

ESPnet-SE++: Speech Enhancement for Robust Speech Recognition, Translation, and Understanding

Yen-Ju Lu^{*1}, Xuankai Chang^{*2}, Chenda Li³, Wangyou Zhang³, Samuele Cornell⁴, Zhaoheng Ni⁵, Yoshiki Masuyama^{2,7}, Brian Yan², Robin Scheibler⁶, Zhong-Qiu Wang², Yu Tsao¹, Yanmin Qian³, Shinji Watanabe²

¹Academia Sinica, Taipei ²Carnegie Mellon University, USA

³Shanghai Jiao Tong University, Shanghai ⁴Università Politecnica delle Marche, Italy

⁵Meta AI, USA ⁶LINE Corporation, Japan ⁷Tokyo Metropolitan University, Japan

Abstract

This paper presents recent progress on integrating speech separation and enhancement (SSE) into the ESPnet toolkit. Compared with the previous ESPnet-SE work, numerous features have been added, including recent state-of-the-art speech enhancement models with their respective training and evaluation recipes. Importantly, a new interface has been designed to flexibly combine speech enhancement front-ends with other tasks, including automatic speech recognition (ASR), speech translation (ST), and spoken language understanding (SLU). To showcase such integration, we performed experiments on carefully designed synthetic datasets for noisy-reverberant multi-channel ST and SLU tasks, which can be used as benchmark corpora for future research. In addition to these new tasks, we also use CHiME-4 and WSJ0-2Mix to benchmark multi- and single-channel SE approaches. Results show that the integration of SE front-ends with back-end tasks is a promising research direction even for tasks besides ASR, especially in the multi-channel scenario. The code is available online at <https://github.com/ESPnet/ESPnet>. The multi-channel ST and SLU datasets, which are another contribution of this work, are released on HuggingFace.

Index Terms: speech enhancement, speech recognition, speech translation, spoken language understanding

1. Introduction

Speech separation and enhancement (SSE) aims at extracting a target speech signal from noise, reverberation, and interfering speakers. It is essential to robust speech recognition [1, 2], assistive hearing [3], and robust speaker recognition [4]. SSE methods have benefited greatly from the recent development of deep learning (DL) approaches, and DL-based methods are now the de-facto standard. This opens up the possibility for end-to-end integration of SSE methods with many downstream speech processing back-end tasks [e.g., automatic speech recognition (ASR), keyword spotting, and speech translation (ST), to name a few]. In fact, many works have been recently exploring this research direction, mainly concerning ASR [5–8].

To accelerate research in SSE, ESPnet-SE toolkit [9] was developed and currently supports multiple state-of-the-art enhancement approaches and various corpora. In the meantime, there is currently a lot of effort and interest for robust ASR, ST, and spoken language understanding (SLU) in noisy and possibly distant speech scenarios [10, 11], such as ones encoun-

tered by smart-speaker devices. This motivated us to extend ESPnet-SE into ESPnet-SE++, which has a re-designed interface focused on modularity. This new interface allows for a seamless combination of different front-end SSE models with various downstream tasks such as ASR, ST and SLU.

The contributions of ESPnet-SE++ are summarized below:

- We significantly extend ESPnet-SE by providing new recipes for several enhancement corpora and challenges.
- In addition to new recipes, several state-of-the-art single-channel and multi-channel enhancement approaches have been added. Including unsupervised separation [12] and generative speech enhancement [13] approaches.
- A redesigned modular interface allows a flexible, “plug-and-play” combination of SE front-ends with different ESPnet back-end tasks such as ASR, SLU, ST, etc. As we showcase in Section 4 for ST and SLU, this easily allows to test and possibly fine-tune (as done for SLU) the same pre-trained front-end for multiple applications.
- We develop two multi-channel noisy-reverberant datasets derived from SLURP and Libri-Trans, respectively. We simulate potential applications of SLU and ST in a distant speech setting and showcase how tight integration of SE front-end with the back-end opens up promising research directions for other tasks besides ASR.

An extensive experimental evaluation is performed to showcase the flexibility of ESPnet-SE++. In particular, we conducted different experiments using four different datasets: CHiME-4, WSJ0-2mix, and the two purposely developed SLU and ST multi-channel datasets mentioned previously. Results show that speech enhancement can improve the performance in a wide variety of downstream tasks.

