ctPuLSE: Close-talk, and pseudo-label based far-field, speech enhancement

Zhong-Qiu Wang[a] (ID)

Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, Guangdong, People's Republic of China

ABSTRACT:
The current dominant approach for neural speech enhancement is via purely supervised deep learning on simulated pairs of far-field noisy-reverberant speech (i.e., mixtures) and clean speech. The trained models, however, often exhibit limited generalizability to real-recorded mixtures. To deal with this, this paper investigates training enhancement models directly on real mixtures. However, a major difficulty challenging this approach is that, since the clean speech of real mixtures is unavailable, there lacks a good supervision for real mixtures. In this context, assuming that a training set consisting of real-recorded pairs of close-talk and far-field mixtures is available, we propose to address this difficulty via close-talk speech enhancement, where an enhancement model is first trained on simulated mixtures to enhance real-recorded close-talk mixtures and the estimated close-talk speech can then be utilized as a supervision (i.e., pseudo-label) for training far-field speech enhancement models directly on the paired real-recorded far-field mixtures. We name the proposed system *ctPuLSE*. Evaluation results on the popular CHiME-4 dataset show that ctPuLSE can derive high-quality pseudo-labels and yield far-field speech enhancement models with strong generalizability to real data. © 2025 Acoustical Society of America. https://doi.org/10.1121/10.0039557

I. INTRODUCTION

Dramatic progress has been made in speech enhancement (Wang and Chen, 2018), thanks to the rapid development of deep learning. The current dominant approach is based on supervised deep learning, where paired noisy-reverberant speech (i.e., mixtures) and clean speech are simulated and utilized to train deep neural networks (DNN) to predict the clean speech based on its paired mixtures in a purely supervised, discriminative way (Wang and Chen, 2018; Zheng *et al.*, 2023; Araki *et al.*, 2025). The trained models, however, often exhibit limited generalizability to real-recorded mixtures (Wang, 2024b; Pandey and Wang, 2020; Zhang *et al.*, 2021; Tzinis *et al.*, 2022a; Tzinis *et al.*, 2022b; Cox *et al.*, 2023; Leglaive *et al.*, 2023; Cornell *et al.*, 2023; Haeb-Umbach *et al.*, 2019; Zhang *et al.*, 2023; Zhang *et al.*, 2024), mainly because the simulated training data is typically, and in many cases inevitably, mismatched with real test data.

To improve the generalizability, this paper investigates training models directly on real-recorded, target-domain mixtures. This approach, however, cannot be straightforwardly realized, since the underlying clean speech is not available for real mixtures, unlike simulated mixtures, where the paired clean speech can be readily available through room simulation and can serve as a fine-grained supervision at the sample level for model training. The key to enable successful training on real mixtures, we think, is to

figure out a mechanism that can reliably compute a high-quality supervision signal (i.e., pseudo-label or pseudo-target) for real-recorded mixtures.

One possible way, we propose, is to leverage close-talk mixtures. During data collection, besides using far-field microphones to record target speech, a close-talk microphone is often placed near the target speaker to record the close-talk speech at the same time. See Fig. 1 for an illustration. The recorded close-talk mixture, while the target speaker is talking, usually has a much higher SNR of the target speaker than any far-field mixture, simply due to the very short distance from the target speaker to its close-talk microphone. Although the close-talk mixture is typically not perfectly clean, as non-target signals (such as environmental noises, room reverberation, and competing speech) could also be picked up by the close-talk microphone, it usually exhibits a very high input SNR and hence could be utilized to compute a reliable, high-quality supervision for real-recorded far-field mixtures.

With this understanding, this paper first investigates *close-talk speech enhancement*, a task which aims at enhancing close-talk mixtures and estimating close-talk speech. Solving this task could enable many applications. One of them, which is investigated in this paper, is that, based on the estimated close-talk speech, a pseudo-label can be derived for each real-recorded far-field mixture and used as a supervision to train supervised models directly on real-recorded far-field mixtures, thereby potentially realizing better generalizability to real data. We summarize the contributions of this paper as follows:

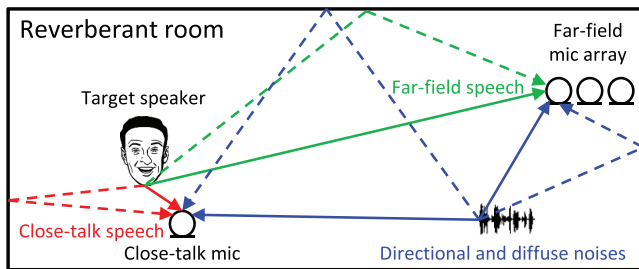
[a] Email: wang.zhongqiu41@gmail.com

FIG. 1. Scenario illustration. Close-talk mixture consists of close-talk speech and noises, and far-field mixture consists of far-field speech and noises.

- For the purpose of achieving better far-field speech enhancement, we propose to first investigate close-talk speech enhancement (CTSE), which aims at enhancing close-talk mixtures to estimate close-talk speech. Although CTSE is a particular form of speech enhancement (i.e., in close-talk conditions) and has been studied for other purposes (Jiang *et al.*, 2013; Jiang *et al.*, 2016; Tan *et al.*, 2021), we point out that it is a valuable task that could yield high-quality pseudo-labels for real-recorded data, and is hence worth investigating.
- We propose ctPuLSE, a pseudo-label based approach for far-field speech enhancement, where estimated close-talk speech is utilized to derive a supervision for training supervised enhancement models directly based on real far-field mixtures.
- Following SuperM2M (Wang, 2024b), an earlier algorithm that trains enhancement models by alternating between supervised and unsupervised/weakly supervised learning, we propose a co-learning algorithm that trains the same enhancement model using both simulated and real data, where the pseudo-label of real data is derived based on the estimated close-talk speech. This way, the model can also learn from massive amount of simulated training data, which can be easily simulated and can be very helpful when the real training data is scarce. Compared with SuperM2M, we observe that ctPuLSE obtains comparable robust automatic speech recognition (ASR) performance, while much better enhancement performance.

Although ctPuLSE is simple and straightforward, it obtains strong robust ASR and speech enhancement performance on the public CHiME-4 dataset (Barker *et al.*, 2015; Vincent *et al.*, 2017; Barker *et al.*, 2017), the most popular benchmark to date in robust ASR and speech enhancement. The evaluation results suggest that ctPuLSE can effectively train enhancement models on real-recorded far-field mixtures, and can yield better generalizability to real data than purely supervised models trained only on simulated data. A sound demo is provided online (Wang, 2025).

II. RELATED WORK

ctPuLSE is related to earlier works in four key aspects.

A. Generalizability of supervised enhancement models

Improving the generalizability of supervised learning based speech enhancement models (trained on simulated data) to real-recorded data has received decade-long research interests. The current dominant approach (Chen and Wang, 2017; Chen *et al.*, 2016; Wang and Chen, 2018; Zhang *et al.*, 2023; Zhang *et al.*, 2024) is to simulate massive amount of training data to cover as many variations (that could happen in real-recorded test data) as possible. However, the generalizability is often limited by the current simulation techniques, which usually cannot simulate mixtures as realistic as real-recorded mixtures. This can be observed from recent speech enhancement/separation and robust ASR challenges such as CHiME-{4,5,6,7} (Vincent *et al.*, 2017; Watanabe *et al.*, 2020; Cornell *et al.*, 2023), AliMeeting (Yu *et al.*, 2022), MISP (Wu *et al.*, 2024), and Clarity (Cox *et al.*, 2023), where using the immediate outputs from DNN-based enhancement or separation models trained on simulated mixtures for robust ASR and human hearing has had limited successes (Haeb-Umbach *et al.*, 2019; Haeb-Umbach *et al.*, 2021). Different from this stream of research, this paper investigates training enhancement models directly on real-recorded data to improve the generalizability.

B. Leveraging close-talk mixtures for speech enhancement

Leveraging real-recorded close-talk mixtures as a weak supervision to train speech enhancement and speaker separation models directly on real-recorded far-field mixtures has attracted research interests recently. A representative algorithm in this direction is SuperM2M (Wang, 2024b), which builds upon a weakly supervised speaker separation algorithm named mixture to mixture (M2M) (Wang, 2024a) and an unsupervised speaker separation algorithm named UNSSOR (Wang and Watanabe, 2023). M2M trains enhancement models to produce a speech estimate and a noise estimate such that the two estimates can be linearly filtered to recover observed mixtures, and SuperM2M, building upon M2M, leverages supervised learning on simulated mixtures to improve M2M. Although SuperM2M has shown strong potential for neural speech enhancement and robust ASR, we often observe that SuperM2M cannot sufficiently suppress non-target signals, likely because its loss function is defined on reconstructed mixtures (i.e., the summation of linearly filtered source estimates), rather than directly on source estimates. In comparison, the loss function in ctPuLSE, which we will show, is defined based on pseudo-labels derived from close-talk mixtures. As long as the pseudo-labels are high-quality, we can reasonably expect that the suppression of non-target signals by ctPuLSE would be more sufficient and aggressive.

On the other hand, there are studies (Wu *et al.*, 2024) using oracle speaker-activity timestamps and pre-trained models [e.g., DNSMOS (Reddy *et al.*, 2021)] to select

segments of close-talk mixtures that are almost clean, and then using the selected segments of mixtures to synthesize far-field mixtures for training supervised learning based models. However, the training data is still simulated, and the models are not trained on real data.

C. Pseudo-label based speech enhancement

There have been studies adapting pre-trained speech enhancement models to target domains via pseudo-labeling. RemixIT (Tzinis *et al.*, 2022a), a representative algorithm in this direction, first uses a pre-trained enhancement model (named *teacher*) to enhance target-domain real-recorded mixtures (and generate pseudo-labels), and then another enhancement model (named *student*) is trained in a supervised way to estimate the generated pseudo-labels. The teacher and student models are designed to update continuously and iteratively to gradually improve each other. Another recent study in this direction is SSST (Frenkel *et al.*, 2024), which, building upon RemixIT, introduces an adversarial training algorithm to learn domain-invariant hidden representations and proposes a data-selection mechanism to pick a subset of target-domain mixtures whose pseudo-labels are sufficiently reliable for RemixIT-style training.

