples of inter-channel PSD ratio and phase difference in (a)-(b) hand-held and (c)-(d) hands-free scenarios.

Both IID and IPD (or ITD) can be implicitly exploited by learning a multi-channel complex spectral mapping [41], where the IID and the IPD are encoded in the dual-channel complex spectrogram of the noisy mixture. In contrast to conventional beamforming that typically exploits second-order statistics of multiple channels [34], such an approach has the potential to extract all discriminative cues in dual-channel complex-domain inputs through deep learning. In addition, complex spectral mapping simultaneously enhances magnitude and phase responses of target speech [32], which is advantageous over magnitude-domain approaches that ignore phase.

III. Model Description

In this section, we first introduce our proposed densely-connected convolutional recurrent network for dual-channel complex spectral mapping, and then elaborate the network configurations for a noncausal enhancement system with a large model size, as well as a causal and lightweight version for real-time processing. We also propose a structured pruning technique

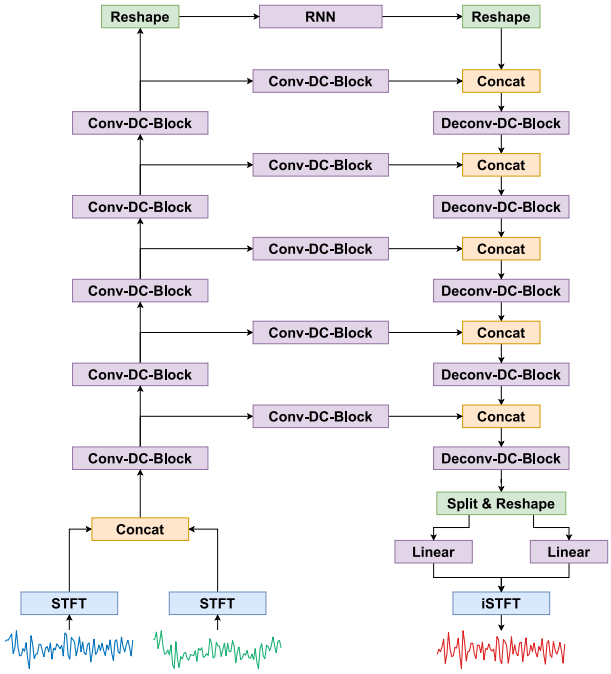


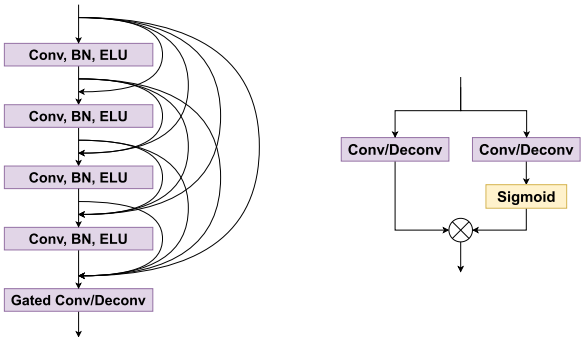
Fig. 3. (Color Online). Diagram of the DC-CRN for dual-channel complex spectral mapping.

to compress the DC-CRN without significantly sacrificing the enhancement performance.

A. Densely-Connected Convolutional Recurrent Network

In [32], we have recently developed a gated convolutional recurrent network (GCRN) to perform complex spectral mapping for monaural speech enhancement, which substantially outperforms an earlier convolutional neural network (CNN) that learns complex spectral mapping [7]. The GCRN has an encoder-decoder architecture with skip connections between the encoder and the decoder. A two-layer long short-term memory (LSTM) is inserted between the encoder and decoder to aggregate temporal contexts. The encoder is a stack of gated convolutional layers, and the decoder a stack of gated deconvolutional layers. Such an architecture benefits from both the feature extraction capability of the convolutional autoencoder and the sequential modeling capability of the LSTM, and can effectively capture the local and global spectral structure in a spectrogram.

This study develops the CRN architecture for dual-channel complex spectral mapping. The diagram of the proposed approach is shown in Fig. 3. The input complex spectrograms are computed by applying STFT to the time-domain waveforms of the dual-channel mixtures. We concatenate the real and imaginary components of the dual-channel spectrograms [42], which amount to a 3-dimensional (3-D) representation with four channels. Subsequently, the 3-D representation is passed into a convolutional encoder, which comprises a stack of five convolutional densely-connected (DC) blocks. The 3-D representation learned by the encoder is reshaped to a sequence of 1-D features, which is then modeled by a recurrent neural



(a) Densely-connected Block (b) Gated Convolution/Deconvolution

Fig. 4. (Color Online). Diagrams of the densely-connected block (a) and the gated convolution/deconvolution (b). The symbol $\bigotimes$ represents the element-wise multiplication.

network (RNN). We reshape the output of the RNN back to a 3-D representation and subsequently feed it into a decoder, i.e. a stack of five deconvolutional DC blocks. The output of the last block is split into two equal-sized 3-D representations along the channel dimension, one for the real spectrum estimation and the other for the imaginary spectrum estimation. These two 3-D representations are individually reshaped to a sequence of 1-D features, and then passed through a linear projection layer to produce estimates of the real and imaginary components of the clean spectrogram, respectively. We apply the iSTFT to the estimated real and imaginary spectrograms to resynthesize the time-domain waveform of enhanced speech for the primary channel.

Unlike the skip connections that directly bypass the output of each encoder layer to the corresponding decoder layer in [33] and [32], a convolutional DC block is employed to process the features learned by each DC block in the encoder, prior to concatenating them with the output of the corresponding DC block in the decoder. Such a design is inspired by U-Net++ [47] for image segmentation, which uses DC blocks to bridge the semantic gap between the feature maps of the encoder and the decoder prior to fusion. The introduction of DC block based skip pathways can enrich the feature maps from the encoder, which would help to increase the similarity between the feature maps from the encoder and the decoder and thus improves their fusion.