2. Related Works

In this section, we briefly compare ESPnet-SE with other open-source deep learning-based speech enhancement and separation toolkits, such as *nusl* (North-western University Source Separation Library) [14], *Onssen* (An Open-source Speech Separation and Enhancement Library) [15], *Asteroid* (Audio source separation on Steroids) [16], and *SpeechBrain* [17]. While *nusl* provides several state-of-the-art SE methods, the data preparation and experiments of *nusl* and *Onssen* are not easily configurable from the command line [16]. On the other hand, *Asteroid* and *SpeechBrain* provide a whole pipeline from data preparation to enhancing and evaluating the testing speech.

* denotes equal contribution.

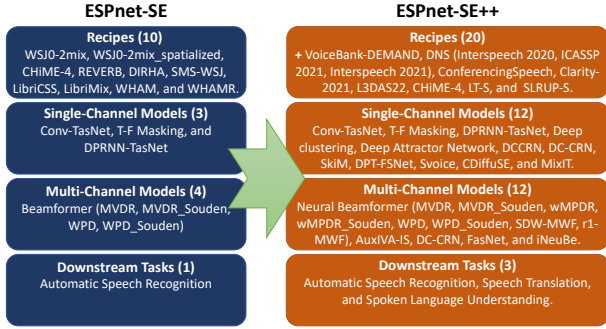


Figure 1: Comparison for ESPnet-SE and ESPnet-SE++. Recipes in ESPnet-SE are not shown in the block of ESPnet-SE++.

However, *Asteroid* is exclusively designed for SE front-end processing. On the other hand, *SpeechBrain*, supports both SSE front-end and back-end (e.g. ASR) techniques along with corresponding recipes but does not yet integrate multiple speech processing tasks into a single recipe. Although the application of SE can already be beneficial, especially in the multi-channel case, various studies suggest that joint-optimization with downstream tasks can further boost the performance [10, 18, 19]. We summarize the difference between ESPnet-SE and ESPnet-SE++ in Figure 1. ESPnet-SE++ owns 20 SSE recipes, with 24 different enhancement/separation models.

3. Features of ESPnet-SE++

ESPnet-SE covers a wide range of speech enhancement/separation models and recipes, including (1) single-channel and multi-channel; (2) single-source and multi-source; (3) time-domain and frequency-domain enhancement models. ESPnet-SE++, is extended to support the latest state-of-the-art models and the latest enhancement/separation recipes, and also provides the combination of enhancement with other downstream tasks, including ASR, ST, and SLU. In addition, ESPnet-SE++ supports the native complex datatype built by PyTorch for complex time-frequency domain processing instead of relying on custom workarounds built on top of PyTorch real-valued tensors.

3.1. Recipes

ESPnet-SE provides various recipes for several SSE benchmark corpora which are derived from WSJ [20] and LibriSpeech [21] corpora. ESPnet-SE++, on top of these, adds 10 new recipes, including speech enhancement/separation corpora: Voicebank-DEMAND [22], speech enhancement challenges: DNS (Interspeech 2020 [23], ICASSP 2021 [24], and Interspeech 2021 [25]), ConferencingSpeech 2021 [26], Clarity-2021 [27], and L3DAS22 [28]. In addition, we develop a new recipe that enables an all-in-one combination of multiple speech processing components with a front-end SSE module. This includes robust ASR (CHiME-4), ST (Libri-Trans [29] mixture), and SLU (SLURP [30] mixture) datasets. These recipes contain the data preparation steps for both enhancement and downstream speech processing tasks, the joint-training of the front-end and back-end models, and the output of enhanced/separated speech with the recognition/translation/understanding results. The function of each stage in the unified pipeline is depicted in Figure 2.

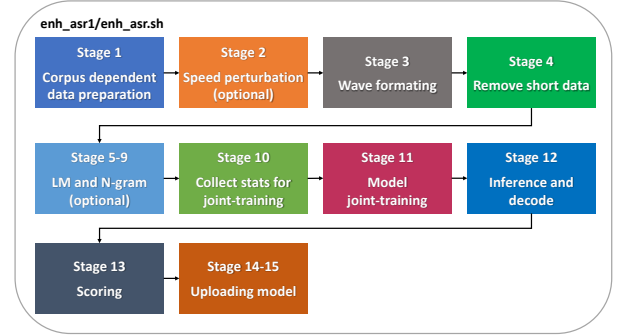


Figure 2: Block diagram of `enh_asr1/enh_asr.sh` in ESPnet-SE++, the combination of SE and back-end models.

