ESPNET-SE: END-TO-END SPEECH ENHANCEMENT AND SEPARATION TOOLKIT DESIGNED FOR ASR INTEGRATION

Chenda Li[1], Jing Shi[2,3*], Wangyou Zhang[1*], Aswin Shanmugam Subramanian[3], Xuankai Chang[3],*
Naoyuki Kamo, Moto Hira[4], Tomoki Hayashi[5,6], Christoph Boeddeker[7], Zhuo Chen[8], Shinji Watanabe[3]
[1]Shanghai Jiao Tong University, [2]Institute of Automation, Chinese Academy of Sciences,
[3]Johns Hopkins University, [4]Facebook AI, [5]Nagoya University,
[6]Human Dataware Lab. Co., Ltd., [7]Paderborn University, [8]Microsoft Research

ABSTRACT

We present ESPnet-SE, which is designed for the quick development of speech enhancement and speech separation systems in a single framework, along with the optional downstream speech recognition module. ESPnet-SE is a new project which integrates rich automatic speech recognition related models, resources and systems to support and validate the proposed front-end implementation (i.e. speech enhancement and separation). It is capable of processing both single-channel and multi-channel data, with various functionalities including dereverberation, denoising and source separation. We provide all-in-one recipes including data pre-processing, feature extraction, training and evaluation pipelines for a wide range of benchmark datasets. This paper describes the design of the toolkit, several important functionalities, especially the speech recognition integration, which differentiates ESPnet-SE from other open source toolkits, and experimental results with major benchmark datasets.

Index Terms— Open-source, end-to-end, speech enhancement, source separation, speech recognition

1. INTRODUCTION

As the core parts of the speech processing front-end, speech enhancement and separation (SE) have been studied for decades. Speech enhancement tries to improve the intelligibility and quality of speech contaminated by additive noise and reverberation [1], while speech separation focuses on the speech in multi-speaker conditions, which is also known as the "cocktail party" problem [2]. Speech enhancement and separation are highly desirable for a vast range of applications, such as smart speakers, automatic meeting transcription, automatic captioning for audio/video recordings (e.g., YouTube), multi-party human-machine interaction (e.g., in the world of Internet of things (IoT)), and advanced hearing aids, where noisy speech is commonly encountered [3, 4, 5, 6].

Recently with the great success of deep learning techniques, end-to-end speech enhancement and separation systems (hereinafter collectively referred to as E2E-SE) have grown in popularity and have even started replacing conventional systems described in [1] and [7]. The E2E-SE system, taking the raw waveform of noisy speech as input, outputs one or several speech sources for each target speaker. This end-to-end paradigm can easily integrate signal processing algorithms along with various learnable neural networks, making full use of neural network toolkits. Meanwhile, many downstream speech processing back-end tasks (i.e., automatic speech recognition, keyword spotting and speech translation) can also benefit greatly by cascading after it.

In this paper, we introduce a new E2E-SE toolkit named ESPnet-SE[1], which is an extension of the open-source speech processing toolkit ESPnet [8]. ESPnet-SE fully considers the various forms of speech input in the front-end scenes and meanwhile flexibly and organically integrated with the downstream automatic speech recognition (ASR) task, making it a user-friendly toolkit to easily build totally end-to-end robust ASR systems, even without need for clean speech signals. The toolkit provides adaptability to different speech data, including (1) single and multiple speakers, (2) single and multiple channels, (3) anechoic and reverberant conditions. Moreover, thanks to the ripe and efficient ASR modules in ESPnet, rich speech recognition related models, resources and systems can be optionally concatenated after the E2E-SE system, enabling evaluation and joint optimization with ASR, which distinguishes ESPnet-SE from other open source toolkits.

2. RELATED WORK

This section briefly compares ESPnet-SE to other open-source deep learning-based speech enhancement and separation toolkits. We pick up four well-maintained frameworks in this range, including *nussl* (North-western University Source Separation Library) [9], *Onssen* (An Open-source Speech Separation and Enhancement Library) [10], *Open-Unmix* [11] and *Asteroid* (Audio source separation on Steroids) [12]. Other projects (e.g., *padertorch*[2]) may also contain similar features in E2E-SE to some extent, but because they are not systematically designed to address these tasks or still under construction in the paper submission stage, no comparison is made here.

For the first two frameworks *nussl* and *Onssen*, they are based on the *PyTorch* [13] platform and provide several state-of-the-art SE methods, along with the training and evaluation scripts. However, data preparation steps are not provided in the recipes and experiments are not easily configurable from the command line [12]. On the other hand, *Open-Unmix* and *Asteroid* provide a whole pipeline from data preparation to evaluation with common line support. Note that the former is aimed for a specific model of *Open-Unmix* on the music separation task. The same discussion is also applied to a series of isolated speech separation projects, such as `kaituoxu/TasNet`, `kaituoxu/Conv-TasNet`, `yluo42/TAC` and so on. For *Asteroid*, it is well-designed for the speech separation task, which supports a large range of datasets and architectures, and a set of recipes to reproduce some important papers. However, its overall framework

---

*These authors contributed equally to this work.