As shown in Fig. 4(a), we propose a dense connectivity pattern in each DC block to improve the information flow between layers, i.e. introducing direct connections from any layer to all subsequent layers. In other words, each layer receives the outputs of all preceding layers:

$$\mathbf{z}_l = \mathcal{H}_l\left([\mathbf{z}_0, \ldots, \mathbf{z}_{l-1}]\right), l = 1, \ldots, L, \tag{3}$$

where $\mathcal{H}_l$ denotes the mapping function defined by the l-th layer in the DC block, and $[\cdot, \ldots, \cdot]$ the concatenation operation. The output of the l-th layer is represented by $\mathbf{z}_l$, and $\mathbf{z}_0$ is the input to the DC block. By encouraging feature reuse, the dense connections exploit the differences learned by different preceding layers. In this study, we set L to 5. Specifically, each of the first four layers in a DC block consists of a 2-D convolutional layer successively followed by batch normalization [15] and exponential linear activation function [4]. The last layer in the DC block is a gated convolutional or deconvolutional layer as illustrated in Fig. 4(b), which incorporates the gated linear units developed in [5]. Note that "Conv-DC-Block" in Fig. 3 performs gated convolution in the last layer, and "Deconv-DC-Block" gated deconvolution in the last layer.

It should be noted that using an RNN for sequential modeling is typically more memory-efficient than time-dilated convolutions [24], [30] or temporal attention [17], particularly with strict memory limitation. The use of time-dilated convolutions necessitates storing intermediate activations for many past time steps in the receptive fields of all layers. Similarly, it is necessary to store intermediate activations from many past time steps in order to perform temporal attention. In contrast, an RNN only needs the input at the current time step and the hidden state from the last time step to calculate the output at the current step. Therefore, the RNN would demand far less working memory than a comparably sized DNN based on time-dilated convolutions or temporal attention, even if the RNN may have more trainable parameters than the DNN.

B. Network Configurations

1) Noncausal DC-CRN: In order to systematically examine the proposed architecture, we first configure the DC-CRN into a noncausal system with a reasonably large model size. In each convolutional or deconvolutional DC block, each of the first four layers has 8 output channels with a kernel size of 1×3 (time $\times$ frequency), where a zero-padding of size 1 is applied to each side of the feature maps along the frequency dimension. For the DC blocks in the encoder and the decoder, the last layer in each of them has a kernel size of 1×4, where a stride of 2 and a zero-padding of 1 (for each side) is applied along the frequency dimension. Note that the kernel size is set to 1×4 rather than 1×3 in order to alleviate the checkerboard artifacts [1], which arise when the kernel size of a strided deconvolution is not divisible by the stride. Moreover, the DC blocks in the encoder have 16, 32, 64, 128 and 256 output channels successively, and those in the decoder have 256, 128, 64, 32 and 16 output channels successively. The convolutional DC blocks in the skip pathways have the same hyperparameters as those in the encoder, except that the last layer uses a stride of 1 and a kernel size of 1×3. Similarly, these DC blocks have 16, 32, 64, 128 and 256 output channels successively.

In this noncausal DC-CRN, the RNN used for sequential modeling is a two-layer bidirectional LSTM (BLSTM), of which each layer contains 640 units in each direction. As in [32], we adopt a grouping strategy [9] to reduce the number of trainable parameters in the BLSTM without significantly affecting the performance. The number of groups is empirically set to 2.

2) Causal DC-CRN: A causal and small DC-CRN can be easily derived by simply changing the network configurations. First, we set the number of output channels of all DC blocks to 16, except that the last DC block in the decoder only has 2 output channels. Second, we use a two-layer unidirectional LSTM for

sequential modeling, which has 80 units in each layer. All other settings are the same as in the noncausal DC-CRN.

C. Training Objective

Following [41], we train the DC-CRN to perform dual-channel complex spectral mapping with a loss function as follows:

$$\begin{aligned}\mathcal{L}_{\text{RI+Mag}} = \frac{1}{M \cdot F} \sum_{m,f} &\left|\hat{S}_1^{(r)}(m,f) - S_1^{(r)}(m,f)\right| \\ &+ \left|\hat{S}_1^{(i)}(m,f) - S_1^{(i)}(m,f)\right| \\ &+ \left|\left|\hat{S}_1(m,f)\right| - \left|S_1(m,f)\right|\right|, \end{aligned} \tag{4}$$

where $\hat{S}_1^{(r)}$, $\hat{S}_1^{(i)}$, $S_1^{(r)}$ and $S_1^{(i)}$ represent the real (r) and imaginary (i) components of the enhanced spectrogram $\hat{S}_1$ and the clean spectrogram S_1 for the primary channel, respectively. Here $\|\cdot\|_1$ denotes the ℓ_1 norm, and M and F the number of time frames and frequency bins respectively. The estimated spectral magnitude is calculated from the estimated real and imaginary spectra, i.e. $|\hat{S}_1(m,f)| = \sqrt{\hat{S}_1^{(r)}(m,f)^2 + \hat{S}_1^{(i)}(m,f)^2}$.

The inclusion of the magnitude loss term penalizes the magnitude estimation error accompanied with the phase estimation error, given that the magnitude and the phase are coupled in the real and imaginary components. This penalty is beneficial due to the relative importance of the magnitude over the phase [35].

D. Iterative Structured Pruning

To further reduce the number of trainable parameters, we propose a structured pruning method to compress the causal DC-CRN, without significantly sacrificing the enhancement performance. Structured pruning is a class of coarse-grained parameter pruning techniques, and it leads to more regular sparsity patterns than unstructured pruning. For example, structured pruning can remove an entire column of a weight matrix, unlike unstructured pruning that prunes individual weights. The regularity of sparse structure makes it easier to apply hardware acceleration [21].

To prune the DC-CRN, we define the pruning granularity as follow. For each of the convolutional and deconvolutional layers, the weights compose a 4-D tensor of shape $C_1 \times C_2 \times K_1 \times K_2$, where C_1 and C_2 represent the output and input channel dimensions respectively, and K_1 and K_2 the shapes of convolution kernels. We treat each kernel (i.e. a $K_1 \times K_2$ matrix) as a weight group for pruning. Moreover, each of the LSTM layers is defined by the following equations:

$$\mathbf{i}_t = \sigma(\mathbf{W}_{ii}\mathbf{x}_t + \mathbf{b}_{ii} + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{b}_{hi}), \tag{5}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{if}\mathbf{x}_t + \mathbf{b}_{if} + \mathbf{W}_{hf}\mathbf{h}_{t-1} + \mathbf{b}_{hf}), \tag{6}$$