3.2. Models

In ESPnet-SE++, we add various classical and state-of-the-art supervised SSE models, including (1) single-channel models: Deep Clustering [31], Deep Attractor Network [32], DCCRN [33], DC-CRN [34], SkiM [35], DPT-FSNet [36], SVoice [37], and (2) multi-channel models: neural beamformer with more supported types (c.f. Section 3.4), FaSNet [38], and iNeuBe [39, 40]. In addition, we further extend the ESPnet-SE framework to support various training and inference procedures, including (3) generative enhancement model, CDiffuSE [13], and (4) unsupervised separation model, MixIT [12].

3.3. Training Objectives

The training objective can vary a lot between different speech enhancement and separation tasks. To make it more flexible and customizable, we disentangle the training objective into two abstract classes: a *criterion* and a *wrapper*.

The *criterion* is an implementation of a simple loss function, e.g. SI-SNR [41, 42], CI-SDR [43, 44], mean square error (MSE) on time-frequency masks, etc. These losses usually take both target and estimated tensors as input and output a scalar loss value.

The *wrapper* instead deals with a post-processing of the *criterion* output. For example, the procedure to find the best permutation in the PIT [45] algorithm is implemented as a *wrapper*. We have also added MixIT [12] *wrapper* for unsupervised speech separation. This modular design allows researchers to add complex training objectives conveniently by writing custom *wrapper* classes and combining it with a suitable *criterion*.

Furthermore, ESPnet-SE++ supports multiple training objectives. Different training objectives can be combined with different weights in a multi-task learning (MTL) fashion. The specific implementation of training objectives, including their *wrapper*, *criterion* and MTL weight, can be selected and configured directly in the ESPnet *yaml* configuration file.

3.4. Single- and Multi-channel Speech Enhancement

As mentioned above, ESPnet-SE++ adds several models on top of ESPnet-SE. For single-channel models, DCCRN [33] and DC-CRN [34] are added. To catch up with the latest development of deep learning-based multi-channel speech enhancement systems, we further added several multi-channel SSE models, including FaSNet [38], DC-CRN [34], and iNeuBe [39, 40]. The first one is a time-domain multi-channel model, while the other two operate in the complex time-frequency domain. In addition, there are also various

Table 1: Results of single- and multi-channel speech enhancement approaches on CHiME-4.

Models	PESQ	STOI	SI-SNR (dB)
No processing	2.18	0.870	7.51
Wang <i>et al.</i> [50]	3.68	0.986	22.00
<i>Single-channel Models</i>			
Conv-TasNet (baseline)	2.58	0.892	11.57
*DCCRN	2.59	0.895	12.57
*DC-CRN	2.43	0.880	11.59
<i>Multi-channel Models</i>			
MVDR	2.61	0.954	13.98
MVDR.Souden	2.66	0.954	15.25
*wMPDR	2.60	0.951	13.53
Neural Beamformer *wMPDR.Souden	2.64	0.951	15.24
WPD	2.60	0.950	13.43
WPD.Souden	2.64	0.950	14.89
*SDW-MWF	2.43	0.922	11.87
*r1-MWF	2.67	0.953	15.08
*AuxIVA-ISS	2.49	0.900	10.34
*DC-CRN	2.95	0.948	17.04
*FaSNet	2.70	0.935	14.83
*iNeuBe (DNN ₁)	3.24	0.969	19.52

* The symbol * denotes models newly-added to ESPnet-SE++.

notable updates to the existing ESPnet-SE neural beamformer model. Firstly, the `DNN_Beamformer` module is refactored to be more user-friendly, while keeping backward compatibility. Secondly, more beamformer variants are supported, such as weighted minimum power distortionless response (wMPDR) [46], speech distortion weighted multi-channel Wiener filtering (SDW-MWF) [47], and rank-1 multi-channel Wiener filter (r1-MWF) [1] etc. A differentiable implementation of the blind source separation algorithm AuxIVA-ISS [48] has also been added. Thirdly, these beamforming solutions now rely on the native complex tensors data-type built in PyTorch 1.9.0+, which shows a faster speed and comparable performance compared to our previously-adopted implementation.¹ We also integrate multi-channel processing methods provided by TorchAudio [49] such as power spectral density (PSD) computation, MVDR beamforming, etc.² Last, several numerical stability-related techniques proposed in [2] have been integrated to improve the training stability and performance.