[1]The demo page is https://colab.research.google.com/drive/1fjRJCh96SoYLZPRxsjF9VDv4Q2VoIckI
[2]https://github.com/fgnt/padertorch

Table 1: Comparison with other open-source deep learning based speech enhancement and separation toolkits, where ⋆ denotes an on-going project in August 2020.

	nussl	Onssen	Open-Unmix	Asteroid	ESPnet-SE
T-F masking	✓	✓		✓	✓
Chimera	✓	✓		✓	
Deep clustering	✓	✓		✓	⋆
TasNet	⋆	✓		✓	✓
DPRNN	⋆			✓	✓
Wavesplit				⋆	⋆
WPE					✓
Neural beamformer					✓
OpenUnmix	⋆		✓	⋆	
Provide data preparation?			✓	✓	✓
Support training and evaluation?	✓	✓	✓	✓	✓
Support speech source separation?	✓	✓		✓	✓
Support music source separation?	✓		✓	⋆	
Support downloading pretrained model?	✓	⋆	✓	✓	✓
Support integration with ASR?					✓
# of supported datasets	2	3	5	10	10
Backend	PyTorch	PyTorch	PyTorch & NNabla	PyTorch	PyTorch
License	MIT	GPL-3.0	MIT	MIT	Apache-2.0

still belongs to the category of the front-end, which outputs the separated or enhanced speech signals. Although this application itself is quite important, it would not be fully optimized for the downstream application, including ASR systems. Actually, with the gradual maturity of speech separation technologies, many works have paid their attention to the optimization of speech enhancement / separation using speech recognition objectives [5, 14, 15, 16, 17, 18, 19, 20].

Taking into account this trend and the direction of future works in the community, we propose the ESPnet-SE toolkit with the design for ASR integration. In our framework, we provide the optional module of ASR to take part in the training and evaluation. Plenty of the well-established or customizable recognition models are provided based on both connectionist temporal classification (CTC) and attention-based encoder-decoder networks. Users can easily choose to use the recognition network for joint training with E2E-SE or to just apply the downloadable pretrained models to recognize the enhanced or separated speech signals. Even on some unsupervised or real collection of speech recordings (e.g. CHiME-4 [19]) without the reference speech signals, due to the existence of transcription, our systems can be used to train directly with the recognition loss. The overall comparison is summarized in Table 1, and it can be seen that the functionalities of our toolkit are complementary to the other toolkits.

3. FEATURES OF ESPNET-SE

The ESPnet-SE toolkit provides two main components: the implementation of different speech enhancement/separation models, and recipes for many common speech enhancement/separation datasets. The ESPnet-SE models are implemented using PyTorch [13] with rich features. The recipes are carefully designed to follow a unified pipeline, with a stage-by-stage processing style. Below we introduce each component in detail.

3.1. Models

Our implemented ESPnet-SE models cover a wide range of scenarios, including (1) time-domain and frequency-domain; (2) single-channel and multi-channel; (3) single-source and multi-source. For time domain models, we implemented the time-domain audio separation network (TasNet) [21, 22] and the dual-path RNN (DPRNN) [23] based variants. For frequency-domain models, we support both time-frequency (T-F) masking [24] and neural beamformer [25, 26] networks. Among these four models, TasNet and DPRNN operate on the single-channel data, while T-F masking and the neural beamformer can operate on both single-channel and multi-channel data. All models support both speech enhancement for single-source input and speech separation for multi-source input. The input for each model is the raw waveform, and the short-time Fourier transform (STFT) will be used in the frequency-domain models to transform the input to frequency domain. All the processing including STFT is implemented with differentiable PyTorch functions.

T-F masking is a widely used approach to speech enhancement and separation. It estimates a T-F mask for each source in the input, and the mask is used to reconstruct the STFT spectrum of each source. The T-F masking model consists of a recurrent neural network (RNN)[3] followed by several parallel linear layers, with each layer corresponding to one source.

TasNet enhances or separates the audio signal in the time domain. It uses a convolution encoder to encode the audio signal into high-level representations, and then perform enhancement or separation on it. The processed representations are then reconstructed back to audio signals by a deconvolution decoder. Recently, the dual-path RNN (DPRNN) based TasNet [23] achieves state-of-the-art performance on the WSJ0-2mix corpus [27].

The neural beamformer is designed for multi-channel input. It consists of a dereverberation module (optional) and a beamforming module. Each module contains a T-F mask estimation network that

---

[3]We can also easily replace it with other architectures from the ESPnet-ASR module, such as CNN, self-attention and conformer.
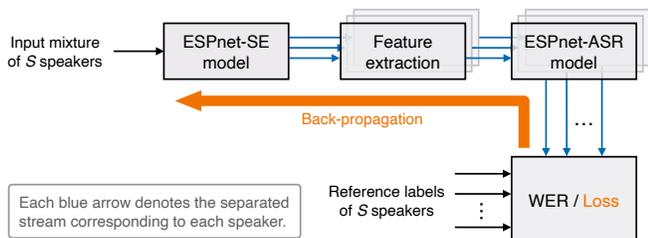
Fig. 1: Block diagram of ESPnet-SE models with ASR integration. The orange part shows the back-propagation process for joint optimization.

is similar to the T-F masking model. The dereverberation module is based on the weighted prediction error (WPE) algorithm [28], and the beamforming module supports various beamforming types, including minimum-variance distortionless response (MVDR) [29], minimum-power distortionless response (MPDR) [30], and weighted power minimization distortionless response (WPD) [31].

3.2. Training for SE and SS

We support multiple loss functions for training the ESPnet-SE models, and they can be divided into three categories: mask approximation loss, signal approximation loss, and metric-based loss.