$$\mathbf{g}_t = \tanh(\mathbf{W}_{ig}\mathbf{x}_t + \mathbf{b}_{ig} + \mathbf{W}_{hg}\mathbf{h}_{t-1} + \mathbf{b}_{hg}), \tag{7}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{io}\mathbf{x}_t + \mathbf{b}_{io} + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{b}_{ho}), \tag{8}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \tag{9}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \tag{10}$$

where $\mathbf{x}_t$, $\mathbf{g}_t$, $\mathbf{c}_t$ and $\mathbf{h}_t$ denote the input, forget, cell and output gates at time step t, respectively. Here $\mathbf{W}$'s and $\mathbf{b}$'s represent weight matrices and bias vectors respectively, and σ and $\odot$ the sigmoid nonlinearity and the element-wise multiplication respectively. In the implementation of LSTM, the weight matrices for the four gates are typically concatenated, i.e. $\mathbf{W}_i = [\mathbf{W}_{ii}, \mathbf{W}_{if}, \mathbf{W}_{ig}, \mathbf{W}_{io}] \in \mathbb{R}^{4D_1 \times D_2}$ and $\mathbf{W}_h = [\mathbf{W}_{hi}, \mathbf{W}_{hf}, \mathbf{W}_{hg}, \mathbf{W}_{ho}] \in \mathbb{R}^{4D_1 \times D_1}$, where D_1 and D_2 are the output and input dimensions of the LSTM layer, respectively. We treat each column of $\mathbf{W}_i$ and $\mathbf{W}_h$ as a weight group for pruning. Similarly, we treat each column of the weight matrix of each linear layer as a weight group for pruning. Since the number of biases is small relative to that of weights, we only prune weights.

Algorithm 1: Per-tensor Sensitivity Analysis.

Input: (1) Validation set $\mathcal{V}$; (2) set $\mathcal{G}_l$ of all nonzero weight groups in the l-th weight tensor $\widetilde{\mathbf{W}}_l$, $\forall l$; (3) loss function $\mathcal{L}_{\text{RI+Mag}}(\mathcal{V}, \Theta)$, where Θ is the set of all nonzero trainable parameters in the model; (4) predefined tolerance value α.

Output: Pruning ratio β_l for weight tensor $\widetilde{\mathbf{W}}_l$, $\forall l$.

```
for each tensor W̃_l do
  for β in {0%, 5%, 10%, ..., 90%, 95%, 100%} do
    Let U ⊆ G_l be the set of the β(%) of nonzero weight groups with the smallest ℓ1 norms in tensor W̃_l;
    Calculate I_U = L_RI+Mag(V, Θ | g = 0, ∀g ∈ U) − L_RI+Mag(V, Θ);
    if I_U > α then
      β_l ← β − 5%;
      break
    end if
  end for
  if β_l is not assigned any value then
    β_l ← 100%;
  end if
end for
return β_l for weight tensor W̃_l, ∀l
```

In order to achieve a high compression rate, we adopt a group sparse regularization technique [27] to impose the group-level sparsity of the weight tensors. Specifically, we introduce the following sparse group lasso (SGL) [28] penalty:

$$\mathcal{R}_{\text{SGL}} = \frac{\lambda_1}{n(\mathcal{W})} \sum_{w \in \mathcal{W}} |w| + \frac{\lambda_2}{n(\mathcal{G})} \sum_{\mathbf{g} \in \mathcal{G}} \sqrt{p_{\mathbf{g}}} \left\|\mathbf{g}\right\|_2, \tag{11}$$

where $\mathcal{W}$ and $\mathcal{G}$ denote the set of all weights and that of all weight groups, respectively. The function $n(\cdot)$ calculates the cardinality of a set, and $\|\cdot\|_2$ the ℓ_2 norm. The number of weights in each weight group $\mathbf{g}$ is represented by $p_{\mathbf{g}}$. Here λ_1 and λ_2 are predefined weighting factors. Hence, the new loss function can be written as

$$\mathcal{L} = \mathcal{L}_{\text{RI+Mag}} + \mathcal{R}_{\text{SGL}}. \tag{12}$$

The importance of a specific set $\mathcal{U}$ of weight groups can be quantified by the error induced by removing (or zeroing out) it.

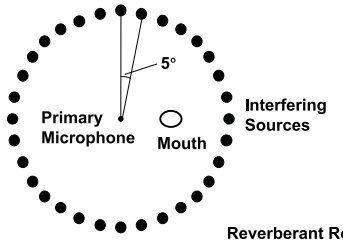


Fig. 5. Illustration of diffuse babble noise simulation.

This induced error can be measured as the increase in the loss on a validation set $\mathcal{V}$:

$$\mathcal{I}_{\mathcal{U}} = \mathcal{L}_{\text{RI+Mag}}(\mathcal{V}, \Theta | \mathbf{g} = \mathbf{0}, \forall \mathbf{g} \in \mathcal{U}) - \mathcal{L}_{\text{RI+Mag}}(\mathcal{V}, \Theta), \quad (13)$$

where Θ is the set of all trainable parameters in the model, and $\mathcal{U}$ can be any subset of $\mathcal{G}$. To determine the pruning ratio for each weight tensor, we perform a per-tensor sensitivity analysis following Algorithm 1. Subsequently, we perform group-level pruning as per tensor-wise pruning ratios, and then fine-tune the pruned model. We evaluate the fine-tuned model on the validation set by two standard metrics, i.e. short-time objective intelligibility (STOI) [29] and perceptual evaluation of speech quality (PESQ) [26]. This procedure is repeated until the number of pruned weights becomes trivial in an iteration or a significant degradation of STOI or PESQ is observed on the validation set. Note that the parameter set Θ becomes smaller after each iteration.

IV. Experimental Setup

A. Data Preparation

In our experiments, we use the training set of the WSJ0 corpus [10] for evaluation, which includes 12776 utterances from 101 speakers. These speakers are split into three groups for training, cross validation and testing, which contain 89, 6 (3 males and 3 females) and 6 (3 males and 3 females), respectively. Specifically, these groups include 11084, 846 and 846 clean utterances for creating the training, validation and test sets, respectively. We simulate a rectangular room with a size of $10\times7\times3$ m^3 using the image method [2]. The target speech source (mouth) is located at the center of the room, while the primary microphone is placed on a sphere centered at the target speech source with a radius randomly sampled between 0.01 m and 0.15 m. Such a distance range covers both hand-held and hands-free scenarios. We fix the geometry of the dual-channel microphone array, where the distance between microphones is 0.1 m. Thus the location of the secondary microphone is randomly chosen on a sphere with a radius of 0.1 m, which is centered at the primary microphone. The reverberation time (T_{60}) is randomly sampled between 0.2 s and 0.5 s. Following this procedure, we simulate a set of 5000 dual-channel RIRs for training and cross validation, and another set of 846 dual-channel RIRs for testing.