In Table 1, we list the SSE performance of various models supported in the current toolkit on the CHiME-4 corpus, where the newly-added models are marked with *.

3.5. Single-channel Speech Separation

In the initial version of ESPnet-SE [9], we implemented some time-frequency (TF) domain [45] and time domain models [51, 52]. In the new implementation, we design these two kinds of models into a unified framework, which consists of an *encoder*, a *separator* and a *decoder*. Depending on the model type, *encoder* and *decoder* could be short-time Fourier transform (STFT) and inverse STFT (iSTFT) for the TF domain models; or they could be convolutional layers and transposed convolutional layers for time domain models such as [41]. The *separator* is typically a sequence mapping neural network. It takes the input from the *encoder*, and generates S output features. S is the number of speech sources to be separated, and

¹https://github.com/kamo-naoyuki/pytorch_complex

²<https://pytorch.org/audio/main/functional#multi-channel>

Table 2: Results of single-channel speech separation models on WSJ0-2mix.

Models	PESQ	STOI	SI-SNR (dB)
No processing	2.01	0.738	0.00
Conv-TasNet (baseline)	3.25	0.953	15.94
DPRNN-TasNet	3.47	0.968	17.91
*Deep Clustering	2.15	0.845	7.91
*Deep Attractor Network	2.68	0.893	10.30
*DC-CRN	3.11	0.935	13.01
*SkiM	3.47	0.966	18.45
*MixIT (Conv-TasNet)	3.00	0.938	13.50

* The symbol * denotes models newly-added to ESPnet-SE++.

Table 3: WER for system combination of speech enhancement with speech recognition on CHiME-4 corpus.

Models	SIMU (%)	REAL (%)
No processing	19.7	18.0
<i>Single-channel Models</i>		
Conv-TasNet	17.4	15.2
DCCRN	16.3	15.8
<i>Multi-channel Models</i>		
FaSNet	15.7	23.8
Neural Beamformer	10.8	13.7
iNeuBe (DNN ₁)	9.0	35.8

for most speech enhancement models, $S = 1$. The *decoder* transforms the features into the target audios.

An interface holding together the *encoder*, *separator* and *decoder* has been designed. This modular design allows to explore many different architectural variations with less boilerplate code. Based on the unified framework, we also enrich the speech separation models in ESPnet-SE, including deep clustering [31], deep attractor network [32], DC-CRN [34], SkiM [35] and, MixIT [12]. The reproduced results on the WSJ0-2mix [31] benchmark are listed in Table 2.

4. Combination tasks of ESPnet-SE++

ESPnet-SE++ allows for a tight and easy integration of SSE front-end processing techniques and back-end tasks as introduced in Sec. 3.1. To showcase this, we provide three combined recipes, CHiME-4, LT-S, and SLURP-S, with speech enhancement (SE) front-end subtask, followed by a back-end ASR/ST/SLU subtask. We performed several experiments with different techniques to assess if and how SE could improve the results of the back-end tasks even when multi-condition training is employed.

4.1. Data Simulation

We created two multi-channel noisy-reverberant datasets based on SLURP and Libri-Trans (LT): SLURP-S and LT-S, where S stands for spatialized. We augmented the original SLURP and LT datasets using room impulse responses generated via Pyroomacoustics [53], and noises from FSD50k [54], SINS [55]. In detail, for each original utterance from SLURP and LT we simulate a smart-speaker scenario where the target signal is captured by a 4-microphone circular array with a diameter of 10 cm. For each utterance, we sample a room size from uniform distribution $U(10, 100) m^2$ and a reverberation time (T60) from $U(0.2, 0.6) s$, typical of most indoor settings [56].