Similarly to those studies, ctPuLSE leverages pseudo-labeling as well, but it has a very different problem setup, where paired close-talk and far-field mixtures are assumed available for model training. In this setup, much better pseudo-labels could potentially be computed for far-field mixtures, simply due to the innate high input SNR of close-talk mixtures. This potential could make ctPuLSE a more attractive solution for practical product development, as long as the paired close-talk mixtures can be recorded while collecting far-field mixtures for training.

D. Relations to DAPS dataset

A dataset named DAPS (Mysore, 2015) is proposed for the task of transforming speech recorded on common consumer (and possibly band limited) devices in real-world noisy-reverberant environments into professional production quality speech. A loudspeaker is used to play back clean speech in real-world environments, and the speech is then recorded by consumer devices to create pairs of clean and degraded speech signals for model training. Although the DAPS paper proposes a task similar to the one in this study, it proposes a new dataset only rather than an actual algorithm. On the other hand, in our study the close-talk mixture is recorded by attaching a close-talk microphone to the target talker and hence would inevitably capture some interference signals, and the pseudo-label is then derived based on the close-talk mixture. Differently, DAPS uses played-back clean speech, which is used to derive pseudo-labels.

E. Close-talk speech enhancement

Our study leverages close-talk speech enhancement to derive high-quality pseudo-labels for far-field speech enhancement. There are existing studies (Jiang *et al.*, 2013;

Jiang *et al.*, 2016; Tan *et al.*, 2021) on close-talk speech enhancement, but for different purposes. For example, in Jiang *et al.* (2013) and Jiang *et al.* (2016), the task is to enhance the target speech captured by a close-talk microphone with the help of an ear-mounted microphone, and in Tan *et al.* (2021), the task is to enhance close-talk speech when users talk to dual-microphone mobile phones at a very short distance. Their purpose is not on training speech enhancement models on real data and improving the generalizability to real data.

III. PHYSICAL MODEL AND APPROACH OVERVIEW

In a noisy-reverberant environment with a P-microphone far-field microphone array and a single target speaker wearing a close-talk microphone (see Fig. 1 for an illustration), the physical model of the recorded close-talk mixture and each far-field mixture can be respectively formulated, in the short-time Fourier transform (STFT) domain, as follows:

$$Y_0(t,f) = X_0(t,f) + V_0(t,f), \tag{1}$$

$$Y_p(t,f) = X_p(t,f) + V_p(t,f), \tag{2}$$

where the subscript $p \in \{1, \dots, P\}$ indexes the P far-field microphones, subscript 0 indexes the close-talk microphone, $t \in \{0, \dots, T-1\}$ indexes T frames, and $f \in \{0, \dots, F-1\}$ indexes F frequency bins. In Eq. (1), $Y_0(t,f)$, $X_0(t,f)$, and $V_0(t,f)$, respectively, denote the STFT coefficients of the close-talk mixture, close-talk speech, and non-speech signals (such as environmental noises) captured by the close-talk microphone at time t and frequency f. Similarly, in Eq. (2), $Y_p(t,f)$, $X_p(t,f)$, and $V_p(t,f)$ are, respectively, the STFT coefficients of the far-field mixture, far-field speech, and non-speech signals captured by far-field microphone p at time t and frequency f. In the rest of this paper, when dropping the indices p, t, and f, we refer to the corresponding spectrograms. In this paper, V is assumed to contain an unknown number of strong, non-stationary diffuse and directional noises sources.

Figure 2 illustrates the proposed ctPuLSE algorithm. Given a set of simulated pairs of clean speech and mixtures as well as a set of real-recorded pairs of close-talk and far-field mixtures for training, the proposed system consists of three steps: (a) we train a speech enhancement model (denoted as CTSEnet) based on the simulated training mixtures for close-talk speech enhancement, (b) we apply the trained CTSEnet to enhance each real-recorded close-talk mixture and obtain an estimated close-talk speech, and (c) we leverage the estimated close-talk speech as pseudo-labels to train a far-field speech enhancement model (denoted as ctPuLSEnet) on the paired real-recorded far-field mixtures.

Next, we detail CTSEnet and ctPuLSEnet.

IV. CTSEnet

Based on a set of simulated training mixtures, we train CTSEnet via single-microphone complex spectral mapping
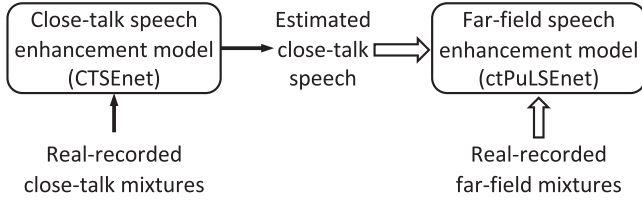
FIG. 2. ctPuLSE illustration. A CTSEnet is first trained on simulated mixtures for close-talk speech enhancement. It is then used to enhance real-recorded close-talk mixtures and obtain estimated close-talk speech. Next, a ctPuLSEnet is trained for far-field speech enhancement, based on the estimated close-talk speech and its paired real-recorded far-field mixtures.

(Tan and Wang, 2020; Wang *et al.*, 2020) to predict the real and imaginary (RI) components of target speech X_a based on the RI components of the input mixture Y_a, where, depending on the application scenarios, the microphone index a can index all or a subset of the simulated microphones in the training data. Following Wang *et al.* (2021c), we define the loss function on the RI components and magnitude of the DNN estimate $\hat{X}_a$,

$$\mathcal{L}_{X,a}^{\text{simu}} = \frac{\sum\limits_{t,f} \mathcal{F}(\hat{X}_a(t,f), X_a(t,f))}{\sum\limits_{t',f'} |X_a(t',f')|}, \quad (3)$$

$$\begin{aligned} \mathcal{F}(\hat{X}_a(t,f), X_a(t,f)) = & |\mathcal{R}(\hat{X}_a(t,f)) - \mathcal{R}(X_a(t,f))| \\ & + |\mathcal{I}(\hat{X}_a(t,f)) - \mathcal{I}(X_a(t,f))| \\ & + ||\hat{X}_a(t,f)| - |X_a(t,f)||, \quad (4) \end{aligned}$$

where $|\cdot|$ computes the magnitude or absolute value, $\mathcal{R}(\cdot)$ and $\mathcal{I}(\cdot)$, respectively, extract the RI components, and the denominator in Eq. (3) balances the loss values across different training mixtures. Other configurations of CTSEnet are detailed later in Secs. VI C and VI D.

V. CTPULSENET

Once a CTSEnet has been trained, we apply it to enhance each monaural real-recorded close-talk mixture to obtain an estimate of the close-talk speech, denoted as $\hat{X}_0^{\text{CTSE}}$, which is then leveraged as a pseudo-label to train ctPuLSEnet on real-recorded far-field mixtures. This section describes the DNN configurations, loss functions, and a co-learning algorithm that trains ctPuLSEnet on both simulated and real mixtures.

A. DNN configurations

ctPuLSEnet is trained via single- or multi-microphone complex spectral mapping (Wang and Wang, 2020; Wang *et al.*, 2021a; Tan *et al.*, 2022) to predict the RI components of target speech at a designated reference microphone $q \in \{1, \dots, P\}$ based on the RI components of stacked input mixtures. In monaural cases, the DNN is trained to predict the target speech X_q based on input mixture Y_q, and in multi-channel cases, it is trained to predict X_q based on all the input mixtures stacked in a fixed microphone order. We denote the

DNN estimate as $\hat{X}_q$. In addition to predicting X_q, the DNN can also be trained to additionally predict non-speech signals V_q. We observe that this form of multi-task learning can improve enhancement and robust ASR in our experiments.

Next, we describe the loss functions for the speech estimate $\hat{X}_q$ in Sec. V B and for the noise estimate $\hat{V}_q$ in V D. Other DNN setups are detailed in Secs. VI C and VI D.

B. Loss functions based on estimated close-talk speech

Assuming that the close-talk microphone and far-field microphones are approximately time-synchronized, we directly leverage $\hat{X}_0^{\text{CTSE}}$ as the pseudo-label for each far-field mixture Y_p. Since $\hat{X}_0^{\text{CTSE}}$ is time- and gain-aligned to close-talk speech X_0 rather than to far-field speech X_p in far-field mixture Y_p, special care is needed to account for the time delay and gain differences between $\hat{X}_0^{\text{CTSE}}$ and X_p.

In this context, we propose to first linearly filter the DNN estimate $\hat{X}_p$ at each frequency to align it to $\hat{X}_0^{\text{CTSE}}$ before loss computation,

$$\mathcal{L}_X^{\text{real}} = \frac{\sum\limits_{t,f} \mathcal{F}(\hat{X}_{p\to0}(t,f), \hat{X}_0^{\text{CTSE}}(t,f))}{\sum\limits_{t',f'} |\hat{X}_0^{\text{CTSE}}(t',f')|}, \quad (5)$$

$$\hat{X}_{p\to0}(t,f) = \hat{\mathbf{g}}_p(f)^{\text{H}} \tilde{\hat{\mathbf{X}}}_p(t,f), \quad (6)$$

where $\tilde{\hat{\mathbf{X}}}_p(t,f) = [\hat{X}_p(t-I+1,f), \dots, \hat{X}_p(t,f), \dots, \hat{X}_p(t+J,f)] \in \mathbb{C}^{I+J}$ stacks a window of T-F units, $\hat{\mathbf{g}}_p(f) \in \mathbb{C}^{I+J}$ denotes an estimated multi-tap linear filter to be described in Eq. (7), and $\mathcal{F}(\cdot,\cdot)$ is a distance metric defined in Sec. IV. Following the forward convolutive prediction (FCP) algorithm (Wang *et al.*, 2021), we estimate $\mathbf{g}_p(f)$ by linearly projecting the DNN estimate $\hat{X}_p(\cdot,f)$ to pseudo-label $\hat{X}_0^{\text{CTSE}}(\cdot,f)$ at each frequency f,

$$\hat{\mathbf{g}}_p(f) = \underset{\mathbf{g}_p(f)}{\text{argmin}} \sum_t |\hat{X}_0^{\text{CTSE}}(t,f) - \mathbf{g}_p(f)^{\text{H}} \tilde{\hat{\mathbf{X}}}_p(t,f)|^2. \quad (7)$$

This is a quadratic problem, which has a closed-form solution. Next, we plug the closed-form solution into Eq. (6), compute the loss in Eq. (5), and train the DNN.