As illustrated in Fig. 5, we simulate a diffuse babble noise in the following way. We first concatenate the utterances spoken by each of the 630 speakers in the TIMIT corpus [11], and then split them into 480 and 150 speakers for training and testing. Following [45], we randomly select 72 speech clips from 72 randomly chosen speakers, and place them on a horizontal circle centered at and with the same height as the primary microphone, where the azimuths range from 0° to 355° with 5° steps. The distance between the primary microphone and each of the interfering sources is 2 m.

We create a training set including 40 000 mixtures, each of which is simulated by mixing a diffuse babble noise and a randomly sampled WSJ0 utterance convolved with a randomly selected RIR. To create the validation set, we convolve each of the 846 validation utterances with a randomly selected RIR, and then mix the reverberant speech signal with a random cut of diffuse babble noise at each channel. In order to mimic the head shadow effect, we downscale the amplitude of the speech signal at the secondary channel prior to mixing, where the downscaling ratio is randomly sampled between -10 and 0 dB. For both training and validation data, the SNR is randomly sampled between -5 and 0 dB, where the SNR is with respect to the reverberant speech signal and the reverberant noise signal at the primary channel. Similarly, we create a test set consisting of 846 mixtures for each of four SNRs, i.e. -5, 0, 5 and 10 dB.

In our experiments, all signals are sampled at 16 kHz. We rescale each noisy mixture by a factor, such that the root mean square of the mixture waveform is 1. The same factor is used to rescale the corresponding target speech waveform. Such noncausal signal level normalization is applied because we focus on speech enhancement and assume that the root mean square power of all input signals is the same. Thus the causal models can benefit from this noncausal normalization in our experiments. Real applications may need a causal automatic gain control for signal level normalization. Moreover, we use a 20-ms Hamming window to segment time-domain signals into a set of frames, with a 50% overlap between adjacent frames. A 320-point (16 kHz $\times$ 20 ms) discrete Fourier transform is applied to each frame, yielding 161-D one-sided spectra.

B. Baselines

In our preliminary study [33], the PSM is used as the training target, which is originally defined for the primary channel as follows:

$$\begin{aligned}\text{PSM}_1(m,f) &= \text{Re}\left\{\frac{|S_1(m,f)|e^{j\theta_{s_1}}}{|Y_1(m,f)|e^{j\theta_{y_1}}}\right\} \\ &= \frac{|S_1(m,f)|}{|Y_1(m,f)|}\cos(\theta_{s_1}-\theta_{y_1}), \end{aligned} \quad (14)$$

where $|S_1(m,f)|$ and $|Y_1(m,f)|$ denote the spectral magnitudes of clean speech and noisy speech within the T-F unit at frame m and frequency f respectively, and θ_{s_1} and θ_{y_1} the phases of clean speech and noisy speech within the unit respectively. $Re\{\cdot\}$ computes the real component. In [33], however, a modified

TABLE I
COMPARISONS OF ALTERNATIVE MODELS IN STOI, PESQ AND SNR. HERE √ INDICATES CAUSAL MODEL, AND ✗ INDICATES NONCAUSAL MODEL

Test SNR	-5 dB			0 dB			5 dB			10 dB			# Param.	Causal
Metric	STOI (%)	PESQ	SNR (dB)	STOI (%)	PESQ	SNR (dB)	STOI (%)	PESQ	SNR (dB)	STOI (%)	PESQ	SNR (dB)		
Unprocessed	58.71	1.49	-5.03	72.08	1.73	-0.05	83.53	2.04	4.91	91.41	2.38	9.76	-	-
NC-CRN-PSM$_1$	84.65	2.11	6.43	91.13	2.53	9.31	94.87	2.89	12.40	96.96	3.16	15.49	12.99 M	✗
NC-CRN-PSM$_2$	85.48	2.20	6.21	90.79	2.60	8.11	93.82	2.93	9.78	95.47	3.17	11.07	12.99 M	✗
NC-DC-CRN-RI	**92.77**	**3.07**	**10.90**	**96.09**	**3.41**	**13.82**	**97.66**	**3.63**	**16.38**	**98.45**	**3.78**	**18.49**	8.36 M	✗
IRM	92.02	2.83	6.47	94.21	3.10	9.07	96.24	3.39	11.95	97.74	3.68	14.95	-	-
PSM	94.08	3.16	9.02	96.26	3.40	11.81	97.87	3.66	15.03	98.87	3.88	18.43	-	-
C-CRN-PSM$_1$	78.20	1.72	5.25	87.30	2.17	8.29	92.76	2.59	11.53	95.76	2.99	14.63	73.15 K	✓
C-CRN-PSM$_2$	78.77	1.76	5.13	86.80	2.18	7.29	91.53	2.56	9.18	94.05	2.88	10.60	73.15 K	✓
C-DC-CRN-RI	**87.57**	**2.56**	**8.61**	**93.36**	**2.99**	**11.95**	**96.35**	**3.30**	**15.01**	**97.74**	**3.53**	**17.53**	290.44 K	✓
C-DC-CRN-RI-P1	86.88	2.54	8.55	93.08	2.97	11.82	96.16	3.26	14.76	97.63	3.46	17.15	124.96 K	✓
C-DC-CRN-RI-P2	87.13	2.56	8.46	93.10	2.98	11.75	96.14	3.27	14.72	97.62	3.47	17.11	113.68 K	✓
C-DC-CRN-RI-P3	86.64	2.52	8.50	92.89	2.95	11.77	96.07	3.26	14.75	97.61	3.47	17.18	108.77 K	✓
C-DC-CRN-RI-P4	86.63	2.49	8.43	92.85	2.91	11.70	96.03	3.22	14.66	97.59	3.44	17.11	106.21 K	✓
C-DC-CRN-RI-P5	86.63	2.48	8.36	92.86	2.90	11.62	96.07	3.20	14.54	97.65	3.43	16.97	104.76 K	✓
C-DC-CRN-RI-P6	86.45	2.51	8.21	92.64	2.94	11.42	95.88	3.27	14.29	97.47	3.51	16.60	103.07 K	✓