Room height is sampled from $U(2.5, 4)$ m. The target speech is contaminated by 1 up to 4 point-source localized noises from FSD50k and, in addition, also one diffuse noise audio sample from SINS. Diffuse noise is simulated using the technique outlined in [57]. The positions of the array, point-source noises and target are sampled randomly in the virtual room with the constraint of being at least 0.5 m apart from each other and the walls. The Signal-to-Noise ratio (SNR) for target versus point-source noises is sampled from $U(0, 15)$ dB, while for diffuse noise is sampled from $U(12, 35)$ dB.

Regarding SLURP, since it already comprises noisy-reverberant examples, we use DNSMOS model [58] to select only a subset of sufficiently clean utterances (with estimated BAK ≥ 3.2) from training real, development and test, to use for SLURP-S. This process leaves 24497 utterances for training real, and 6387 and 6099 for development and test respectively. No utterance is discarded from training synthetic. The discarded utterances are kept but treated as single-channel examples with no oracle SE supervision, and are left for possible future work on semi/self-supervised joint SE+SLU. Instead, LT-S retains the same number of utterances of the original LT as it features only clean speech.

Finally, in order to assess generalization to unseen noise conditions, we also generate, for both datasets, an additional test set with diffuse noise derived from QUT [59] instead of SINS.

4.2. Combination of SE with ASR on CHiME-4

In Table 3 we report the results obtained by combining different SE front-ends with an ASR model on the CHiME-4 corpus [11]. CHiME-4 consists of both real and simulated noisy speech signals and is particularly suited for this analysis. We used a transformer encoder-decoder as the E2E ASR model that is pre-trained using the official ESPnet CHiME-4 recipe. The SE models are instead pre-trained using simulated data. The performance of these systems in terms of signal-based metrics is summarized in Sec. 3.4. After pre-training, the entire SE+ASR system is fine-tuned together with both SE and ASR losses. On the simulated data, all SE+ASR systems improve performance with respect to using solely the ASR model. This result confirms the effectiveness of jointly fine-tuning SE and ASR systems for robust speech recognition. Meanwhile, some models resulted in worse WERs on the real data. This is a commonly observed phenomenon when evaluating SE models on simulated and real data [19, 60], especially when they are not jointly optimized with the downstream ASR task. The neural beamformer achieved the best performance in both simulated and real sets.

4.3. Combination of SE with ST on LT-S

In the left part of Table 4, we explore the combination of SE with a ST task performed on the simulated LT-S dataset outlined in Section 4.1. For the ST model, we used a conformer-encoder transformer-decoder as the back-end model. The ST model was trained on the original clean Libri-Trans dataset and achieved a BLEU score of 16.7 on the test set. The front-end SE models were pre-trained using both LT-S and SLURP-S datasets using the anechoic clean speech as target (thus performing joint denoising and dereverberation). For the evaluation, we directly concatenate the pretrained SE models with the ST model in an end-to-end manner. Note that the whole system was not fine-tuned. We can see that the addition of the SE front-end generally improves the performance against the noisy speech even without fine-tuning. Among the SE models considered,

Table 4: Results of combining speech enhancement with speech translation on LT-S corpus and speech enhancement with spoken language understanding on SLURP-S corpus.

SE	LT-S (BLEU)		SLURP-S (Acc%)	
	TEST	TEST QUT	TEST	TEST QUT
No processing	7.3	7.4	74.2	73.4
Single-channel Models				
Conv-TasNet	7.4	7.5	64.3	64.0
DCCRN	10.1	10.1	77.0	76.3
Multi-channel Models				
FaSNet	7.7	3.6	73.1	72.5
iNeuBe (DNN ₁)	13.2	13.2	77.8	77.5
iNeuBe (mfmcwf)	13.0	13.0	80.1	80.2
iNeuBe (DNN ₂)	14.4	14.6	80.4	80.3

both iNeuBe models based on DNN₂ and multi-frame multi-channel Wiener filter (mfmcwf), bring the largest performance gain. In the single-channel case, DCCRN, as expected, outperforms Conv-Tasnet.