An alternative way for linear filtering is to estimate the filter in the time domain, following the ideas behind CI-SDR (Boeddeker *et al.*, 2021). In detail, we compute a time-domain multi-tap Wiener filter to align $\hat{x}_p$ to $\hat{x}_0^{\text{CTSE}}$, where $\hat{x}_p = \text{iSTFT}(\hat{X}_p)$ is the re-synthesized time-domain signal of $\hat{X}_p$ obtained via inverse STFT (iSTFT), and similarly $\hat{x}_0^{\text{CTSE}} = \text{iSTFT}(\hat{X}_0^{\text{CTSE}})$. The filter is computed by solving the following problem:

$$\hat{h}_p = \underset{h_p}{\text{argmin}} \sum_n |\hat{x}_0^{\text{CTSE}}[n] - (h_p * \hat{x}_p)[n]|^2, \quad (8)$$

Zhong-Qiu Wang

where n indexes time-domain samples, $*$ denotes linear convolution, and $h_p \in \mathbb{R}^{K+1+K}$ denotes a $(K + 1 + K)$-tap linear filter with K past and K future taps. Equation (8) is also a quadratic linear regression problem, where a closed-form solution can be readily computed. After that, we compute the STFT spectrogram of the filtered time-domain DNN estimate

$$\hat{X}_{p \to 0} = \text{STFT}(\hat{h}_p * \hat{x}_p), \tag{9}$$

which is then plugged into Eq. (5) to compute the loss for training.

C. Addressing time-synchronization issues in pre-processiing

In Sec. V B, when performing linear filtering for loss computation, we assume that far-field and close-talk microphones are reasonably time-synchronized. In practical data collection, the close-talk microphone and far-field microphone array are usually managed and processed by two different devices. Although the microphones on each device are typically synchronized, there could be synchronization errors between the microphones on different devices. Later in Sec. VI B, we will slightly modify the classic GCC-PHAT algorithm (DiBiase *et al.*, 2001; Wang *et al.*, 2019) to roughly synchronize the close-talk microphone and far-field array. This technique is utilized as a pre-processing operation prior to training.

D. Co-learning on simulated and real mixtures

In practical application scenarios, the amount of real data is typically scarce, as it is often effort-consuming to collect real-recorded close-talk and far-field mixture pairs. In this case, even if ctPuLSEnet can be trained on real data, the performance is often limited, simply due to the limited amount of training data. On the other hand, simulated data can be easily and massively generated through simulation. In this case, following the SuperM2M algorithm (Wang, 2024b), which combines supervised learning on simulated data with unsupervised or weakly supervised learning on real data, we propose to train ctPuLSEnet on both simulated and real data. See Fig. 3 for an illustration, where supervised learning on simulated data is shown in Fig. 3(b).

Specifically, at each training step, we randomly sample either a mini-batch of real far-field mixtures and their pseudo-labels computed by CTSEnet or a mini-batch of simulated far-field mixtures (where the clean speech is available) to train the same DNN. If the sampled mini-batch is real, the loss function can be the $\mathcal{L}_{X,q}^{\text{real}}$ loss defined in Eq. (5), while if the sampled mini-batch is simulated, the loss can be $\mathcal{L}_{X,q}^{\text{simu}}$ in Eq. (3).

Alternatively, we can train ctPuLSEnet to not only predict target speech but also non-target signals via multi-task learning, as is described earlier in Sec. V A. In this case, the loss on simulated data can be

$$\mathcal{L}_q^{\text{simu}} = \mathcal{L}_{X,q}^{\text{simu}} + \mathcal{L}_{V,q}^{\text{simu}} + \mathcal{L}_{Y,q}, \tag{10}$$

where $\mathcal{L}_{V,q}^{\text{simu}}$ denotes the loss on estimated noise $\hat{V}_q$ and is defined, by following $\mathcal{L}_{X,q}^{\text{simu}}$ in Eq. (3), as

$$\mathcal{L}_{V,q}^{\text{simu}} = \frac{\sum_{t,f} \mathcal{F}(\hat{V}_q(t,f), V_q(t,f))}{\sum_{t',f'} |V_q(t',f')|}, \tag{11}$$

and $\mathcal{L}_{Y,q}$ is a mixture-constraint loss (Wang *et al.*, 2023) encouraging the speech and noise estimates to sum up to the observed mixture

$$\mathcal{L}_{Y,q} = \frac{\sum_{t,f} \mathcal{F}(\hat{X}_q(t,f) + \hat{V}_q(t,f), Y_q(t,f))}{\sum_{t',f'} |Y_q(t',f')|}. \tag{12}$$

Accordingly, the loss on real data can be modified to

$$\mathcal{L}_q^{\text{real}} = \mathcal{L}_{X,q}^{\text{real}} + \mathcal{L}_{Y,q}, \tag{13}$$

where $\mathcal{L}_{X,q}^{\text{real}}$ is defined in Eq. (5).

Notice that in each of the loss functions in Eqs. (3), (5), (11), and (12), a normalization term is used in the denominator to balance the loss with the others. This is particularly useful for $\mathcal{L}_X^{\text{real}}$ in Eq. (5), as the close-talk speech estimated based on the close-talk mixture (i.e., $\hat{X}_0^{\text{CTSE}}$) could have a gain level very different from far-field mixtures.

We can also add a weighting term $\alpha \in \mathbb{R}_{>0}$ between the loss on simulated data and the loss on real data to balance their importance. The overall loss is defined as

$$\mathcal{L}_q = \begin{cases} \alpha \times \mathcal{L}_q^{\text{simu}}, & \text{for mini-batches of simulated data,} \\ \mathcal{L}_q^{\text{real}}, & \text{for mini-batches of real data.} \end{cases} \tag{14}$$
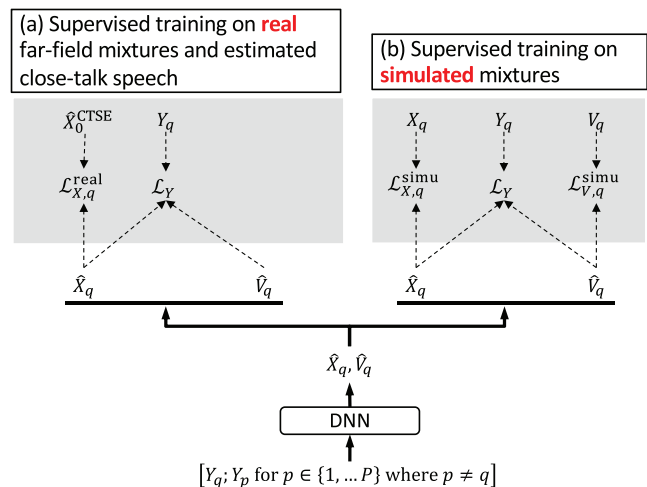

FIG. 3. Illustration of co-learning, which trains the same model via supervised learning based on (a) real-recorded far-field mixtures and their pseudo-labels derived from CTSEnet and (b) simulated mixtures.

J. Acoust. Soc. Am. **158** (4), October 2025

Zhong-Qiu Wang 2853

VI. EXPERIMENTAL SETUP

Based on real-recorded mixtures, the main objective of our experiments is to demonstrate whether ctPuLSEnet can achieve better enhancement on real-recorded far-field mixtures than purely supervised learning based approaches, which can only train enhancement models on simulated data. Bearing this objective in mind, we validate the proposed algorithms on the CHiME-4 dataset (Barker *et al.*, 2015; Vincent *et al.*, 2017; Barker *et al.*, 2017), the most popular benchmark to date in robust ASR and speech enhancement. It contains both real-recorded and simulated mixtures for both training and testing. Since, for real mixtures, the clean speech is not available for evaluation, we mainly check whether the enhanced speech can yield better ASR performance, considering that ASR scores can reflect speech intelligibility and indicate the degrees of speech distortion. The ASR evaluation pipeline is shown in Fig. 4, where enhanced close-talk or far-field speech is directly fed to a strong pre-trained ASR system for recognition.

The rest of this section describes the CHiME-4 dataset, synchronization of close-talk and far-field mixtures, setup for ASR evaluation, miscellaneous system configurations, baselines for comparison, and evaluation metrics.

A. CHiME-4 Dataset

CHiME-4 (Barker *et al.*, 2015; Vincent *et al.*, 2017; Barker *et al.*, 2017) is a major benchmark for evaluating far-field speech recognition and enhancement algorithms. The far-field recording device is a tablet mounted with six microphones, with the second microphone placed on the rear and the other five facing front. During data collection, the target speaker hand-holds the tablet and reads text prompts shown on the screen of the tablet. The target speaker wears a close-talk microphone so that a monaural close-talk mixture can be recorded along with each six-channel far-field mixture.

The mixtures are recorded in four representative daily environments (including buses, cafeteria, pedestrian areas, and streets), where multiple strong, non-stationary directional and diffuse noises can naturally co-exist. In CHiME-4, the room reverberation is weak, and the major challenge is in how to deal with the multi-source non-stationary noise signals.

The number of mixtures in CHiME-4 is summarized in Table I. In addition to real mixtures, CHiME-4 also provides simulated far-field mixtures for training and testing. We emphasize that, for each real mixture in the training set, in total it has seven channels (i.e., one close-talk plus six far-field microphones), while, for each simulated training mixture, it is far-field, six-channel, and does not have the paired close-talk mixture.

The sampling rate is 16 kHz.

B. Synchronization of close-talk and far-field mixtures

In CHiME-4, we observe that the far-field mixtures are reasonably synchronized with each other, while significant synchronization errors exist between close-talk and far-field

mixtures. For some utterances, the time delay between the close-talk and far-field mixtures can be as large as 0.05 s, which, if correct, means that the distance between the close-talk microphone and far-field array can be as large as $\sim 17\,\mathrm{m}$ (i.e., 0.05×340, assuming that the speed of sound in the air is 340 m per second). This clearly does not make sense in the CHiME-4 setup, as the speaker hand-holds the tablet while talking. The root cause of the synchronization errors is that the close-talk microphone and far-field array in CHiME-4 are placed on, and processed by, two different devices.