version is used:

$$\begin{aligned}\text{PSM}_2(m,f) &= \text{Re}\left\{\frac{|S_1(m,f)|e^{j\theta_{s_1}}}{|Y_1(m,f)|e^{j\theta_{y_1-y_2}}}\right\} \\ &= \frac{|S_1(m,f)|}{|Y_1(m,f)|}\cos(\theta_{s_1}-\theta_{y_1-y_2}),\end{aligned} \tag{15}$$

where $\theta_{y_1-y_2}$ represents the phase of the noisy signal difference between channels, i.e. $y_1 - y_2$. For PSM$_2$, $\theta_{y_1-y_2}$ is used to resynthesize waveforms, which was shown to improve both STOI and PESQ over using PSM$_1$ and θ_{y_1}. An interpretation is that the inter-channel PLD of speech signals is typically larger than that of noise signals due to the head shadow in hand-held scenarios. With a possible signal cancellation effect due to the subtraction, $y_1 - y_2$ may have a higher SNR and thus cleaner phase than y_1.

In [33], a CRN is employed to estimate PSM$_2$ from both intra-channel features (i.e. $|Y_1|$ and $|Y_2|$) and inter-channel features (i.e. $|Y_1 - Y_2|$ and $|Y_1 + Y_2|$). We refer to the approach in [33] as "C-CRN-PSM$_2$" ("C-CRN" indicates causal CRN), and another version that estimates PSM$_1$ as "C-CRN-PSM$_1$". In addition, we train a noncausal version of each of these two baselines, where the configuration of the CRN is changed as follows. The numbers of output channels for the layers in the encoder are changed to 16, 32, 64, 128 and 256 successively, and those for each layer in the decoder to 128, 64, 32, 16 and 1 successively. The two-layer LSTM is replaced by a two-layer BLSTM, of which each layer contains 512 units in each direction. These noncausal baselines are denoted as "NC-CRN-PSM$_1$" and "NC-CRN-PSM$_2$".

C. Training Methodology

The models are trained on 4-second segments using the AMSGrad optimizer [25] with a minibatch size of 16. The learning rate is initialized to 0.001, which decays by 0.98 every two epochs. We apply gradient clipping with a maximum ℓ_2 norm of 5 during training. The validation set is used for both selecting the best model among different epochs and performing the sensitivity analysis prior to pruning.

For structured pruning, the initial values of λ_1 and λ_2 (see Eq. (11)) are empirically set to 1 and 0.1, both of which decay by 10% every pruning iteration. We alternately prune and fine-tune the causal DC-CRN for 6 iterations. The tolerance value α for sensitivity analysis (see Algorithm 1) is set to 0.02.

V. EXPERIMENTAL RESULTS AND COMPARISONS

A. Model Comparison

Comprehensive comparisons among alternative models are shown in Table I, in terms of STOI, PESQ and SNR, where the numbers represent the averages over the test set in each condition. The proposed models with noncausal and causal DC-CRNs are denoted as "NC-DC-CRN-RI" and "C-DC-CRN-RI," respectively. The pruned DC-CRN model for the k-th iteration is represented by "C-DC-CRN-RI-Pk".

We can observe that using PSM$_1$ yields similar results to using PSM$_2$, unlike the finding that PSM$_2$ produces significantly better results than PSM$_1$ in [33]. This is likely because $\theta_{y_1-y_2}$ is not always cleaner than θ_{y_1} due to the variety of inter-channel decay ratios and no head shadow in hands-free scenarios. Moreover, our proposed approach substantially outperforms the approach in [33] in all the metrics. At -5 dB SNR, for example, "NC-DC-CRN-RI" improves STOI by 7.6%, PESQ by 0.89 and SNR by 4.77 dB, over "NC-CRN-PSM$_2$". Similar improvements are observed for "C-DC-CRN-RI" over "C-CRN-PSM$_2$". We additionally compare our approach with two ideal masks, i.e. the PSM (PSM$_1$) and the ideal ratio mask (IRM) [38], defined as

$$\text{IRM}(m,f) = \frac{|S_1(m,f)|}{|S_1(m,f)| + |H_1(m,f) + N_1(m,f)|}, \tag{16}$$

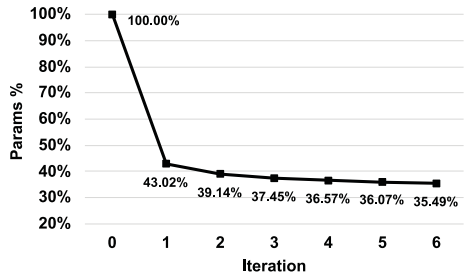
where H_1 and N_1 are the STFTs of reverberation and reverberant noise at the primary channel, respectively. As shown in Table I, our noncausal enhancement system ("NC-DC-CRN-RI") produces better results than the IRM in terms of all the three metrics. In addition, our system yields slightly lower STOI and PESQ but higher SNR than the PSM.

To demonstrate the generalization capability of the trained models, we create an additional test set by mixing real-recorded speech signals and simulated diffuse noise signals at -5, 0, 5 and 10 dB SNRs. Specifically, the diffuse noise is simulated using the same recipe as described in Section IV-A, where the noise source signals are recorded in eight different environments. The speech signals are recorded by a dual-microphone mobile phone (Meizu 15) that is mounted on a dummy head. The source signals[1] contain 20 utterances from four speakers (two

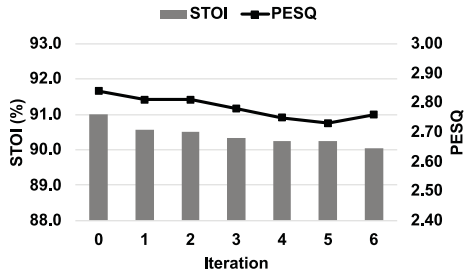
[1][Online] Available: https://docbox.etsi.org/stq/Open/TS%20103%20106%20Wave%20files/Annex_C_Dynastat%20Speech%20Data/

TABLE II
EVALUATION ON REAL-RECORDED SPEECH SIGNALS

Test SNR	-5 dB		0 dB		5 dB		10 dB	
Metric	STOI (%)	PESQ	STOI (%)	PESQ	STOI (%)	PESQ	STOI (%)	PESQ
Unprocessed	66.78	1.66	78.96	1.91	88.27	2.28	93.99	2.62
C-CRN-PSM$_1$	79.82	2.13	88.44	2.49	93.14	2.74	95.41	2.94
C-DC-CRN-RI	89.44	2.57	94.21	2.83	96.51	2.97	97.71	3.06
C-DC-CRN-RI-P6	87.51	2.58	92.99	2.76	95.80	2.87	97.23	2.97