The mask approximation loss is mainly designed for models with mask estimation components, such as the T-F masking network and the neural beamformer. It measures the error between the estimated mask $\hat{m}$ and the target mask m. Both mean square error (MSE) and cross-entropy (CE) cost functions can be used. We support various types of target masks, including ideal binary mask (IBM) [32], ideal ratio mask (IRM) [33], ideal amplitude mask (IAM, also known as FFT-mask in [34]) and phase-sensitive mask (PSM) [35].

The signal approximation loss measures the error between the enhanced signal $\hat{x}$ and the reference signal x. It can be calculated on either the magnitude spectrum or the complex spectrum. A typical cost function for signal approximation is the mean squared error.

The metric-based loss directly measures the quality of the enhanced signal with a specific metric, such as the scale-invariant signal-to-noise ratio (SI-SNR) [21].

Except for the time domain model that does not support the mask approximation loss, all models have implemented different *forward* functions to support the aforementioned loss functions for training, and it can be easily configured through a configuration file.

For all types of loss functions in speech separation, the permutation invariant training (PIT) [24] is applied to solve the label ambiguity problem. In addition, we also support the unsupervised source separation method from the newly published work [36].

3.3. Evaluation for SE and SS

We evaluate the speech enhancement or separation with various commonly used evaluation metrics, which can be divided into two classes: signal-level metrics and perception-level metrics. For signal-level measurement, the source-to-noise ratio (SNR), source-to-distortion ratio (SDR) [37], source-to-interference ratio (SIR) [37] and source-to-artifact ratio (SAR) [37] are used by default. The scale-invariant signal-to-noise ratio (SI-SNR) [21] is also available. For perception-level measurement, the default metric is the short-time objective intelligibility (STOI) [38] and optionally the
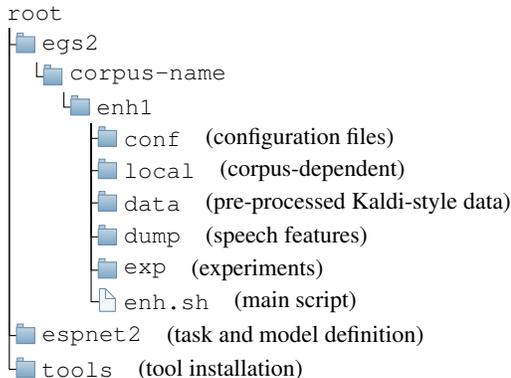

Fig. 2: Directory structure of ESPnet-SE.

perceptual evaluation of speech quality (PESQ) [39] is also available.

Furthermore, the enhanced or separated speech can be evaluated with pretrained single speaker ASR systems, which will be introduced in detail in the following subsection.

3.4. Integration with ASR

Thanks to the unified design, we can easily enable the integration with ASR.

Firstly, the ESPnet-SE models can be directly evaluated with a pretrained ASR model, as illustrated in Fig. 1. The input mixture of S speakers is first separated by the ESPnet-SE model into S streams, corresponding to S different speakers. Then we extract features (e.g. FilterBank) for each stream separately, and feed them into the ASR model to generate the recognized token sequence. Finally, we calculate the word error rate (WER) of the recognition results. When $S > 1$, the final WER is determined with the best permutation between recognition results and reference labels, which minimizes the total WER of all speakers.

Furthermore, we can even jointly optimize both ESPnet-SE and the ASR models. The forward process is the same as described above, after replacing WER with the ASR loss. And permutation invariant training is used when $S > 1$. Since the forward operations are fully differentiable, the loss can be back-propagated from the ASR model to the ESPnet-SE model, so that the entire system can be jointly optimized.

3.5. Recipes

We provide various all-in-one recipes for several SE benchmark corpora, including WSJ0-2mix [27, 40] and its spatialized version [41], CHiME-4 [19], REVERB [20], DIRHA [42], SMS-WSJ [43], LibriCSS [17], LibriMix [44], WHAM [45] and WHAMR [46]. Note that all these datasets are based on the WSJ [47] or Librispeech [48] corpora, so we can use strong ESPnet-ASR pre-trained models for evaluation and joint optimization. The directory structure of ESPnet-SE recipes is depicted in Fig. 2.

All recipes follow a unified pipeline with a stage-by-stage processing style. The function of each stage in `enh.sh` is described below:

Stage 1: Corpus-dependent data is prepared in the `data` directory, following the Kaldi style.

Table 2: Performance (PSEQ / STOI / SDR) on the CHiME-4 1-channel and 6-channel tracks.

Model	Track	Dev (Simu)			Test (Simu)		
		PESQ	STOI	SDR (dB)	PESQ	STOI	SDR (dB)
Noisy Input (CH5)		2.17	0.86	5.78	2.18	0.87	7.54
T-F masking	1-ch	2.55	0.91	10.36	2.46	0.89	11.61
Conv-TasNet		2.46	0.90	11.02	2.40	0.88	12.16
BeamformIt[7]	6-ch	2.31	0.88	5.51	2.20	0.86	6.25
BLSTM MVDR		**2.68**	**0.95**	**13.40**	**2.68**	**0.95**	**14.10**

Stage 2: (Optional) Speed perturbation is performed on the training dataset[4].

Stage 3: The pipe-style audio data in `wav.scp`, such as "`sph2pipe -f wav some/wav/path |`", is dumped to real audio files under the `dump` directory. The sampling rate and and audio format can also be changed if specified.