If the synchronization errors between the close-talk and far-field mixtures are not properly addressed before using them to train ctPuLSEnet, the loss functions in Eq. (5) would be less effective, since the hypothesized filter taps of $\mathbf{g}_p(f)$ in Eq. (7) or h_p in Eq. (8), which are hyperparameters shared by all the training utterances, may not be able to compensate the synchronization errors for every training utterance. To deal with this, we could use a very long linear filter to cover the maximum synchronization error of all the training mixtures. This is however problematic, as the longer the filter is, the more likely that the filter can filter any DNN estimate (even if the estimate is a random signal) to accurately fit the estimated close-talk speech.

To mitigate the synchronization errors, we slightly modify the classic GCC-PHAT algorithm (DiBiase *et al.*, 2001; Wang *et al.*, 2019) to approximately align the close-talk mixture to far-field mixtures. We emphasize that this is a pre-processing step at the very beginning of our proposed system (i.e., before training).

In detail, for each of the close-talk and far-field mixtures, we consider its magnitude in each frequency as a one-dimensional sequence, and find an integer frame shift $\hat{d} \in \mathbb{Z}$ (shared by all the frequencies) that can, at every frequency, best align the magnitude sequence of the close-talk mixture to those of the far-field mixtures. Specifically, let $M_p(f) = |Y_p(\cdot, f)| \in \mathbb{R}^T$ denote the magnitude sequence at frequency f for microphone $p \in \{0, 1, \ldots, P\}$, and $R_p(f) = \mathrm{FFT}(M_p(f)) \in \mathbb{C}^T$ denote the FFT coefficients after applying a T-point faster Fourier transform (FFT) to the magnitude sequence. At each frequency f, we first compute the GCC-PHAT coefficients between the magnitude sequence of the close-talk microphone and that of a far-field microphone p by

$$\mathrm{GCC\text{-}PHAT}_p(t, f, d)$$
$$= \mathcal{R}\left(\frac{R_0(t, f) \times R_p(t, f)^*}{|R_0(t, f)| \times |R_p(t, f)^*|} e^{-j \times 2\pi(t/T)d} \right)$$
$$= \cos\left(\angle R_0(t, f) - \angle R_p(t, f) - 2\pi \frac{t}{T} d \right), \quad (15)$$

where j denotes the imaginary unit, $d \in \mathbb{Z}$ a hypothesized frame delay, and $\mathcal{R}(\cdot)$ extracts the real component. We then enumerate a set of hypothesized frame delays and find a
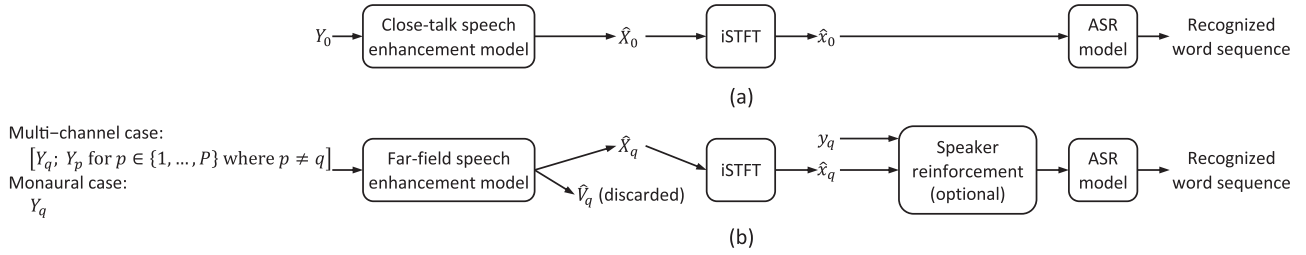
FIG. 4. Pipeline of ASR evaluation for (a) close-talk speech enhancement and (b) far-field speech enhancement. Enhanced speech [e.g., $\hat{x}_0 = \text{iSTFT}(\hat{X}_0)$ in close-talk speech enhancement] is directly fed to a pre-trained backend ASR system for recognition. No joint training between ASR and enhancement models is performed. In (b), an optional speaker reinforcement module (Zorilă and Doddipatla, 2022), which adds a scaled version of input mixture y_q to $\hat{s}_q$, can be included.

delay that can produce the largest summation of the GCC-PHAT coefficients at all the far-field microphones and T-F units

$$\hat{d} = \underset{d \in \Omega}{\arg\min} \sum_{p=1}^{P} \sum_{t=0}^{T-1} \sum_{f=0}^{F-1} \text{GCC-PHAT}_p(t,f,d), \qquad (16)$$

where $\Omega \in \{-D, \dots, 0, \dots, D\}$ is a set of candidate frame delays with D, a tunable hyper-parameter, denoting a hypothesized maximum delay on each side. If the resulting best time delay $\hat{d}$ is positive, we advance the close-talk mixture by $\hat{d}$ frames (and pad zeros to the right), and delay it by $\hat{d}$ frames otherwise (and pad zeros to the left).

Notice that this algorithm is designed to compute an integer number of frame shift for each close-talk mixture. In our experiments, for the STFT configuration of Eqs. (15) and (16), we set the window size to 16 ms and hop size to 1 ms (please do not confuse this STFT configuration used in the pre-processing stage with that in the subsequent DNN training stage). A small STFT hop size is used here, as our aim is to roughly align close-talk and far-field mixtures in this pre-processing stage.

We highlight that the alignment is performed at the granularity of 1 ms. Although sample-level synchronization is definitely desired, it would dramatically increase the amount of computation, as the enumeration would be conducted at the granularity of samples and this computation would become intolerable when the candidate time delay (and time advance) can be as large as 0.05 s in CHiME-4. In addition, we may not really need accurate sample-level synchronization, since the STFT window size of our enhancement models can be as large as 32 ms and the linear filtering in, e.g., Eq. (6) could account for slight synchronization errors (that are sufficiently smaller than the 32 ms window size).

C. Training setup

For CTSEnet, it is trained based on all the 7138×6 monaural simulated mixtures. Following Wang (2024b), we apply an SNR augmentation technique to the simulated training mixtures of CHiME-4. That is, during training, we optionally modify the SNR of each simulated mixture, on the fly, by u dB, with u uniformly sampled from the range $[-10, +15]$ dB. No other data augmentation is used.

For ctPuLSEnet, we use all the 7138 simulated and 1600 real utterances for training. For monaural ctPuLSEnet, we train it on all the $(7138 + 1600) \times 6$ monaural mixtures. For 2-channel ctPuLSEnet, at each training step we sample two microphones from the front five microphones as input, and ctPuLSEnet is trained to predict the target speech at the first of the two selected microphones. For 6-channel ctPuLSEnet, it stacks all the six microphones in a fixed order as input to predict the target speech at the fifth microphone. We apply the same SNR augmentation used in CTSEnet to the simulated training mixtures when training ctPuLSEnet. We always train ctPuLSEnet on the combination of the simulated and real mixtures using the co-learning algorithm introduced in Sec. V D, as there are only 1600 real mixtures (which amount to only ~ 2.7 h) in the training set of CHiME-4.

For simplicity, we did not filter out microphone signals with any microphone failures in training and inference. We expect ctPuLSEnet to learn to robustly deal with the failures, as it is trained on real mixtures.

D. Miscellaneous system configurations

For STFT, the window and hop sizes are, respectively, 32 and 8 ms, and the square root of Hann window is used as the analysis window.

TF-GridNet (Wang *et al.*, 2023) is employed as the DNN architecture. It has shown strong separation and

TABLE I. Number of utterances in CHiME-4.

Type	Close-talk or far-field?	#mics	Training set	Validation set	Test set
SIMU	Far-field	6	7138 (~ 15.1 h)	1640 (~ 2.9 h)	1320 (~ 2.3 h)
SIMU	Close-talk	—	N/A	N/A	N/A
REAL	Far-field	6	1600 (~ 2.7 h)	1640 (~ 2.7 h)	1320 (~ 2.2 h)
REAL	Close-talk	1	1600 (~ 2.7 h)	1640 (~ 2.7 h)	1320 (~ 2.2 h)

enhancement performance recently in several representative supervised speech separation benchmarks. Following the symbols defined in Table I of Wang *et al.* (2023), we set its hyper-parameters to $D = 128$, $B = 4$, $I = 1$, $J = 1$, $H = 200$, $L = 4$, and $E = 4$ (please do not confuse the symbols with the ones defined in this paper). The model has ~ 5.4 million trainable parameters. This configuration of TF-GridNet is used for both CTSEnet and ctPuLSEnet.

We train both CTSEnet and ctPuLSEnet on eight-second segments using a mini-batch size of one. Adam is used as the optimizer. The learning rate starts from 0.001 and is halved if the loss is not improved in two epochs.

E. ASR model

For both close-talk and far-field speech enhancement, we check whether they can result in better ASR performance by directly feeding their enhanced speech to a pre-trained ASR model for decoding, following the evaluation pipeline shown in Figs. 4(a) and 4(b).

The ASR model is pre-trained in a multi-conditional way on the official CHiME-4 simulated and real mixtures plus the clean speech signals in WSJ0 by using the script proposed in (Chang *et al.*, 2022), which is available in the ESPnet toolkit (Kamo, 2025). It is an encoder-decoder- and transformer-based system, trained on WavLM features (Chen *et al.*, 2022) and using a transformer-based language model for decoding. From the results reported in Chang *et al.* (2022) and Masuyama *et al.* (2023), this model is the current best ASR model on CHiME-4.

We have successfully reproduced the ASR system in Chang *et al.* (2022). The mixture ASR results (shown later in row 0 of Table III) are very close to the ones reported in row 7 of Table I of Chang *et al.* (2022).

F. Evaluation metrics

For real-recorded mixtures, where the corresponding clean speech is unavailable for evaluation, we use word error rates (WER) as the major evaluation metric. WER can partially reflect the intelligibility of enhanced speech. In addition to WER, DNSMOS (Reddy *et al.*, 2021) is employed to evaluate the quality of enhanced speech.

For simulated mixtures, where the clean speech is available, our evaluation metrics include short-time objective intelligibility (STOI) (Taal *et al.*, 2011), wideband perceptual evaluation of speech quality (wPESQ) (Rix *et al.*, 2001), signal-to-distortion ratio (SDR) (Le Roux *et al.*, 2019), and scale-invariant SDR (SISDR) (Vincent *et al.*, 2006). They are designed to evaluate the intelligibility, quality, and accuracy of the magnitude and phase of enhanced speech. They are widely adopted metrics in speech enhancement.