(a)



(b)

Fig. 6. The percent of the original number of trainable parameters at different pruning iterations (a), and corresponding STOI and PESQ scores on the validation set (b).

males and two females), where each speaker reads five IEEE sentences [14]. The total duration of these utterances is roughly 80 seconds. They are mixed with the eight types of noises at the four SNRs, which amount to a set of noisy speech signals with a total duration of roughly 43 minutes ($\approx 80\times8\times4$ s). As shown in Table II, "C-DC-CRN-RI" and "C-DC-CRN-RI-P6" produce significantly higher STOI and PESQ than "C-CRN-PSM$_1$". Moreover, "C-DC-CRN-RI-P6" produces substantial improvements in STOI and PESQ over unprocessed mixtures, consistent with our finding from Table I. This suggests the robustness of our training data simulation method described in Section IV-A.

Furthermore, we compare the pruned DC-CRN models of different pruning iterations. As presented in Table I, the causal DC-CRN originally has 290.44 K trainable parameters. After 6 iterations of pruning, the number of trainable parameters in the DC-CRN becomes 103.07 K, which is comparable to that of the CRN in [33], i.e. 73.15 K. The model size reduction over pruning iterations is shown in Fig. 6(a). Compared with the original model, the performance of the pruned model after 6 iterations degrades only slightly. Take, for example, the 0 dB SNR case. Iterative pruning decreases STOI by 0.72%, PESQ by 0.05 and SNR by 0.53 dB. Fig. 6(b) shows the STOI and PESQ scores on the validation set over pruning iterations.

TABLE III
EFFECTS OF DENSE CONNECTIVITY AT -5 DB SNR

Test SNR	-5 dB			# Param.
Metric	STOI (%)	PESQ	SNR (dB)	
Unprocessed	58.71	1.49	-5.03	-
C-DC-CRN-RI	**87.57**	**2.56**	**8.61**	290.44 K
− DC$_{\text{Skip}}$ (i)	87.23	2.53	8.49	253.32 K
− DC$_{\text{ED}}$ (ii)	86.26	2.42	8.02	218.69 K
− DC$_{\text{Skip}}$ − DC$_{\text{ED}}$ (iii)	82.77	2.10	6.37	181.57 K

Moreover, we calculate the number of multiply-accumulate (MAC) operations on a 4-second noisy mixture, which is another common metric for evaluating model complexity. The number of MAC operations decreases from 1.97 G for "C-DC-CRN-RI" to 502.40 M for "C-DC-CRN-RI-P6". Thus the average number of MAC operations for processing a 1-second input signal is 125.60 M, which is amenable to mobile phones on the market. We additionally measure the computation time for "C-DC-CRN-RI-P6" on a Lenovo ThinkPad X1 laptop with Intel Core i7-10510U@1.80 GHz processors, and the average time of processing a 20-ms time frame is 2.78 ms, demonstrating real-time feasibility.

B. Ablation Study of Dense Connectivity

To investigate the contribution of dense connectivity in the DC-CRN, we conduct an ablation study at -5 dB SNR, as shown in Table III. Several variants of the causal DC-CRN are compared: (i) replacing the DC block based skip pathways by skip connections as in [33]; (ii) replacing each DC block in the encoder and the decoder by a corresponding gated convolutional or deconvolutional layer, as in [32]; (iii) doing both (i) and (ii). We can see that all these variants underperform the proposed causal DC-CRN, which suggests the effectiveness of dense connectivity. Without dense connectivity in the encoder and the decoder, for example, STOI decreases by 1.31% and PESQ by 0.14. Only removing the dense connectivity in the skip pathways does not significantly degrade the enhancement performance, if the DC blocks in the encoder and the decoder are preserved. However, going from (ii) to (iii) results in a significant performance loss. This is likely because the dense connectivity in the skip pathways compensates for the reduced representation power without DC blocks in the encoder and the decoder.

C. Inter-Channel Features

The approach in [33] exploits both intra- and inter-channel features in the magnitude domain, while our proposed approach performs dual-channel complex spectral mapping without explicitly using any inter-channel features. We now investigate the inclusion of inter-channel features for both these approaches.

TABLE IV
INVESTIGATION OF INTER-CHANNEL FEATURES FOR MAGNITUDE- AND COMPLEX-DOMAIN APPROACHES. "ICFS" REPRESENT THE INTER-CHANNEL FEATURES

Test SNR	-5 dB			Domain
Metric	STOI (%)	PESQ	SNR (dB)	
Unprocessed	58.71	1.49	-5.03	-
C-CRN-PSM$_1$ w/ ICFs	78.20	1.72	5.25	Magnitude
C-CRN-PSM$_1$ w/o ICFs	76.41	1.63	4.96	Magnitude
C-CRN-PSM$_2$ w/ ICFs	78.77	1.76	5.13	Magnitude
C-CRN-PSM$_2$ w/o ICFs	76.14	1.67	4.56	Magnitude
C-DC-CRN-RI w/ ICFs	87.64	2.56	8.44	Complex
C-DC-CRN-RI w/o ICFs	87.44	2.56	8.61	Complex

As shown in Table IV, the inclusion of inter-channel features significantly improves STOI, PESQ and SNR for the magnitude-domain approaches. For our approach based on complex spectral mapping, we use the real and imaginary components of $Y_1 - Y_2$ and $Y_1 + Y_2$ as the inter-channel features. With multi-channel complex spectral mapping, the explicit use of these inter-channel features does not produce performance gain, as shown in Table IV. Unlike the magnitude spectrograms, the complex spectrograms encode both magnitude and phase information. Hence inter-channel features can be captured implicitly through DNN training that learns multi-channel complex spectral mapping, consistent with [41] which demonstrates the effectiveness of multi-channel to single-channel complex spectral mapping for speech dereverberation.