4.4. Combination of SE with SLU on SLURP-S

In the right part of Table 4, we explore instead the combination of SE with a SLU back-end and report the intent classification accuracy. Regarding the SLU model, we used the conformer-encoder transformer-decoder trained with multi-condition training (clean and noisy-reverberant) on the SLURP-S corpus. After SLU back-end pre-training, we fine-tuned the SE and SLU systems together. The SE models are the same as used in the ST experiments in Sec. 4.3. Notably, we can see that in this instance, contrary to what is observed on CHiME-4 which is arguably less acoustically challenging than SLURP-S, Conv-TasNet and FaSNet lead to degraded performance. On the other hand, both DCCRN and iNeuBe are able to outperform the multi-condition SLU model. This confirms that models that rely on complex spectral mapping [40, 50] are more robust in challenging acoustical conditions.

5. Conclusions

In this work we presented the ESPnet-SE++ toolkit. Built on top of ESPnet-SE, it includes new state-of-the-art models, losses, and recipes for speech enhancement corpora and challenges in various scenario. The toolkit interface has also been overhauled and improved with greater modularity. This allows a flexible combination of speech enhancement with ESPnet back-end tasks such as ASR, ST, and SLU. As an additional contribution, to showcase such integration of front-end and back-end components, we developed two multi-channel noisy-reverberant corpora based on SLURP and Libri-Trans. Experiments show that the use of SE front-end models improves both signal-based evaluation metrics and back-end tasks such as ASR, ST, and SLU, even when the back-end is trained on a multi-condition fashion. Future work could investigate further integration of different SE techniques with back-end tasks, including self-supervised pre-training via MixIT and generative enhancement models, with the goal of improving generalization to unseen acoustical conditions.

6. References

- [1] Z. Wang *et al.*, “Rank-1 constrained multichannel Wiener filter for speech recognition in noisy environments,” *Computer Speech & Lang.*, vol. 49, pp. 37–51, 2018.