Stage 4: Underlength and overlength samples are removed from both training and development sets.

Stage 5: The input information of both training and development sets, including the sample length and number of channels, is collected for model training.

Stage 6: Model training is performed.

Stage 7 & 8: Speech enhancement/separation is performed on the evaluation sets to generate enhanced audios, and scores are calculated with different metrics, such as PESQ, STOI and SDR.

Stage 9 & 10: Decoding with a pretrained ASR model is conducted and scores are calculated.

Stage 11 & 12: (Optional) The trained model is packed and then uploaded to the Zenodo community[5] via a specialized API `espnet_model_zoo`[6]. This makes it easy to share and reuse state-of-the-art models obtained with ESPnet.

For the joint optimization of SE and ASR models, Stage 6 above covers the training of both networks as a whole system. As a result, the decoding is directly performed with the trained model itself.

4. EXPERIMENTS EVALUATION

To demonstrate the effectiveness of the proposed ESPnet-SE framework, we conducted several experiments with different datasets from speech enhancement to separation. In this section, we mainly introduce some experimental evaluations on the CHiME-4 speech enhancement challenge, single-channel WSJ0-2mix speech separation task and the spatialized version with multi-channel speech separation. Besides, the speech recognition experiments have also been conducted to show the functionality of our proposed ASR integration.

4.1. Speech enhancement on CHiME-4

We first conduct experiments on CHiME-4 [19], a speech enhancement task where only one speaker exists in each sample. The

Table 3: WER (%) on the CHiME-4 simulation and real data. All results were obtained with the same pretrained ASR model.

Track	Model	Dev		Test	
		real	simu	real	simu
1-ch	Baseline [19]	11.6	13.0	23.7	20.8
	ESPnet-ASR	10.9	12.6	19.5	19.9
	+ Conv-TasNet	15.0	16.4	30.1	26.8
6-ch	Baseline [19]	5.8	6.8	11.5	10.9
	ESPnet-ASR	10.9	12.6	19.5	19.9
	+ BeamformIt[7]	7.3	8.4	13.2	13.9
	+ BLSTM MVDR	**5.9**	**5.3**	**9.8**	**8.0**

CHiME-4 dataset contains both simulated data and real recordings, based on the Wall Street Journal (WSJ0) corpus. The sampling rate of CHiME-4 data is 16 kHz. In our experiments, we only use the simulated data for training and evaluation, because there are no clean speech labels for the real recordings. The simulated dataset consists of a 15.1-hour training set, a 2.9-hour development set, and a 2.3-hour test set of 6-channel noisy speech. The 1-channel track of CHiME-4 only selects one of the six channels as input for evaluation, while the 6-channel track uses all channels as input.

We evaluate our implementation of the T-F masking model and TasNet on the 1-channel track and the neural beamformer (denoted as BLSTM MVDR) on the 6-channel track. We also evaluate BeamformIt [49] as a baseline for the 6-channel track. For both tracks, we take the 5-th channel (CH5) clean signal as the reference for computing all metrics, including PESQ, STOI and SDR. The results are shown in Table 2. It shows that our ESPnet-SE models work well on both 1-channel and 6-channel tracks, with significant improvement over the baselines on all metrics.

Furthermore, we also evaluate the speech recognition performance of the ESPnet-SE models with an ASR model, which was pretrained on CHiME-4. The WER results on both simulation and real data are present in Table 3. It can be observed that a simple combination of the neural beamformer and the pretrained ASR model can bring a significant performance improvement, compared to a single ASR model on the 1-channel track. And it also achieves better performance than the baseline model (6-channel track) in [19].

4.2. Speech separation and recognition on WSJ0-2mix

WSJ0-2mix [27] is the current benchmark dataset for the single-channel speech separation task. This dataset consists of a 30-hour training set, a 10-hour validation set, and a 5-hour test set of single-channel two-speaker mixtures without noise and reverberation. In our experiments, we use the WSJ0-2mix data with 8-kHz sampling

---

[4]We found this is helpful for training ESPnet-SE models on data without reverberation, as shown in Table 4.

[5]https://zenodo.org/communities/espnet/

[6]https://github.com/espnet/espnet_model_zoo

[7]We use the BeamformIt toolkit implemented in https://github.com/xanguera/BeamformIt for evaluation.

Table 4: Performance (PSEQ / STOI / SDRi / WER) on the WSJ0-2mix dataset. WERs are evaluated with the same ASR model.

Model	Dev				Eval			
	PESQ	STOI	SDRi (dB)	WER (%)	PESQ	STOI	SDRi (dB)	WER (%)
Mixture	2.00	0.72	-	-	2.01	0.74	-	-
uPIT-PSM-relu [6]	-	-	9.4	-	-	-	9.4	-
uPIT-PSM-relu	2.96	0.89	10.3	31.9	2.92	0.90	10.2	29.1
+ speed perturb	3.04	0.90	10.6	29.2	2.99	0.90	10.6	26.0
Conv-TasNet [22]	-	-	-	-	-	-	15.6	-
Conv-TasNet	3.38	0.95	17.1	17.7	3.30	0.95	16.0	19.1
+ speed perturb	3.46	0.95	17.9	16.1	3.39	0.96	17.2	16.7
DPRNN [23]	-	-	-	-	-	-	19.0	-
DPRNN	3.43	0.95	17.8	17.2	3.42	0.97	17.9	16.1

Table 5: Performance (PSEQ / STOI / SDR / WER) on the spatialized WSJ0-2mix dataset.