D. Comparison With Beamforming

We now compare the proposed approach with DNN-based beamforming (BF) [12], [13], [46]. Following [40], we formulate an MVDR beamformer, where the speech and noise covariance matrices are estimated as

$$\hat{\mathbf{\Phi}}_s(f) = \sum_m \frac{\eta(m,f)}{\sum_m \eta(m,f)} \mathbf{Y}(m,f)\mathbf{Y}(m,f)^H, \tag{17}$$

$$\hat{\mathbf{\Phi}}_v(f) = \sum_m \frac{\xi(m,f)}{\sum_m \xi(m,f)} \mathbf{Y}(m,f)\mathbf{Y}(m,f)^H, \tag{18}$$

where $(\cdot)^H$ denotes the conjugate transpose, and $\eta(m,f)$ and $\xi(m,f)$ the weighting factors representing the importance of each T-F unit for the covariance matrix computation. These weighting factors are calculated as the product of estimated T-F masks for different channels:

$$\eta(m,f) = \prod_{i=1}^{D} \hat{M}_i(m,f), \tag{19}$$

$$\xi(m,f) = \prod_{i=1}^{D} (1 - \hat{M}_i(m,f)), \tag{20}$$

where $D = 2$ is the number of channels, and $\hat{M}_i(m,f)$ the ratio mask for the i-th microphone. These ratio masks are individually estimated by a noncausal DC-CRN that is monaurally trained to estimate the IRM for each channel. We treat the primary channel as the reference channel, and estimate the inter-channel relative transfer function (i.e. steering vector) as

$$\hat{\mathbf{r}}(f) = \mathcal{P}\{\hat{\mathbf{\Phi}}_s(f)\} = [\hat{r}_1(f), \hat{r}_2(f)]^T, \tag{21}$$

$$\hat{\mathbf{c}}(f) = \frac{\hat{\mathbf{r}}(f)}{\hat{r}_1(f)}, \tag{22}$$

where $\mathcal{P}\{\cdot\}$ computes the principal eigenvector. The MVDR filter is then calculated as

$$\hat{\mathbf{w}}(f) = \frac{\hat{\mathbf{\Phi}}_v(f)^{-1}\hat{\mathbf{c}}(f)}{\hat{\mathbf{c}}(f)^H \hat{\mathbf{\Phi}}_v(f)^{-1}\hat{\mathbf{c}}(f)}, \tag{23}$$

and the enhanced spectrogram is obtained by $\hat{S}(m,f) = \hat{\mathbf{w}}(f)^H \mathbf{Y}(m,f)$. To improve the enhancement performance, the monaural DC-CRN trained for IRM estimation is used as a post-filter (PF). As shown in Table V, this masking-based beamforming algorithm outperforms a noncausal DC-CRN that estimates the IRM from the magnitude spectrograms of the two channels, in terms of both STOI and PESQ.

TABLE V
COMPARISONS WITH BEAMFORMING IN MAGNITUDE AND COMPLEX DOMAINS AT -5 AND 0 DB SNRS

Test SNR	-5 dB		0 dB		Domain
Metric	STOI (%)	PESQ	STOI (%)	PESQ	
Unprocessed	58.71	1.49	72.08	1.73	-
Mask-BF	68.85	1.64	81.37	1.92	Magnitude
Mask-BF-PF	**86.32**	**2.49**	**92.60**	**2.91**	Magnitude
NC-DC-CRN-IRM	85.21	2.42	90.95	2.79	Magnitude
RI-BF	71.93	1.65	82.51	1.93	Complex
RI-BF-PF	91.03	2.94	95.03	3.31	Complex
NC-DC-CRN-RI	**92.77**	**3.07**	**96.09**	**3.41**	Complex

In addition, we formulate a variant of the aforementioned MVDR beamformer following [41], where the speech and noise covariance matrices are estimated as

$$\hat{\mathbf{\Phi}}_s(f) = \frac{1}{M}\sum_{m=1}^{M} \hat{\mathbf{S}}(m,f)\hat{\mathbf{S}}(m,f)^H, \tag{24}$$

$$\hat{\mathbf{\Phi}}_v(f) = \frac{1}{M}\sum_{m=1}^{M} \hat{\mathbf{V}}(m,f)\hat{\mathbf{V}}(m,f)^H, \tag{25}$$

where M is the number of time frames. The complex spectrogram $\hat{\mathbf{S}}$ is estimated by performing a monaural complex spectral mapping using a noncausal DC-CRN. Then the estimated noise spectrogram is calculated as $\hat{\mathbf{V}} = \mathbf{Y} - \hat{\mathbf{S}}$. Akin to masking-based beamforming, we obtain the spatial filter using (21)–(23). The DC-CRN for monaural complex spectral mapping is used as a post-filter. As shown in Table V, our proposed approach outperforms the beamformer in terms of both STOI and PESQ, which further suggests that dual-channel complex spectral mapping can effectively exploit spatial cues encoded in the dual-channel complex spectrogram.

VI. CONCLUSION

In this study, we have proposed a novel framework for dual-channel speech enhancement on mobile phones, which employs a new causal DC-CRN to perform dual-channel complex spectral mapping. By applying an iterative structured pruning technique, we derive a low-latency and memory-efficient enhancement

system that is amenable to real-time processing on mobile phones. Evaluation results demonstrate that the proposed approach significantly outperforms an earlier method for speech enhancement for dual-microphone mobile phones. Moreover, our approach consistently outperforms a DNN-based beamformer, which suggests that multi-channel complex spectral mapping can effectively extract and utilize spatial cues encoded in the multi-channel complex spectrogram.

REFERENCES

[1] A. Aitken, C. Ledig, L. Theis, J. Caballero, Z. Wang, and W. Shi, "Checkerboard artifact free sub-pixel convolution: A note on sub-pixel convolution, resize convolution and convolution resize," 2017, *arXiv:1707.02937*.

[2] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Amer.*, vol. 65, no. 4, pp. 943–950, 1979.

[3] Y.-Y. Chen, "Speech enhancement of mobile devices based on the integration of a dual microphone array and a background noise elimination algorithm," *Sensors*, vol. 18, no. 5, 2018, Art. no. 1467.

[4] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," in *Proc.Int. Conf. Learn. Representations*, 2016.

[5] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 933–941.

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

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

[8] Z.-H. Fu, F. Fan, and J.-D. Huang, "Dual-microphone noise reduction for mobile phone application," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2013, pp. 7239–7243.

[9] F. Gao, L. Wu, L. Zhao, T. Qin, X. Cheng, and T.-Y. Liu, "Efficient sequence learning with group recurrent networks," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Lang. Technol.*, vol. 1, 2018, pp. 799–808.