G. Baselines

A major baseline is purely supervised speech enhancement models trained only on the CHiME-4 simulated data. That is, we only use Fig. 3(b) for model training, and use

exactly the same training setup as that in ctPuLSEnet. On the other hand, since CHiME-4 is a popular public dataset, we can compare our ASR and enhancement results with the ones obtained by many earlier studies.

H. Tricks to improve ASR performance

At run time, in default we feed $\hat{x}_q$ produced by ctPuLSEnet for ASR decoding. Alternatively, following Wang (2024b) we apply an existing technique named speaker reinforcement (Zorilă and Doddipatla, 2022) to mitigate speech distortion incurred by enhancement [see Fig. 4(b) for an illustration]. It adds a scaled version of the mixture signal y_q back to enhanced speech $\hat{x}_q$ before performing ASR decoding. That is, the signal sent for ASR is $\hat{x}_q + \eta \times y_q$, where η is computed such that $10 \times \log_{10}(||\hat{x}_q||_2^2/||\eta \times y_q||_2^2) = \gamma$ dB. This technique has been found effective at improving ASR performance in Zorilă and Doddipatla (2022) and Wang (2024b).

VII. EVALUATION RESULTS

A. Results of close-talk speech enhancement

Table II presents the results of monaural CTSEnet.

Row 0 reports the results of unprocessed close-talk mixtures. We observe that the ASR results are very good (e.g., 1.49% on the real test set), indicating that the close-talk mixtures already have very high input SNRs and that microphone failures in the close-talk mixtures are minimal. In comparison, the DNSMOS scores are not good (e.g., 2.50 on the real test set), indicating that non-speech signals captured along with close-talk speech are still significant. These non-speech signals need to be removed in order to derive high-quality pseudo-labels for far-field mixtures.

Row 1 reports the results of CTSEnet trained via supervised learning on the CHiME-4 simulated mixtures (denoted as **S** in the "training data" column). Compared with row 0, the DNSMOS scores are clearly improved (e.g., from 2.50 to 3.22 on the real test set). The ASR performance becomes worse, possibly because of two reasons: (a) CTSEnet is trained only on simulated data, which usually mismatches real data and (b) CTSEnet could introduce some distortion to target speech while enhancing the mixture. Nonetheless, the ASR performance degrades rather slightly (e.g., from 1.49% to 1.56% WER on the real test set), meaning that the estimated close-talk speech is still of high quality and could be a good pseudo-label for far-field mixtures.

With a grain of salt, the ASR results in row 1 can be viewed as the upper-bound performance of ctPuLSEnet.

TABLE II. Results of CTSEnet on CHiME-4 close-talk mixtures.

| | | | DNSMOS OVRL ↑ | | WER (%)↓ | |
| | | | Val. | Test | Val. | Test |
Row	Training data	Systems	REAL	REAL	REAL	REAL
0	—	Close-talk mixture	2.82	2.50	1.14	1.49
1	S	CTSEnet	3.26	3.22	1.15	1.56

B. Key results of far-field speech enhancement

Tables III and IV, respectively, present the results of monaural and six-channel ctPuLSEnet on CHiME-4.

Let us first provide the hyper-parameter configurations of ctPuLSEnet. The frequency-domain linear filter in Eq. (6) is tuned to 1-tap [i.e., in the text below Eq. (6), $I = 1$ and $J = 0$]. In Eq. (8), the time-domain filter tap K is tuned based on the set of $\{256, 128, 64, 32, 16\}$. In this case, the linear filter length $K + 1 + K \in \{513, 257, 129, 65, 33\}$ is comparable to or shorter than the STFT window length, which is 512 samples long in this paper (see the STFT configurations in Sec. VI D). The weighting term α in Eq. (14) is tuned to 5. In both tables, when the filter tap for time-domain linear filtering (i.e., the "K" column) is denoted as "–," it means that frequency-domain linear filtering in Eq. (6) is used. Otherwise, time-domain linear filtering in Eq. (9) is used.

The key results of this paper are presented in row 0, 1a, and 3a of Tables III and IV. Comparing row 1a with 0, we observe that, on the simulated test data, purely supervised learning based TF-GridNet trained on simulated data obtains strong enhancement results[1] (e.g., 17.3 vs 7.5 dB SISDR in Table III and 22.9 vs 7.5 dB SISDR in Table IV), and strong ASR performance (e.g., 7.63% vs 8.29% WER in Table III and 1.34% vs 8.29% WER in Table IV). In addition, the enhancement results on the simulated test data obtained by TF-GridNet in row 1a are better than strong existing models such as iNeuBe (Wang *et al.*, 2020; Lu *et al.*, 2022), SpatialNet (Quan and Li, 2024), USES (Zhang *et al.*, 2023), and USES2 (Zhang *et al.*, 2024). However, the performance on the real test data is limited. For example, the ASR performance on the real test data is degraded compared to just using unprocessed mixtures for ASR decoding (e.g., 4.47% vs 5.24% WER in Table III). This degradation is much more severe in multi-channel cases (e.g., 4.47% vs 46.71% WER

in Table IV), possibly because simulated inter-microphone characteristics are more likely to mismatch those in real-recorded multi-channel data, compared with monaural cases where only one microphone needs to be simulated. These problems are widely observed in earlier robust ASR studies (Haeb-Umbach *et al.*, 2019; Haeb-Umbach *et al.*, 2021), largely because (a) on real data, enhancement models trained on simulated data usually introduce speech distortion detrimental to ASR and (b) simulated training data is often mismatched with real-recorded test data. In row 3a, our proposed ctPuLSEnet, trained on simulated and real data combined (denoted as $\mathbf{S} + \mathbf{R}$ in the "training data" column), obtains clearly better ASR performance on the real test set over row 1a and 0 (e.g., 3.23% vs 5.24% and 4.47% in Table III and 1.65% vs 46.71% and 4.47% in Table IV). These results indicate that ctPuLSE is an effective mechanism for learning from real-recorded data, and can yield enhancement models with better generalizability to real data than purely supervised approaches which train enhancement models only on simulated data.

C. Ablation results of far-field speech enhancement

Next, we present several ablation results of ctPuLSEnet.

Comparing row 3a with 3b and 3c in Table III, we observe that configuring ctPuLSEnet to estimate noise besides target speech and at the same time including both the loss on the noise estimate and the mixture-constraint loss (i.e., row 3a) lead to better enhancement and ASR performance. Comparing 3a with 1a, 3b with 1b, and 3c with 1c, we observe that ctPuLSE obtains better ASR performance on the real test data for the various loss functions used for the simulated data.

In 4a–4e of Table III, we switch from frequency-domain linear filtering to time-domain linear filtering, and experiment with various filter lengths by tuning K [defined

TABLE III. ctPuLSE vs purely supervised models on CHiME-4 far-field mixtures (single-channel input).

Row	Systems	Training data	DNN arch.	$\mathcal{L}_q^{simu}$	$\mathcal{L}_q^{real}$	K	SISDR (dB)↑	SDR (dB)↑	wPESQ↑	STOI↑	DNSMOS OVRL↑ Val. REAL	DNSMOS OVRL↑ Test REAL	WER (%)↓ Val. SIMU	WER (%)↓ Val. REAL	WER (%)↓ Test SIMU	WER (%)↓ Test REAL
0	Mixture	—	—	—	—	—	7.5	7.5	1.27	0.870	1.52	1.39	5.93	4.07	8.29	4.47
1a	Supervised	S	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	—	—	**17.3**	17.7	2.36	0.960	3.33	3.21	3.49	2.18	7.63	5.24
1b	Supervised	S	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu}$	—	—	17.2	17.6	**2.49**	0.962	3.32	3.21	3.39	2.09	7.41	4.44
1c	Supervised	S	TF-GridNet	$\mathcal{L}_{X,q}^{simu}$	—	—	**17.3**	17.6	2.43	**0.963**	3.30	3.19	3.37	2.24	7.21	5.17
2	Supervised	S	iNeuBe (Wang *et al.*, 2020)	—	—	—	15.1	—	—	0.954	—	—	—	—	—	—
3a	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	$\mathcal{L}_{X,q}^{real} + \mathcal{L}_{Y,q}$	—	17.1	17.6	2.41	0.962	3.36	3.23	**3.16**	2.00	**6.84**	3.23
3b	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu}$	$\mathcal{L}_{X,q}^{real}$	—	16.7	17.3	2.41	0.960	3.32	3.16	3.56	2.08	6.85	3.56
3c	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu}$	$\mathcal{L}_{X,q}^{real}$	—	16.8	17.3	2.39	0.959	3.35	3.20	3.72	2.07	7.64	3.78
4a	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	$\mathcal{L}_{X,q}^{real} + \mathcal{L}_{Y,q}$	256	16.7	17.2	2.29	0.958	3.33	3.19	3.21	1.98	7.06	3.41
4b	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	$\mathcal{L}_{X,q}^{real} + \mathcal{L}_{Y,q}$	128	16.5	17.1	2.37	0.960	3.35	3.22	3.41	1.88	7.08	3.20
4c	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	$\mathcal{L}_{X,q}^{real} + \mathcal{L}_{Y,q}$	64	17.0	17.5	2.40	0.961	**3.38**	**3.27**	3.29	**1.87**	6.97	3.13
4d	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	$\mathcal{L}_{X,q}^{real} + \mathcal{L}_{Y,q}$	32	16.5	17.1	2.36	0.961	3.35	3.22	3.38	1.99	6.96	3.55
4e	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	$\mathcal{L}_{X,q}^{real} + \mathcal{L}_{Y,q}$	16	16.1	16.9	2.36	0.959	3.36	3.24	3.44	1.97	7.23	**3.04**

TABLE IV. ctPuLSE vs purely supervised models on CHiME-4 far-field mixtures (six-channel input).