Model	Condition	Dev			Eval			
		PESQ	STOI	SDR (dB)	PESQ	STOI	SDR (dB)	WER (%)
Original Mixture	anechoic	2.00	0.72	0.16	2.01	0.74	0.15	-
2-ch Deep clustering [41]		-	-	-	-	-	12.9	-
IRM / IBM [41]		-	-	-	-	-	12.7 / 13.5	-
2-ch BLSTM MVDR		**3.48**	**0.97**	**20.6**	**3.43**	**0.97**	**20.5**	34.4
Original Mixture	reverberant	1.83	0.64	-0.29	1.81	0.66	-0.30	-
8-ch Oracle MCWF [41]		-	-	-	-	-	10.9	-
IRM / IBM [41]		-	-	-	-	-	11.9 / 12.7	-
8-ch BLSTM MVDR		2.45	0.81	10.19	2.40	0.82	10.05	70.3
+ Nara-WPE [50]		**2.67**	**0.83**	**11.55**	**2.67**	**0.84**	**11.94**	56.7
+ DNN-WPE[10]		2.17	0.75	4.90	2.18	0.78	5.33	-

rate for training and evaluation. With this benchmark, plenty of speech separation methods have been proposed in recent years, including deep clustering [27], PIT [24], TasNet [21, 22], and DPRNN [23].

Among these methods, we compare our implementation of three representative methods in the time-frequency domain (uPIT [6]) and time domain (Conv-TasNet [22] and DPRNN) to those from prior studies. The results are shown in Table 4. As we can see, our implementations achieve promising results on the WSJ0-2mix, with a performance comparable to originally reported.

Interestingly, we also find that speed perturbation [51], a broadly used method for data augmentation in ASR, is also helpful in training the ESPnet-SE model. By applying speed perturbation with speed factors of 0.9, 1.0 and 1.1, we can observe an absolute SDR improvement of 1.2 dB on the TasNet model and 0.4 dB on the T-F masking model.

In addition, we also evaluate these well-trained speech separation models with an ESPnet-ASR model pretrained on WSJ. The WER of the pretrained ASR model on WSJ Dev93 and Eval92 are 9.9% and 7.0% respectively. By simply combining the ESPnet-SE model and ASR model, we obtain a surprisingly good performance on the WSJ0-2mix dataset. As shown in the last column in Table 4, the combination of Conv-TasNet and ASR models achieves 16.7% WER on the evaluation set, and the DPRNN achieves better performance with 16.1% WER.

4.3. Speech separation on spatialized WSJ0-2mix

The spatialized WSJ0-2mix [41] is a multi-channel speech separation dataset, extended from the original WSJ0-2mix dataset. A room

impulse response (RIR) generator[8] based on the image method [52] is used to spatialize the WSJ0-2mix dataset. It contains an anechoic and a reverberant version, both of which consists of 8-channel two-speaker mixtures. In our experiments, we use the spatialized WSJ0-2mix data with 8-kHz sampling rate for training and evaluation.

We evaluate our implementation of the neural beamformer (denoted as BLSTM MVDR) on both anechoic and reverberant conditions. For evaluation on both anechoic and reverberant data, we use the anechoic signal of each speaker at the first channel as the reference, and the results are shown in Table 5. We can observe that in the anechoic condition, our implemented neural beamformer outperforms the prior work [41], with more than 7dB SDR improvement. In the reverberant condition[9], our model still achieves a good performance on a par with the prior work [41]. And with an additional WPE module for dereverberation, the performance can be further improved.

Finally, we also evaluate the performance with ASR integration. The same ASR model trained on WSJ is used for ASR evaluation. The results are present in the last column in Table 5.

4.4. Joint training of SE and ASR

As illustrated in Fig. 1, our entire system could be optimized with the final ASR loss, while the SE loss could also be an additive option to train the same architecture. Here, we showed two example experiments on the WSJ0-2mix dataset trained with both SE (speech separation here) and ASR loss. The results are shown in Table 6. The