[10] J. Garofolo, D. Graff, D. Paul, and D. Pallett, "CSR-I (WSJ0) Complete LDC93S6A. *Web Download. Philadelphia: Linguistic Data Consortium*, vol. 83, 1993.

[11] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, "DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM," *NIST Speech Disc 1-1.1. STIN*, vol. 93, 1993, Art. no. 27403.

[12] J. Heymann, L. Drude, and R. Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2016, pp. 196–200.

[13] T. Higuchi, N. Ito, T. Yoshioka, and T. Nakatani, "Robust MVDR beamforming using time-frequency masks for online/offline ASR in noise," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2016, pp. 5210–5214.

[14] IEEE, "IEEE recommended practice for speech quality measurements," *IEEE Trans. Audio Electroacoust.*, vol. 17, no. 3, pp. 225–246, Jun. 1969.

[15] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[16] M. Jeub, C. Herglotz, C. Nelke, C. Beaugeant, and P. Vary, "Noise reduction for dual-microphone mobile phones exploiting power level differences," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2012, pp. 1693–1696.

[17] Y. Koizumi, K. Yaiabe, M. Delcroix, Y. Maxuxama, and D. Takeuchi, "Speech enhancement using self-adaptation and multi-head self-attention," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 181–185.

[18] I. López-Espejo, J. A. González, Á. M. Gómez, and A. M. Peinado, "A. deep neural network approach for missing-data mask estimation on dual-microphone smartphones: Application to noise-robust speech recognition," in *Proc. Adv. Speech Lang. Technol. Iberian Lang.*, 2014, pp. 119–128.

[19] I. López-Espejo, J. M. Martín-Doñas, A. M. Gomez, and A. M. Peinado, "Unscented transform-based dual-channel noise estimation: Application to speech enhancement on smartphones," in *Proc. 41st Int. Conf. Telecommun. Signal Process.*, 2018, pp. 1–5.

[20] I. López-Espejo, A. M. Peinado, A. M. Gomez, and J. M. Martín-Doñas, "Deep neural network-based noise estimation for robust ASR in dual-microphone smartphones," in *Proc. Int. Conf. Adv. Speech Lang. Technol. Iberian Lang.*, 2016, pp. 117–127.

[21] H. Mao *et al.*, "Exploring the granularity of sparsity in convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2017, pp. 13–20.

[22] W. Nabi, N. Aloui, and A. Cherif, "Speech enhancement in dual-microphone mobile phones using kalman filter," *Appl. Acoust.*, vol. 109, pp. 1–4, 2016.

[23] G. Naithani, T. Barker, G. Parascandolo, L. Bramsl, N. H. Pontoppidan, and T. Virtanen, "Low latency sound source separation using convolutional recurrent neural networks," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2017, pp. 71–75.

[24] A. Pandey and D. L. Wang, "Densely connected neural network with dilated convolutions for real-time speech enhancement in the time domain," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 6629–6633.

[25] K. Tan and D.L. Wang, "Learning complex spectral mapping with gated convolutional recurrent networks for monaural speech enhancement," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 380–390, 2020.

[26] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)-A new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., Proc. (Cat. No 01CH37221)*, vol. 2, 2001, pp. 749–752.

[27] S. Scardapane, D. Comminiello, A. Hussain, and A. Uncini, "Group sparse regularization for deep neural networks," *Neurocomputing*, vol. 241, pp. 81–89, 2017.

[28] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A sparse-group lasso," *J. Comput. Graphical Statist.*, vol. 22, no. 2, pp. 231–245, 2013.

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

[30] K. Tan, J. Chen, and D. L. Wang, "Gated residual networks with dilated convolutions for monaural speech enhancement," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 1, pp. 189–198, Jan. 2019.

[31] K. Tan and D. L. Wang, "A. convolutional recurrent neural network for real-time speech enhancement," in *Proc. INTERSPEECH*, 2018, pp. 3229–3233.

[32] K. Tan and D. L. Wang, "Learning complex spectral mapping with gated convolutional recurrent networks for monaural speech enhancement," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, no. 1, pp. 380–390, 2020.

[33] K. Tan, X. Zhang, and D. L. Wang, "Real-time speech enhancement using an efficient convolutional recurrent network for dual-microphone mobile phones in close-talk scenarios," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 5751–5755.

[34] B. D. Van Veen and K. M. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE ASSP Mag.*, vol. 5, no. 2, pp. 4–24, Apr. 1988.

[35] D. Wang and J. Lim, "The unimportance of phase in speech enhancement," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 30, no. 4, pp. 679–681, Aug. 1982.

[36] D. L. Wang and G. J. Brown, editors. *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. Wiley-IEEE Press, 2006.

[37] D. L. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 10, pp. 1702–1726, Oct. 2018.

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

[39] Y. Wang and D. L. Wang, "A. deep neural network for time-domain signal reconstruction," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2015, pp. 4390–4394.

[40] Z.-Q. Wang and D. L. Wang, "All-neural multi-channel speech enhancement," in *Proc. INTERSPEECH*, 2018, pp. 3234–3238.

[41] Z.-Q. Wang and D. L. Wang, "Multi-microphone complex spectral mapping for speech dereverberation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 486–490.

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

[43] N. Yousefian, A. Akbari, and M. Rahmani, "Using power level difference for near field dual-microphone speech enhancement," *Appl. Acoust.*, vol. 70, no. 11-12, pp. 1412–1421, 2009.

[44] J. Zhang, R. Xia, Z. Fu, J. Li, and Y. Yan, "A. fast two-microphone noise reduction algorithm based on power level ratio for mobile phone," in *Proc. 8th Int. Symp. Chin. Spoken Lang. Process.*, 2012, pp. 206–209.

[45] X. Zhang and D. L. Wang, "Deep learning based binaural speech separation in reverberant environments," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 5, pp. 1075–1084, May 2017.

[46] X. Zhang, Z.-Q. Wang, and D. L. Wang, "A. speech enhancement algorithm by iterating single-and multi-microphone processing and its application to robust ASR," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2017, pp. 276–280.

[47] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet : A nested U-net architecture for medical image segmentation," in *Deep Learn. Med. Image Anal. Multimodal Learn. Clin. Decis. Support*. Springer, 2018, pp. 3–11.