Row	Systems	Training data	DNN arch.	$\mathcal{L}_q^{simu}$	$\mathcal{L}_q^{real}$	K	SISDR (dB)↑	SDR (dB)↑	wPESQ↑	STOI↑	DNSMOS OVRL↑ Val. REAL	DNSMOS OVRL↑ Test REAL	WER (%)↓ Val. SIMU	WER (%)↓ Val. REAL	WER (%)↓ Test SIMU	WER (%)↓ Test REAL
0	Mixture	—	—	—	—	—	7.5	7.5	1.27	0.870	1.52	1.39	5.93	4.07	8.29	4.47
1a	Supervised	S	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	—	—	22.9	**23.3**	**3.34**	0.987	2.12	1.78	0.86	23.16	1.34	46.71
1b	Supervised	S	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu}$	—	—	**23.0**	23.3	3.29	**0.988**	2.06	1.63	0.86	40.27	1.32	64.71
1c	Supervised	S	TF-GridNet	$\mathcal{L}_{X,q}^{simu}$	—	—	22.8	23.2	3.22	0.987	2.04	1.63	**0.83**	53.54	**1.30**	74.22
2a	Supervised	S	iNeuBe (Wang et al., 2020)	—	—	—	22.0	22.4	-	0.986	—	—	—	—	—	—
2b	Supervised	S	SpatialNet (Quan and Li, 2024)	—	—	—	22.1	22.3	2.88	0.983	—	—	—	—	—	—
2c	Supervised	S	USES (Zhang et al., 2023)	—	—	—	—	20.6	3.16	0.983	—	—	—	—	4.20	78.10
2d	Supervised	S	USES2 (Zhang et al., 2024)	—	—	—	—	18.8	2.94	0.979	—	2.96	—	—	4.60	12.10
3a	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	$\mathcal{L}_{X,q}^{real} + \mathcal{L}_{Y,q}$	—	22.6	22.8	3.11	0.985	3.38	3.28	0.82	1.28	1.39	1.65
3b	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu}$	$\mathcal{L}_{X,q}^{real}$	—	22.8	23.3	3.26	0.987	**3.39**	3.28	0.87	1.27	1.35	1.73
3c	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu}$	$\mathcal{L}_{X,q}^{real}$	—	22.8	23.1	3.29	0.987	**3.39**	**3.30**	0.84	1.29	**1.30**	**1.56**
4a	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	$\mathcal{L}_{X,q}^{real} + \mathcal{L}_{Y,q}$	256	22.6	23.0	3.11	0.986	3.37	3.25	0.86	1.31	1.34	1.77
4b	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	$\mathcal{L}_{X,q}^{real} + \mathcal{L}_{Y,q}$	128	22.9	23.2	3.22	0.986	3.34	3.25	0.84	1.32	1.34	1.70
4c	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	$\mathcal{L}_{X,q}^{real} + \mathcal{L}_{Y,q}$	64	22.8	23.2	3.20	0.986	**3.39**	3.29	0.86	**1.25**	1.37	1.70
4d	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	$\mathcal{L}_{X,q}^{real} + \mathcal{L}_{Y,q}$	32	22.7	23.1	3.23	0.986	**3.39**	**3.30**	0.87	1.28	1.31	1.67
4e	ctPuLSE	S + R	TF-GridNet	$\mathcal{L}_{X,q}^{simu} + \mathcal{L}_{V,q}^{simu} + \mathcal{L}_{Y,q}$	$\mathcal{L}_{X,q}^{real} + \mathcal{L}_{Y,q}$	16	22.6	23.0	3.25	0.985	3.38	3.27	0.89	1.26	1.42	1.80

in the text below Eq. (8)] based on the set of $\{256, 128, 64, 32, 16\}$. We observe that setting K to 64 (in row 4c) produces the best ASR performance on the real validation set among various options, and the ASR performance is also better than row 3a on the real validation set (i.e., 1.87% vs 2.00% WER in Table III). Similar trend is also observed in the six-channel-input case in Table IV. We therefore choose the setup in row 4c for the rest of experiments in this paper.

D. More ablation results of far-field speech enhancement

This subsection provides more ablation results of ctPuLSE to show the effectiveness of some key hyper-parameters we use for training, including (a) whether pre-training the DNN in ctPuLSE on simulated mixtures, (b) tuning the batch size used for training ctPuLSE, and (c) tuning the weighting term for the loss on simulated mixtures [i.e., α in Eq. (14)]. The results are obtained based on the 6-channel system configuration in row 4c of Table IV.

1. With or without pre-training on simulated mixtures

In default, we train the DNN in ctPuLSE from scratch via co-learning based on simulated and real mixtures. Another strategy is to first pre-train the DNN via supervised learning on simulated mixtures and then fine-tune the DNN on real mixtures or on both simulated and real mixtures. We compare their performance in rows 4c, 5a, and 5b of

TABLE V. Results of ctPuLSE with and without supervised pre-training (based on simulated training mixtures) on CHiME-4 far-field mixtures (six-channel input).

Row	Systems	Training data	DNSMOS OVRL↑ Val. REAL	DNSMOS OVRL↑ Test REAL	WER (%)↓ Val. SIMU	WER (%)↓ Val. REAL	WER (%)↓ Test SIMU	WER (%)↓ Test REAL
0	Mixture	—	1.52	1.39	5.93	4.07	8.29	4.47
1a of Table IV	Supervised	S	2.12	1.78	0.86	23.16	1.34	46.71
4c of Table IV	ctPuLSE	S + R	**3.39**	**3.29**	**0.86**	**1.25**	1.37	1.70
5a	Supervised → ctPuLSE	S → S + R	3.37	3.25	1.08	1.28	2.06	**1.66**
5b	Supervised → ctPuLSE	S → R	3.35	3.22	0.87	1.28	**1.29**	1.77

TABLE VI. Results of ctPuLSE trained with different batch sizes on CHiME-4 far-field mixtures (six-channel input).

| Row | Systems | Batch size | DNSMOS OVRL ↑ | | WER (%)↓ | | | |
| | | | Val. REAL | Test REAL | Val. | | Test | |
					SIMU	REAL	SIMU	REAL
0	Mixture	—	1.52	1.39	5.93	4.07	8.29	4.47
4c of Table IV	ctPuLSE	1	**3.39**	**3.29**	0.86	**1.25**	1.37	1.70
6a	ctPuLSE	2	**3.39**	**3.29**	**0.83**	1.28	1.37	**1.65**
6b	ctPuLSE	4	3.37	3.25	0.89	1.26	**1.34**	1.71
6c	ctPuLSE	6	3.38	3.25	0.88	1.33	1.40	1.73

Table V. We observe that the fine-tuning approach does not produce better WER on the REAL validation set than the default approach.

2. Effects of batch size

In default, the mini-batch size is set to one for model training. In Table VI, we report the results of ctPuLSE when the mini-batch size is configured larger than one. From the results, we do not observe large difference in performance, and setting the mini-batch size to one yields the best WER on the REAL validation set.

3. Weighting term between losses on simulated and real mixtures

In default, this paper sets the weighting term between the losses on simulated and real mixtures [i.e., α in Eq. (14)] to 5. In Table VII, we report the results of enumerating α in the set of $\{10, 5, 2.5, 1.0, 0.5, 0.25, 0.1\}$. From the results, we only observe slight fluctuation in performance and setting α to 5 produces strong performance among all the values.

Notice that there are 7138 simulated mixtures and 1600 real mixtures in the training set. That is, there are many more simulated mixtures than real mixtures. In this case, we however do not observe too much performance difference even if setting α to a value larger than one. This is possibly because the loss value on simulated mixtures is much smaller than that on real mixtures, as the target signal for real mixtures is much harder to fit.

E. Comparison with other approaches

In Table VIII, we compare the performance of ctPuLSEnet with IRIS (Chang *et al.*, 2022), multi-channel IRIS (MultiIRIS) (Masuyama *et al.*, 2023), and SuperM2M (Wang, 2024b) on the 1-, 2-, and 6-channel tasks of CHiME-4.

IRIS and MultiIRIS were the state-of-the-art systems on CHiME-4 before SuperM2M. Comparing row 1a with 1b, 4a with 4b, and 7a with 7b, we observe that IRIS and MultiIRIS need joint frontend-backend training to achieve strong ASR performance, especially in the one- and two-channel cases.

SuperM2M (Wang, 2024b), even without joint frontend-backend training, achieves better ASR performance on the real data than IRIS and MultiIRIS. However, the enhancement results (measured by DNSMOS OVRL) of SuperM2M on the real data are not strong. Upon listening to its processed signals [refer to a sound demo in Wang (2025), which provides a comparison between SuperM2M and ctPuLSE], we observe that it cannot suppress noises sufficiently and tends to maintain some noise signals in its estimate of target speech, especially in multi-channel cases. This is reflected by the DNSMOS OVRL scores in rows 2, 5, and 8, and it is quite counter-intuitive that the score decreases when the number of input microphones increases (e.g., from 3.03 in the monaural case down to 2.90 in the 2-channel case and down to 2.75 in the 6-channel case on the real test data). We think that the mediocre enhancement score is likely because, in SuperM2M (Wang, 2024b), the

TABLE VII. Results of ctPuLSE trained with different weighting term α [for the loss in Eq. (14)] on CHiME-4 far-field mixtures (six-channel input).

| Row | Systems | α in Eq. (14) | DNSMOS OVRL ↑ | | WER (%)↓ | | | |
| | | | Val. REAL | Test REAL | Val. | | Test | |
					SIMU	REAL	SIMU	REAL
0	Mixture	—	1.52	1.39	5.93	4.07	8.29	4.47
4c of Table IV	ctPuLSE	5	3.39	3.29	**0.86**	1.25	1.37	1.70
7a	ctPuLSE	10	3.40	**3.30**	**0.86**	1.26	1.36	1.71
7b	ctPuLSE	2.5	**3.41**	**3.30**	0.87	1.28	**1.31**	**1.61**
7c	ctPuLSE	1.0	3.40	3.29	0.87	1.25	1.32	1.73
7d	ctPuLSE	0.5	3.39	3.29	0.87	**1.24**	1.38	1.69
7e	ctPuLSE	0.25	3.39	3.28	0.89	1.27	1.44	1.71
7f	ctPuLSE	0.1	3.37	3.27	0.98	1.36	1.50	1.81

TABLE VIII. Comparison of ctPuLSE with other approaches on CHiME-4 far-field mixtures. The best scores are highlighted in bold in each of the one-, two-, and six-channel tasks separately. SuperM2M and ctPuLSE use exactly the same TF-GridNet architecture.