---

[8]https://github.com/ehabets/RIR-Generator
[9]Note that the results in the reverberant condition is not fully comparable, because a different reference signal is used in [41].
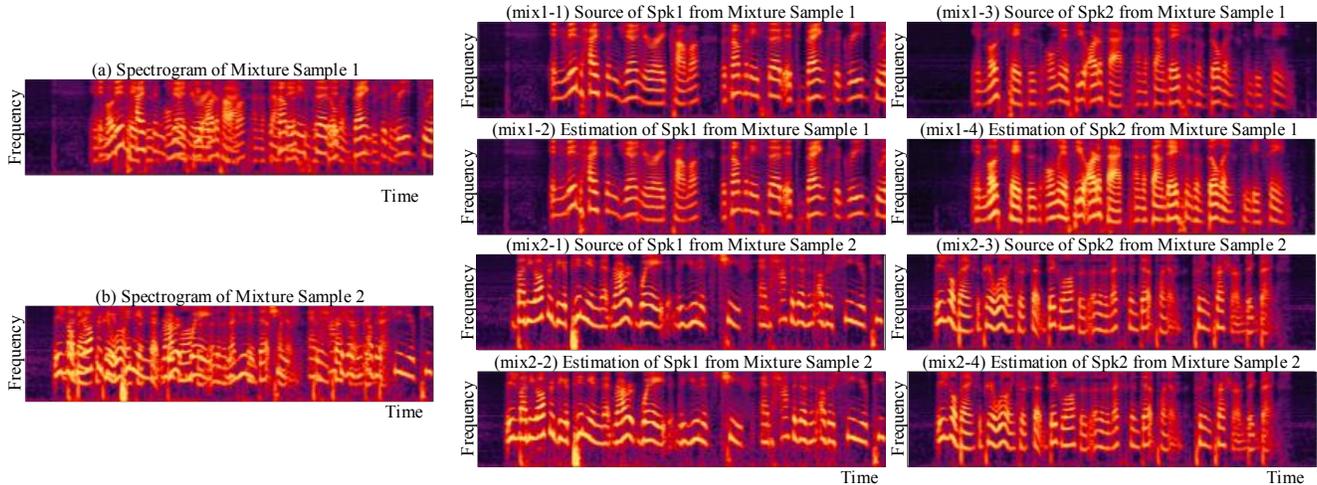[10]The DNN-WPE module is still under development.

(a) Spectrogram of Mixture Sample 1

(mix1-1) Source of Spk1 from Mixture Sample 1

(mix1-3) Source of Spk2 from Mixture Sample 1

(mix1-2) Estimation of Spk1 from Mixture Sample 1

(mix1-4) Estimation of Spk2 from Mixture Sample 1

(mix2-1) Source of Spk1 from Mixture Sample 2

(mix2-3) Source of Spk2 from Mixture Sample 2

(b) Spectrogram of Mixture Sample 2

(mix2-2) Estimation of Spk1 from Mixture Sample 2

(mix2-4) Estimation of Spk2 from Mixture Sample 2

Fig. 3: Visualization of two same-gender mixtures from the WSJ0-2mix test set. Mixture 1 consists of two males and the average SDR of this sample is 18.54 dB, while Mixture2 consists of two females, with the average SDR of 12.88 dB.

Table 6: The performance on the WSJ0-2mix datasets with joint training of E2E-SE and ASR module.

Model	Eval		
(SE + ASR)	SDRi	CER	WER
TF-masking + Transformer	8.5	18.6	33.8
Conv-TasNet + Transformer	14.7	8.1	15.8

model we use is the same as each module of T-F masking or Conv-TasNet separation network and ASR model used in section 4.2, while it is directly trained with WSJ0-2mix rather than WSJ from scratch.

From the results we could observe that, the joint optimization works well for the downstream application. With the training only on the WSJ0-2mix, comparable or better WER has been obtained compared with the strong single-speaker ASR trained on the whole WSJ dataset (see in Table 4). However, the separation performance gets worse than the isolated separation model. This tendency is also observed from some other works [14, 18].

In addition, we are working on the end-to-end joint training with ASR loss only. Previous studies [15, 53, 54] have shown promising results in this direction, and all of them got the implementation with ESPnet. Some ongoing work by us is to reproduce these works. It is optimistic to make it work since the ESPnet-SE shares the similar design ideas and the bases with ESPnet.

4.5. Visualization of enhanced spectrogram

In this section, we visualize the spectrograms of both input and enhanced signals to demonstrate the power of our ESPnet-SE model, which can be easily generated within the ESPnet-SE recipe. Here we adopt the well-trained DPRNN model for speech separation. We randomly selected two same-gender speech mixtures from the WSJ0-2mix test set. The spectrograms of the mixed signals, the corresponding clean source signals, and the estimated source signals are shown in Fig. 3. As we can see, our implemented DPRNN model can generate very good estimates of the source signals from the mixture.

5. CONCLUSIONS

In this paper, we present an end-to-end speech enhancement and separation toolkit, named ESPnet-SE, which can be tightly integrated with ASR. The toolkit is an extension of the open-source toolkit ESPnet, and supports various state-of-the-art speech enhancement and separation models with rich features. It also provides plenty of all-in-one recipes with a unified stage-by-stage processing pipeline, including data preparation, feature extraction, model training and evaluation. A wide range of benchmark datasets are covered in the current toolkit. A core feature that differentiates this toolkit from others is the capability of ASR integration. With the unified design in ESPnet-SE, we can easily integrate speech enhancement with ASR, such as model evaluation with the ASR criterion and joint optimization with ASR.

In the future, we will support more speech enhancement and separation models, such as MIMO-Speech [15], which is tightly integrated with ASR. We will also support target-speaker speech extraction models like speaker beam [55] in the future. In addition, we will work on a more flexible design to enable the integration of external E2E-SE models with ESPnet-ASR.

6. ACKNOWLEDGEMENT

7. REFERENCES

[1] Philipos C Loizou, *Speech enhancement: theory and practice*, CRC press, 2013.

[2] E. Colin Cherry, "Some experiments on the recognition of speech, with one and with two ears," *Journal of the Acoustical Society of America*, vol. 25, no. 5, pp. 975–979, 1953.

[3] Harald Gustafsson, Sven E Nordholm, and Ingvar Claesson, "Spectral subtraction using reduced delay convolution and

adaptive averaging," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 8, pp. 799–807, 2001.

[4] Emmanuel Vincent, Tuomas Virtanen, and Sharon Gannot, *Audio source separation and speech enhancement*, John Wiley & Sons, 2018.