- [2] W. Zhang *et al.*, “End-to-end dereverberation, beamforming, and speech recognition with improved numerical stability and advanced frontend,” in *Proc. ICASSP*, 2021, pp. 6898–6902.
- [3] D. Wang, “Deep learning reinvents the hearing aid,” *IEEE spectrum*, vol. 54, no. 3, pp. 32–37, 2017.
- [4] J. H.L. Hansen and T. Hasan, “Speaker recognition by machines and humans: A tutorial review,” *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 74–99, 2015.
- [5] A. S. Subramanian *et al.*, “Speech enhancement using end-to-end speech recognition objectives,” in *Proc. WASPAA*, 2019, pp. 229–233.
- [6] X. Chang *et al.*, “MIMO-Speech: End-to-end multi-channel multi-speaker speech recognition,” in *Proc. ASRU*, 2019, pp. 237–244.
- [7] A. S. Subramanian *et al.*, “Far-field location guided target speech extraction using end-to-end speech recognition objectives,” in *Proc. ICASSP*, 2020, pp. 7299–7303.
- [8] Z. Chen *et al.*, “Continuous speech separation: Dataset and analysis,” in *Proc. ICASSP*, 2020, pp. 7284–7288.
- [9] C. Li *et al.*, “ESPNet-SE: End-to-end speech enhancement and separation toolkit designed for ASR integration,” in *Proc. SLT*, 2021, pp. 785–792.
- [10] R. Haeb-Umbach *et al.*, “Speech processing for digital home assistants: Combining signal processing with deep-learning techniques,” *IEEE Signal processing magazine*, vol. 36, no. 6, pp. 111–124, 2019.
- [11] E. Vincent *et al.*, “An analysis of environment, microphone and data simulation mismatches in robust speech recognition,” *Computer Speech & Lang.*, vol. 46, pp. 535–557, 2017.
- [12] S. Wisdom *et al.*, “Unsupervised sound separation using mixture invariant training,” in *Proc. NeurIPS*, 2020, pp. 3846–3857.
- [13] Y.-J. Lu *et al.*, “Conditional diffusion probabilistic model for speech enhancement,” *arXiv:2202.05256*, 2022.
- [14] E. Manilow, P. Seetharaman, and B. Pardo, “The Northwestern University source separation library,” in *Proc. ISMIR*, 2018, pp. 23–27.
- [15] Z. Ni and M. I. Mandel, “Onssen: An open-source speech separation and enhancement library,” *arXiv:1911.00982*, 2019.
- [16] M. Pariente *et al.*, “Asteroid: The PyTorch-based audio source separation toolkit for researchers,” in *Proc. Interspeech*, 2020, pp. 2637–2641.
- [17] M. Ravanelli *et al.*, “SpeechBrain: A general-purpose speech toolkit,” *preprint arXiv:2106.04624*, 2021.
- [18] T. von Neumann *et al.*, “Multi-talker asr for an unknown number of sources: Joint training of source counting, separation and asr,” *arXiv preprint arXiv:2006.02786*, 2020.
- [19] W. Zhang *et al.*, “Closing the gap between time-domain multi-channel speech enhancement on real and simulation conditions,” in *Proc. WASPAA*, 2021, pp. 146–150.
- [20] LDC, *LDC Catalog: CSR-1 (WSJ0) Complete*, 1993.
- [21] V. Panayotov *et al.*, “Librispeech: An ASR corpus based on public domain audio books,” in *Proc. ICASSP*, 2015, pp. 5206–5210.
- [22] C. Valentini-Botinhao *et al.*, “Investigating RNN-based speech enhancement methods for noise-robust text-to-speech,” in *Proc. SSW*, 2016, pp. 146–152.
- [23] C. K. Reddy *et al.*, “The INTERSPEECH 2020 deep noise suppression challenge: Datasets, subjective testing framework, and challenge results,” in *Proc. Interspeech*, 2020, pp. 2492–2496.
- [24] C. K. Reddy *et al.*, “ICASSP 2021 deep noise suppression challenge,” in *Proc. ICASSP*, 2021, pp. 6623–6627.
- [25] C. K. Reddy *et al.*, “INTER-SPEECH 2021 deep noise suppression challenge,” in *Proc. Interspeech*, 2021, pp. 2796–2800.
- [26] W. Rao *et al.*, “ConferencingSpeech challenge: Towards far-field multi-channel speech enhancement for video conferencing,” in *Proc. ASRU*, 2021, pp. 679–686.
- [27] S. Graetzer *et al.*, “Clarity-2021 challenges: Machine learning challenges for advancing hearing aid processing,” in *Proc. Interspeech*, 2021, pp. 686–690.
- [28] E. Guizzo *et al.*, “L3DAS22 challenge: Learning 3D audio sources in a real office environment,” *arXiv:2202.10372*, 2022.
- [29] A. C. Kocabiyikoglu, L. Besacier, and O. Kraif, “Augmenting Librispeech with French translations: A multimodal corpus for direct speech translation evaluation,” in *Proc. LREC*, 2018.
- [30] E. Bastianelli *et al.*, “SLURP: A spoken language understanding resource package,” in *Proc. EMNLP*, 2020, pp. 7252–7262.
- [31] J. R. Hershey *et al.*, “Deep clustering: Discriminative embeddings for segmentation and separation,” in *Proc. ICASSP*, 2016, pp. 31–35.