| | | | | | DNSMOS OVRL ↑ | | WER (%)↓ | | | |
| | | | | | Val. REAL | Test REAL | Val. SIMU | Test REAL | SIMU | REAL |
Row	Systems	Frontend	Joint training?	#input mics						
0	Mixture	—	—	1	1.52	1.39	5.93	4.07	8.29	4.47
1a	IRIS (Chang *et al.*, 2022)	Conv-TasNet	✗	1	—	—	5.96	4.37	13.52	12.11
1b	IRIS (Chang *et al.*, 2022)	Conv-TasNet	✓	1	—	—	**3.16**	2.03	**6.12**	3.92
2	SuperM2M (Wang, 2024b)	TF-GridNet	✗	1	3.16	3.03	3.39	**1.84**	6.57	**3.04**
3	ctPuLSE (4c of Table III)	TF-GridNet	✗	1	**3.38**	**3.27**	3.29	1.87	6.97	3.13
4a	MultiIRIS (Masuyama *et al.*, 2023)	Neural WPD	✗	2	—	—	2.28	2.06	2.30	3.63
4b	MultiIRIS (Masuyama *et al.*, 2023)	Neural WPD	✓	2	—	—	2.04	1.66	**2.04**	2.65
5	SuperM2M (Wang, 2024b)	TF-GridNet	✗	2	3.01	2.90	**1.50**	**1.40**	2.08	**1.94**
6	ctPuLSE (4c of Table IV)	TF-GridNet	✗	2	**3.40**	**3.31**	1.57	1.45	2.33	2.08
7a	MultiIRIS (Masuyama *et al.*, 2023)	Neural WPD	✗	6	—	—	1.19	1.32	1.29	1.85
7b	MultiIRIS (Masuyama *et al.*, 2023)	Neural WPD	✓	6	—	—	1.22	1.33	**1.24**	1.77
8	SuperM2M (Wang, 2024b)	TF-GridNet	✗	6	2.84	2.75	**0.83**	1.26	1.37	**1.61**
9	ctPuLSE (4c of Table IV)	TF-GridNet	✗	6	**3.39**	**3.29**	0.86	**1.25**	1.37	1.70

loss function is defined on observed mixtures (i.e., between each observed mixture and reconstructed mixture, which is obtained by summing up linearly filtered source estimates) rather than on individual source estimates. In this case, the DNN would only have a *weak supervision* regarding what the target sources are. That is, the cues leveraged for training the SuperM2M enhancement models are (a) there should be two sources and (b) their estimates, after linear filtering, should add up to each mixture. Such a supervision could be too weak for the DNN to achieve good enhancement.

In comparison, in ctPuLSE, the loss function includes a loss term on the pseudo-target speech provided by CTSEnet [i.e., $\mathcal{L}_X^{\mathrm{real}}$ in Eq. (5)]. As long as the pseudo-target is reasonably accurate, ctPuLSE is expected to more accurately suppress non-target signals. Comparing row 3 with 2, 6 with 5,

and 9 with 8, we observe that ctPuLSE indeed obtains dramatically better DNSMOS OVRL scores than SuperM2M, and the enhanced target speech sounds much cleaner (please see the sound demo mentioned at the end of the Introduction). Although the ASR performance on the real test data is worse, it is only slightly worse and is still very strong.

F. Miscellaneous results

Table IX reports the results of applying speaker reinforcement (see the details in Sec. VI H), where the SNR factor γ is tuned to 10 dB. We observe that the performance gap between SuperM2M and ctPuLSE observed in Table VIII is reduced, especially on the two- and six-channel tasks. Our best performing system in row 6 b obtains ASR results competitive to using

TABLE IX. Effects of speaker reinforcement *Note*: Oracle results (i.e., directly using close-talk mixtures for ASR decoding) are marked in gray.

| | | | | WER (%)↓ | | | |
| | | | | Val. | | Test | |
Row	Systems	#input mics	Speaker reinforcement γ (dB)	SIMU	REAL	SIMU	REAL
0	Mixture	1	—	5.93	4.07	8.29	4.47
1a	SuperM2M (Wang, 2024b)	1	—	3.39	1.84	6.57	3.04
1b	SuperM2M (Wang, 2024b)	1	10	2.40	**1.64**	**4.54**	**2.40**
2a	ctPuLSE	1	—	3.29	1.87	6.97	3.13
2b	ctPuLSE	1	10	**2.29**	1.67	4.76	2.55
3a	SuperM2M (Wang, 2024b)	2	—	1.50	1.40	2.08	1.94
3b	SuperM2M (Wang, 2024b)	2	10	**1.28**	**1.33**	**1.88**	**1.84**
4a	ctPuLSE	2	—	1.57	1.45	2.33	2.08
4b	ctPuLSE	2	10	1.45	1.36	2.00	1.85
5a	SuperM2M (Wang, 2024b)	6	—	**0.83**	1.26	**1.37**	1.61
5b	SuperM2M (Wang, 2024b)	6	10	**0.83**	1.23	**1.37**	1.58
6a	ctPuLSE	6	—	0.86	1.25	**1.37**	1.70
6b	ctPuLSE	6	10	0.86	**1.22**	**1.37**	**1.56**
7	Close-talk Mixture	—	—	—	1.14	—	1.49

close-talk mixtures for ASR decoding, suggesting the effectiveness of ctPuLSE and the overall robust ASR system.

VIII. LIMITATIONS

We emphasize that the performance of ctPuLSE depends on the quality of CTSE. Our study builds upon the assumption that CTSE can produce reasonably good enhancement to close-talk mixtures so that reasonably good pseudo-labels can be derived for far-field mixtures. This assumption, in many cases, can be satisfied as close-talk mixtures usually have a very high input SNR already. In cases when the input SNR is low, one could leverage directional microphones to record close-talk speech during data collection. Another strategy is to leverage techniques such as crosstalk reduction (Wang *et al.*, 2024) to train a DNN to estimate the close-talk speech based on close-talk mixtures.

IX. CONCLUSION

We have investigated close-talk speech enhancement, and have proposed a novel approach, ctPuLSE, for far-field speech enhancement, where estimated close-talk speech produced by close-talk speech enhancement is utilized as pseudo-labels for training supervised enhancement models directly on real far-field mixtures to realize better generalizability to real data. Evaluation results on the challenging CHiME-4 dataset show the effectiveness and potential of the proposed algorithms.

Although simple and straightforward, the proposed approach of exploiting close-talk mixtures for far-field speech enhancement, we think, could encourage a new stream of research towards realizing neural speech enhancement models that can generalize better to real data, as it suggests a promising way that can derive, for real mixtures, pseudo-labels which can enable the training of speech enhancement models directly on real mixtures. Thanks to the innate high input SNR of close-talk mixtures, the derived pseudo-labels are often reliable and high-quality. This paper, based on the challenging CHiME-4 dataset, has shown that the derived pseudo-labels can be utilized to build far-field speech enhancement models with better generalizability to real data. Looking forward, we expect the derived pseudo-labels to be also useful in many other applications beyond far-field speech enhancement.

AUTHOR DECLARATIONS
Conflict of Interest

The author has no conflicts to disclose.

DATA AVAILABILITY

The data supporting the findings of this study are available at https://catalog.ldc.upenn.edu/LDC2017S24.

---

[1]By "enhancement results," we mean SISDR, SDR, wPESQ, and STOI scores.

Araki, S., Ito, N., Haeb-Umbach, R., Wichern, G., Wang, Z.-Q., and Mitsufuji, Y. (**2025**). "30+ years of source separation research: Achievements and future challenges," in *Proceedings of ICASSP*.

Barker, J., Marxer, R., Vincent, E., and Watanabe, S. (**2015**). "The third 'CHiME' speech separation and recognition challenge: Dataset, task and baselines," in *Proceedings of ASRU*, pp. 504–511.

Barker, J., Marxer, R., Vincent, E., and Watanabe, S. (**2017**). "The third 'CHiME' speech separation and recognition challenge: Analysis and outcomes," Comput. Speech Lang. **46**, 605–626.

Boeddeker, C., Zhang, W., Nakatani, T., Kinoshita, K., Ochiai, T., Delcroix, M., Kamo, N., Qian, Y., and Haeb-Umbach, R. (**2021**). "Convolutional transfer function invariant SDR training criteria for multi-channel reverberant speech separation," in *Proceedings of ICASSP*, pp. 8428–8432.

Chang, X., Maekaku, T., Fujita, Y., and Watanabe, S. (**2022**). "End-to-end integration of speech recognition, speech enhancement, and self-supervised learning representation," in *Proceedings of Interspeech*, pp. 3819–3823.

Chen, J., and Wang, D. (**2017**). "Long short-term memory for speaker generalization in supervised speech separation," J. Acoust. Soc. Am. **141**, 4705–4714.

Chen, J., Wang, Y., Yoho, S. E., Wang, D., and Healy, E. W. (**2016**). "Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises," J. Acoust. Soc. Am. **139**(5), 2604–2612.

Chen, S., Wang, C., Chen, Z., Wu, Y., Liu, S., Chen, Z., Li, J., Kanda, N., Yoshioka, T., Xiao, X., Wu, J., Zhou, L., Ren, S., Qian, Y., Qian, Y., Wu, J., Zeng, M., Yu, X., and Wei, F. (**2022**). "WavLM: Large-scale self-supervised pre-training for full stack speech processing," IEEE J. Sel. Top. Signal Process. **16**(6), 1505–1518.

Cornell, S., Wiesner, M. S., Watanabe, S., Raj, D., Chang, X., Garcia, P., Maciejewski, M., Masuyama, Y., Wang, Z.-Q., Squartini, S., and Khudanpur, S. (**2023**). "The CHiME-7 DASR challenge: Distant meeting transcription with multiple devices in diverse scenarios," in *Proceedings of CHiME*, pp. 1–6.

Cox, T. J., Barker, J., Bailey, W., Graetzer, S., Akeroyd, M. A., Culling, J. F., and Naylor, G. (**2023**). "Overview of the 2023 ICASSP SP clarity challenge: Speech enhancement for hearing aids," in *Proceedings of ICASSP*.

DiBiase, J., Silverman, H., and Brandstein, M. (**2001**). "Robust localization in reverberant rooms," in *Microphone Arrays* (Springer, Berlin, Germany), pp. 157–180.

Frenkel, L., E., Chazan, S., and Goldberger, J. (**2024**). "Domain adaptation using suitable pseudo labels for speech enhancement and dereverberation," IEEE/ACM Trans. Audio. Speech. Lang. Process. **32**, 1226–1236.