[5] Reinhold 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.

[6] Morten Kolbaek, Dong Yu, Zheng Hua Tan, and Jesper Jensen, "Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks," *IEEE/ACM Trans. ASLP.*, vol. 25, no. 10, pp. 1901–1913, 2017.

[7] Shoji Makino, Te-Won Lee, and Hiroshi Sawada, *Blind speech separation*, vol. 615, Springer, 2007.

[8] Shinji Watanabe et al., "ESPnet: End-to-end speech processing toolkit," in *Proc. ISCA Interspeech*, 2018, pp. 2207–2211.

[9] Ethan Manilow, Prem Seetharaman, and Bryan Pardo, "The northwestern university source separation library," in *Proc. International Society for Music Information Retrieval (ISMIR)*, 2018, pp. 297–305.

[10] Zhaoheng Ni and Michael I Mandel, "Onssen: an open-source speech separation and enhancement library," *arXiv preprint arXiv:1911.00982*, 2019.

[11] Fabian-Robert Stöter, Stefan Uhlich, Antoine Liutkus, and Yuki Mitsufuji, "Open-Unmix – a reference implementation for music source separation," *Journal of Open Source Software*, vol. 4, no. 41, pp. 1667, 2019.

[12] Manuel Pariente et al., "Asteroid: The PyTorch-based audio source separation toolkit for researchers," in *Proc. ISCA Interspeech*, 2020, pp. 2637–2641.

[13] Adam Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *NeurIPS*, 2019, pp. 8026–8037.

[14] Aswin Shanmugam Subramanian, Xiaofei Wang, Murali Karthick Baskar, Shinji Watanabe, Toru Taniguchi, Dung Tran, and Yuya Fujita, "Speech enhancement using end-to-end speech recognition objectives," in *Proc. IEEE WASPAA*, 2019, pp. 229–233.

[15] Xuankai Chang, Wangyou Zhang, Yanmin Qian, Jonathan Le Roux, and Shinji Watanabe, "MIMO-Speech: End-to-end multi-channel multi-speaker speech recognition," in *Proc. IEEE ASRU*, 2019, pp. 237–244.

[16] Aswin Shanmugam Subramanian, Chao Weng, Meng Yu, Shi-Xiong Zhang, Yong Xu, Shinji Watanabe, and Dong Yu, "Far-field location guided target speech extraction using end-to-end speech recognition objectives," in *Proc. IEEE ICASSP*, 2020, pp. 7299–7303.

[17] Zhuo Chen, Takuya Yoshioka, Liang Lu, Tianyan Zhou, Zhong Meng, Yi Luo, Jian Wu, Xiong Xiao, and Jinyu Li, "Continuous speech separation: Dataset and analysis," in *Proc. IEEE ICASSP*, 2020, pp. 7284–7288.

[18] Thilo von Neumann, Christoph Boeddeker, Lukas Drude, Keisuke Kinoshita, Marc Delcroix, Tomohiro Nakatani, and Reinhold Haeb-Umbach, "Multi-talker asr for an unknown number of sources: Joint training of source counting, separation and asr," in *Proc. ISCA Interspeech*, 2020, pp. 3097–3101.

[19] Emmanuel Vincent, Shinji Watanabe, Aditya Arie Nugraha, Jon Barker, and Ricard Marxer, "An analysis of environment,

microphone and data simulation mismatches in robust speech recognition," *Computer Speech & Language*, vol. 46, pp. 535–557, 2017.

[20] Keisuke Kinoshita et al., "A summary of the reverb challenge: state-of-the-art and remaining challenges in reverberant speech processing research," *EURASIP Journal on Advances in Signal Processing*, vol. 2016, no. 1, pp. 7, 2016.

[21] Yi Luo and Nima Mesgarani, "TasNet: time-domain audio separation network for real-time, single-channel speech separation," in *Proc. IEEE ICASSP*, 2018, pp. 696–700.

[22] Yi Luo and Nima Mesgarani, "Conv-TasNet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Trans. ASLP.*, vol. 27, no. 8, pp. 1256–1266, 2019.

[23] Yi Luo, Zhuo Chen, and Takuya Yoshioka, "Dual-path RNN: efficient long sequence modeling for time-domain single-channel speech separation," in *Proc. IEEE ICASSP*, 2020, pp. 46–50.

[24] Dong Yu, Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *Proc. IEEE ICASSP*, 2017, pp. 241–245.

[25] Jahn Heymann, Lukas Drude, and Reinhold Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," in *Proc. IEEE ICASSP*, 2016, pp. 196–200.

[26] Hakan Erdogan, John R Hershey, Shinji Watanabe, Michael I Mandel, and Jonathan Le Roux, "Improved mvdr beamforming using single-channel mask prediction networks," *Proc. ISCA Interspeech*, pp. 1981–1985, 2016.

[27] John R Hershey, Zhuo Chen, Jonathan Le Roux, and Shinji Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. IEEE ICASSP*, 2016, pp. 31–35.

[28] Takuya Yoshioka, Tomohiro Nakatani, Keisuke Kinoshita, and Masato Miyoshi, "Speech dereverberation and denoising based on time varying speech model and autoregressive reverberation model," in *Speech Processing in Modern Communication*, pp. 151–182. 2010.