- [32] Z. Chen, Y. Luo, and N. Mesgarani, “Deep attractor network for single-microphone speaker separation,” in *Proc. ICASSP*, 2017, pp. 246–250.
- [33] Y. Hu *et al.*, “DCCRN: Deep complex convolution recurrent network for phase-aware speech enhancement,” in *Proc. Interspeech*, 2020, pp. 2472–2476.
- [34] K. Tan, X. Zhang, and D. Wang, “Deep learning based real-time speech enhancement for dual-microphone mobile phones,” *IEEE TASP*, vol. 29, pp. 1853–1863, 2021.
- [35] C. Li *et al.*, “SkiM: Skipping memory LSTM for low-latency real-time continuous speech separation,” *arXiv:2201.10800*, 2022.
- [36] F. Dang, H. Chen, and P. Zhang, “DPT-FSNet: Dual-path transformer based full-band and sub-band fusion network for speech enhancement,” *arXiv:2104.13002*, 2021.
- [37] E. Nachmani, Y. Adi, and L. Wolf, “Voice separation with an unknown number of multiple speakers,” in *Proc. ICML*, 2020.
- [38] Y. Luo *et al.*, “FaSNet: Low-latency adaptive beamforming for multi-microphone audio processing,” in *Proc. ASRU*, 2019, pp. 260–267.
- [39] Z.-Q. Wang, G. Wichern, and J. Le Roux, “Leveraging low-distortion target estimates for improved speech enhancement,” *preprint arXiv:2110.00570*, 2021.
- [40] Y.-J. Lu *et al.*, “Towards low-distortion multi-channel speech enhancement: The ESPNet-SE submission to the L3DAS22 challenge,” *preprint arXiv:2202.12298*, 2022.
- [41] Y. Luo and N. Mesgarani, “TasNet: Time-Domain Audio Separation Network for Real-Time, Single-Channel Speech Separation,” in *Proc. ICASSP*, 2018, pp. 696–700.
- [42] J. Le Roux *et al.*, “SDR—half-baked or well done?” In *Proc. ICASSP*, 2019, pp. 626–630.
- [43] C. Boeddeker *et al.*, “Convolutional transfer function invariant SDR training criteria for multi-channel reverberant speech separation,” in *Proc. ICASSP*, 2021, pp. 8428–8432.
- [44] R. Scheibler, “SDR — medium rare with fast computations,” in *Proc. ICASSP*, to appear, 2022.
- [45] M. Kolbaek *et al.*, “Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks,” *IEEE TASP*, vol. 25, no. 10, pp. 1901–1913, 2017.
- [46] C. Boeddeker *et al.*, “Jointly optimal dereverberation and beamforming,” in *Proc. ICASSP*, 2020, pp. 216–220.
- [47] A. Spriet, M. Moonen, and J. Wouters, “Spatially pre-processed speech distortion weighted multi-channel Wiener filtering for noise reduction,” *Signal Process.*, vol. 84, no. 12, pp. 2367–2387, 2004.
- [48] R. Scheibler and M. Togami, “Surrogate source model learning for determined source separation,” in *Proc. ICASSP*, 2021, pp. 176–180.
- [49] Y.-Y. Yang *et al.*, “Torchaudio: Building blocks for audio and speech processing,” *arXiv:2110.15018*, 2021.
- [50] Z.-Q. Wang, P. Wang, and D. Wang, “Complex spectral mapping for single-and multi-channel speech enhancement and robust ASR,” *IEEE TASP*, vol. 28, pp. 1778–1787, 2020.
- [51] Y. Luo and N. Mesgarani, “Conv-TasNet: Surpassing ideal time-frequency magnitude masking for speech separation,” *IEEE TASP*, vol. 27, no. 8, pp. 1256–1266, 2019.
- [52] Y. Luo, Z. Chen, and T. Yoshioka, “Dual-path RNN: Efficient long sequence modeling for time-domain single-channel speech separation,” in *Proc. ICASSP*, 2020, pp. 46–50.
- [53] R. Scheibler, E. Bezzam, and I. Dokmanić, “Pyroomacoustics: A python package for audio room simulation and array processing algorithms,” in *Proc. ICASSP*, 2018, pp. 351–355.
- [54] E. Fonseca *et al.*, “FSD50k: An open dataset of human-labeled sound events,” *IEEE TASP*, vol. 30, pp. 829–852, 2022.
- [55] G. Dekkers *et al.*, “The SINS database for detection of daily activities in a home environment using an acoustic sensor network,” *Proc. DCASE*, pp. 1–5, 2017.
- [56] J. Traer and J. H. McDermott, “Statistics of natural reverberation enable perceptual separation of sound and space,” *Proc. NAS*, vol. 113, no. 48, E7856–E7865, 2016.
- [57] E. A. Habets, I. Cohen, and S. Gannot, “Generating nonstationary multisensor signals under a spatial coherence constraint,” *J. Acoust. Soc. Am.*, vol. 124, no. 5, pp. 2911–2917, 2008.
- [58] C. K. Reddy, V. Gopal, and R. Cutler, “DNSMOS: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors,” in *Proc. ICASSP*, 2021, pp. 6493–6497.
- [59] D. Dean *et al.*, “The QUT-NOISE-TIMIT corpus for evaluation of voice activity detection algorithms,” in *Proc. Interspeech*, 2010, pp. 3110–3113.
- [60] S. E. Eskimez *et al.*, “Human listening and live captioning: Multi-task training for speech enhancement,” in *Proc. Interspeech*, 2021, pp. 2686–2690.