Haeb-Umbach, R., Heymann, J., Drude, L., Watanabe, S., Delcroix, M., and Nakatani, T. (**2021**). "Far-field automatic speech recognition," Proc. IEEE **109**(2), 124–148.

Haeb-Umbach, R., Watanabe, S., Nakatani, T., Bacchiani, M., Hoffmeister, B., Seltzer, M. L., Zen, H., and Souden, M. (**2019**). "Speech processing for digital home assistants: Combining signal processing with deep-learning techniques," IEEE Signal Process. Mag. **36**(6), 111–124.

Jiang, Y., Lu, X., Zu, Y., and Zhou, H. (**2013**). "Classification-based close talk speech enhancement," in *International Conference on Consumer Electronics, Communications and Networks*, pp. 192–195.

Jiang, Y., Zu, Y., and Liu, R. (**2016**). "Auditory features for the close talk speech enhancement with parameter masks," in *International Congress on Image and Signal Processing*, pp. 1194–1198.

Kamo, N. (**2025**). "ESPnet toolkit," https://github.com/espnet/espnet/blob/master/egs2/chime4/asr1/conf/tuning/train_asr_transformer_wavlm_lr1e-3_specaug_accum1_preenc128_warmup20k.yaml (Last viewed October 8, 2025).

Leglaive, S., Borne, L., Tzinis, E., Sadeghi, M., Fraticelli, M., Wisdom, S., Pariente, M., Pressnitzer, D., and Hershey, J. R. (**2023**). "The CHiME-7 UDASE task: Unsupervised domain adaptation for conversational speech enhancement," in *CHiME*, pp. 7–12.

Le Roux, J., Wisdom, S., Erdogan, H., and Hershey, J. R. (**2019**). "SDR—Half-baked or well done?," in *Proceedings of ICASSP*, pp. 626–630.

Lu, Y.-J., Cornell, S., Chang, X., Zhang, W., Li, C., Ni, Z., Wang, Z.-Q., and Watanabe, S. (**2022**). "Towards low-distortion multi-channel speech enhancement: The ESPNet-SE submission to the L3DAS22 challenge," in *Proceedings of ICASSP*, pp. 9201–9205.

Masuyama, Y., Chang, X., Cornell, S., Watanabe, S., and Ono, N. (**2023**). "End-to-end integration of speech recognition, dereverberation, beamforming, and self-supervised learning representation," in *Proceedings of SLT*, pp. 260–265.

Mysore, G. (**2015**). "Can we automatically transform speech recorded on common consumer devices in real-world environments into professional production quality speech?—A dataset, insights, and challenges," IEEE Signal Process. Lett. **22**(8), 1006–1010.

Pandey, A., and Wang, D. (**2020**). "On cross-corpus generalization of deep learning based speech enhancement," IEEE/ACM Trans. Audio. Speech. Lang. Process. **28**, 2489–2499.

Quan, C., and Li, X. (**2024**). "SpatialNet: Extensively learning spatial information for multichannel joint speech separation, denoising and dereverberation," IEEE/ACM Trans. Audio. Speech. Lang. Process. **32**, 1310–1323.

Reddy, C. K. A., Gopal, V., and Cutler, R. (**2021**). "DNSMOS: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors," in *Proceedings of ICASSP*, pp. 6493–6497.

Rix, A., Beerends, J., Hollier, M., and Hekstra, A. (**2001**). "Perceptual evaluation of speech quality (PESQ)—A new method for speech quality assessment of telephone networks and codecs," in *Proceedings of ICASSP*, Vol. 2, pp. 749–752.

Taal, C. H., Hendriks, R. C., Heusdens, R., and Jensen, J. (**2011**). "An algorithm for intelligibility prediction of time-frequency weighted noisy speech," IEEE Trans. Audio. Speech. Lang. Process. **19**(7), 2125–2136.

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

Tan, K., Wang, Z.-Q., and Wang, D. (**2022**). "Neural spectrospatial filtering," IEEE/ACM Trans. Audio. Speech. Lang. Process. **30**, 605–621.

Tan, K., Zhang, X., and Wang, D. (**2021**). "Deep learning based real-time speech enhancement for dual-microphone mobile phones," IEEE/ACM Trans. Audio. Speech. Lang. Process. **29**, 1853–1863.

Tzinis, E., Adi, Y., Ithapu, V. K., Xu, B., Smaragdis, P., and Kumar, A. (**2022a**). "RemixIT: Continual self-training of speech enhancement models via bootstrapped remixing," IEEE J. Sel. Top. Signal Process. **16**(6), 1329–1341.

Tzinis, E., Wisdom, S., Remez, T., R., and Hershey, J. (**2022b**). "AudioScopeV2: Audio-visual attention architectures for calibrated open-domain on-screen sound separation," in *Proceedings of ECCV*, pp. 368–385.

Vincent, E., Gribonval, R., and Févotte, C. (**2006**). "Performance measurement in blind audio source separation," IEEE Trans. Audio. Speech Lang. Process. **14**(4), 1462–1469.

Vincent, E., Watanabe, S., Nugraha, A. A., Barker, J., and Marxer, R. (**2017**). "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," Comput. Speech Lang. **46**, 535–557.

Wang, D., and Chen, J. (**2018**). "Supervised speech separation based on deep learning: An overview," IEEE/ACM Trans. Audio. Speech. Lang. Process. **26**, 1702–1726.

Wang, Z.-Q. (**2024a**). "Mixture to mixture: Leveraging close-talk mixtures as weak-supervision for speech separation," IEEE Signal Process. Lett. **31**, 1715–1719.

Wang, Z.-Q. (**2024b**). "SuperM2M: Supervised and mixture-to-mixture co-learning for speech enhancement and robust ASR," arXiv:2403.10271.

Wang, Z.-Q. (**2025**). "Sound demo of ctPuLSE on CHiME-4 corpus," https://zqwang7.github.io/demos/ctPuLSE_demo/index.html (Last viewed October 8, 2025).

Wang, Z.-Q., Cornell, S., Choi, S., Lee, Y., Kim, B.-Y., and Watanabe, S. (**2023**). "TF-GridNet: Integrating full- and sub-band modeling for speech separation," IEEE/ACM Trans. Audio. Speech. Lang. Process. **31**, 3221–3236.

Wang, Z.-Q., Kumar, A., and Watanabe, S. (**2024**). "Cross-talk reduction," in *Proceedings of IJCAI*, pp. 5171–5180.

Wang, Z.-Q., and Wang, D. (**2020**). "Multi-microphone complex spectral mapping for speech dereverberation," in *Proceedings of ICASSP*, pp. 486–490.

Wang, Z.-Q., Wang, P., and Wang, D. (**2020**). "Complex spectral mapping for single- and multi-channel speech enhancement and robust ASR," IEEE/ACM Trans. Audio. Speech. Lang. Process. **28**, 1778–1787.

Wang, Z.-Q., Wang, P., and Wang, D. (**2021a**). "Multi-microphone complex spectral mapping for utterance-wise and continuous speech separation," IEEE/ACM Trans. Audio. Speech. Lang. Process. **29**, 2001–2014.

Wang, Z.-Q., and Watanabe, S. (**2023**). "UNSSOR: Unsupervised neural speech separation by leveraging over-determined training mixtures," in *NeurIPS*, pp. 34021–34042.

Wang, Z.-Q., Wichern, G., and Le Roux, J. (**2021b**). "Convolutive prediction for monaural speech dereverberation and noisy-reverberant speaker separation," IEEE/ACM Trans. Audio. Speech. Lang. Process. **29**, 3476–3490.

Wang, Z.-Q., Wichern, G., and Le Roux, J. (**2021c**). "On the compensation between magnitude and phase in speech separation," IEEE Signal Process. Lett. **28**, 2018–2022.

Wang, Z.-Q., Zhang, X., and Wang, D. (**2019**). "Robust speaker localization guided by deep learning based time-frequency masking," IEEE/ACM Trans. Audio. Speech. Lang. Process. **27**(1), 178–188.

Watanabe, S., Mandel, M., Barker, J., Vincent, E., Arora, A., Chang, X., Khudanpur, S., Manohar, V., Povey, D., Raj, D., Snyder, D., Subramanian, A. S., Trmal, J., Yair, B. B., Boeddeker, D., Ni, Z., Fujita, Y., Horiguchi, S., Kanda, N., Yoshioka, T., and Ryant, NN. (**2020**). "CHiME-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings," arXiv:2004.09249.

Wu, S., Wang, C., Chen, H., Dai, Y., Zhang, C., Wang, R., Lan, H., Du, J., Lee, C.-H., Chen, J., Siniscalchi, S. M., Scharenborg, O., Wang, Z.-Q., Pan, J., and Gao, J. (**2024**). "The multimodal information based speech processing (MISP) 2023 challenge: Audio-visual target speaker extraction," in *Proceedings of ICASSP*, pp. 8351–8355.

Yu, F., Zhang, S., Fu, Y., Xie, L., Zheng, S., Du, Z., Huang, W., Guo, P., Yan, Z., Ma, B., Xu, X., and Bu, H. (**2022**). "M2MeT: The ICASSP 2022 multi-channel multi-party meeting transcription challenge," in *Proceedings of ICASSP*, pp. 6167–6171.

Zhang, W., Jung, J-w., Watanabe, S., and Qian, Y. (**2024**). "Improving design of input condition invariant speech enhancement," in *Proceedings of ICASSP*, pp. 10696–10700.

Zhang, W., Saijo, K., Wang, Z.-Q., Watanabe, S., and Qian, Y. (**2023**). "Toward universal speech enhancement for diverse input conditions," in *Proceedings of ASRU*.

Zhang, W., Shi, J., Li, C., Watanabe, S., and Qian, Y. (**2021**). "Closing the gap between time-domain multi-channel speech enhancement on real and simulation conditions," in *Proceedings of WASPAA*, pp. 146–150.

Zheng, C., Zhang, H., Liu, W., Luo, X., Li, A., Li, X., C. J., and Moore, B. (**2023**). "Sixty years of frequency-domain monaural speech enhancement: From traditional to deep learning methods," Trends Hear. **27**, 1–52.

Zorilă, C., and Doddipatla, R. (**2022**). "Speaker reinforcement using target source extraction for robust automatic speech recognition," in *Proceedings of ICASSP*, pp. 6297–6301.