[29] Mehrez Souden, Jacob Benesty, and Sofiène Affes, "On optimal frequency-domain multichannel linear filtering for noise reduction," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, no. 2, pp. 260–276, 2009.

[30] Harry L Van Trees, *Optimum array processing: Part IV of detection, estimation, and modulation theory*, John Wiley & Sons, 2004.

[31] Tomohiro Nakatani and Keisuke Kinoshita, "A unified convolutional beamformer for simultaneous denoising and dereverberation," *IEEE Signal Processing Letters*, vol. 26, no. 6, pp. 903–907, 2019.

[32] Yipeng Li and DeLiang Wang, "On the optimality of ideal binary time–frequency masks," *Speech Communication*, vol. 51, no. 3, pp. 230–239, 2009.

[33] Arun Narayanan and DeLiang Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *Proc. IEEE ICASSP*, 2013, pp. 7092–7096.

[34] Yuxuan Wang, Arun Narayanan, and DeLiang Wang, "On training targets for supervised speech separation," *IEEE/ACM Trans. ASLP.*, vol. 22, no. 12, pp. 1849–1858, 2014.

[35] Hakan Erdogan, John R Hershey, Shinji Watanabe, and

Jonathan Le Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. IEEE ICASSP*, 2015, pp. 708–712.

[36] Scott Wisdom, Efthymios Tzinis, Hakan Erdogan, Ron J Weiss, Kevin Wilson, and John R Hershey, "Unsupervised sound separation using mixtures of mixtures," in *ICML 2020 Workshop on Self-supervision in Audio and Speech*, 2020.

[37] Emmanuel Vincent, Rémi Gribonval, and Cédric Févotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, no. 4, pp. 1462–1469, 2006.

[38] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 7, pp. 2125–2136, 2011.

[39] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE ICASSP*, 2001, vol. 2, pp. 749–752.

[40] Yusuf Isik, Jonathan Le Roux, Zhuo Chen, Shinji Watanabe, and John R Hershey, "Single-channel multi-speaker separation using deep clustering," in *Proc. ISCA Interspeech*, 2016, pp. 545–549.

[41] Zhong-Qiu Wang, Jonathan Le Roux, and John R Hershey, "Multi-channel deep clustering: Discriminative spectral and spatial embeddings for speaker-independent speech separation," in *Proc. IEEE ICASSP*, 2018, pp. 1–5.

[42] Luca Cristoforetti, Mirco Ravanelli, Maurizio Omologo, Alessandro Sosi, Alberto Abad, Martin Hagmüller, and Petros Maragos, "The DIRHA simulated corpus," in *the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, 2014, pp. 2629–2634.

[43] Lukas Drude, Jens Heitkaemper, Christoph Boeddeker, and Reinhold Haeb-Umbach, "SMS-WSJ: Database, performance measures, and baseline recipe for multi-channel source separation and recognition," *arXiv preprint arXiv:1910.13934*, 2019.

[44] Joris Cosentino, Manuel Pariente, Samuele Cornell, Antoine Deleforge, and Emmanuel Vincent, "LibriMix: An open-source dataset for generalizable speech separation," *arXiv preprint arXiv:2005.11262*, 2020.

[45] Gordon Wichern et al., "WHAM!: extending speech separation to noisy environments," in *Proc. ISCA Interspeech*, 2019, pp. 1368–1372.

[46] Matthew Maciejewski, Gordon Wichern, Emmett McQuinn, and Jonathan Le Roux, "WHAMR!: Noisy and Reverberant Single-Channel Speech Separation," in *Proc. IEEE ICASSP*, 2019, pp. 696–700.

[47] LDC, *LDC Catalog: CSR-I (WSJ0) Complete*, University of Pennsylvania, 1993, `www.ldc.upenn.edu/Catalog/LDC93S6A.html`.

[48] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *Proc. IEEE ICASSP*, 2015, pp. 5206–5210.

[49] Xavier Anguera, Chuck Wooters, and Javier Hernando, "Acoustic beamforming for speaker diarization of meetings," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15, no. 7, pp. 2011–2021, 2007.

[50] Lukas Drude, Jahn Heymann, Christoph Boeddeker, and Reinhold Haeb-Umbach, "NARA-WPE: A Python package for weighted prediction error dereverberation in Numpy and Tensorflow for online and offline processing," in *13. ITG Fachtagung Sprachkommunikation (ITG 2018)*, 2018.

[51] Tom Ko, Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur, "Audio augmentation for speech recognition," in *Proc. ISCA Interspeech*, 2015.

[52] Jont B Allen and David A Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.

[53] Xuankai Chang, Wangyou Zhang, Yanmin Qian, Jonathan Le Roux, and Shinji Watanabe, "End-to-end multi-speaker speech recognition with transformer," in *Proc. IEEE ICASSP*, 2020, pp. 6129–6133.

[54] Aswin Shanmugam Subramanian, Xiaofei Wang, Shinji Watanabe, Toru Taniguchi, Dung Tran, and Yuya Fujita, "An investigation of end-to-end multichannel speech recognition for reverberant and mismatch conditions," *arXiv preprint arXiv:1904.09049*, 2019.

[55] Marc Delcroix, Katerina Zmolikova, Tsubasa Ochiai, Keisuke Kinoshita, Shoko Araki, and Tomohiro Nakatani, "Compact network for speakerbeam target speaker extraction," in *Proc. IEEE ICASSP*, 2019, pp. 6965–6969.