# Jointly Recognizing Speech and Singing Voices Based on Multi-Task Audio Source Separation

**Ye Bai***, **Chenxing Li***, **Hao Li, Yuanyuan Zhao, Xiaorui Wang**
Kuaishou Technology Co., Ltd, Beijing, China
baiye03@qq.com

## Abstract

In short videos and live livestreams, speech, singing voice, and background music often overlap and obscure each other. This complexity creates difficulties in structuring and recognizing the audio content, which may impair subsequent ASR and music understanding applications. This paper proposes a multi-task audio source separation-based ASR model called JRSV, which **J**ointly **R**ecognizes **S**peech and singing **V**oices. Specifically, the separation module separates the mixed audio into distinct speech and singing voice tracks while removing background music. The CTC/attention hybrid recognition module recognizes both tracks. Online distillation is proposed to improve the robustness of recognition further. A benchmark dataset is constructed and released to evaluate the proposed methods. Experimental results demonstrate that JRSV can significantly improve recognition accuracy on each track of the mixed audio.

## 1 Introduction

The audio signals recorded in live streams and short videos usually contain speech, singing voices, background music, and sound effects. These signals often overlap and obscure each other, which increases the difficulty of speech and lyrics recognition. However, the subsequent recommendation systems and searching engines need speech transcripts and lyrics. It is necessary to improve the accuracy of speech and lyrics recognition in these overlapping scenarios.

Conventionally, cascade systems are used to recognize multi-track speech from monaural audio [1]. However, the mismatch between the separated audio and the natural audio hurts the recognition performance. Moreover, previous separation models, such as deep clustering [2], permutation invariant training (PIT) [3], and TasNet [4], do not distinguish the type of the separated tracks. Extended PIT [5, 6, 7, 8, 9] and serialized output training (SOT) [10] recognize the mixed audio in an end-to-end way and show good performance. However, these methods also cannot distinguish the type of the recognized tracks. Besides, PIT-based methods meet permutation problem, which may confuse automatic speech recognition (ASR) models.

This paper proposes a unified model called JRSV to Jointly Recognize Speech and singing Voices. JRSV provides the types of audio tracks and recognizes the content of the speech and singing voices. This is the first time to investigate how to jointly separate and recognize speech and singing voices in overlapping scenes. The contributions are summarized as follows. (1) multi-task audio source separation (MTASS) [11, 12] -based JRSV is proposed to recognize the content of the speech and singing voices simultaneously. The MTASS module separates the mixed audio into a speech track and a singing voice track. It also removes the background music at the same time. As PIT-free in MTASS, JRSV avoids the permutation and selection problems. Then the ASR module recognizes the content of the two tracks. (2) We adopt two-stage training and employ online distillation to make the

*denotes equal contribution.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

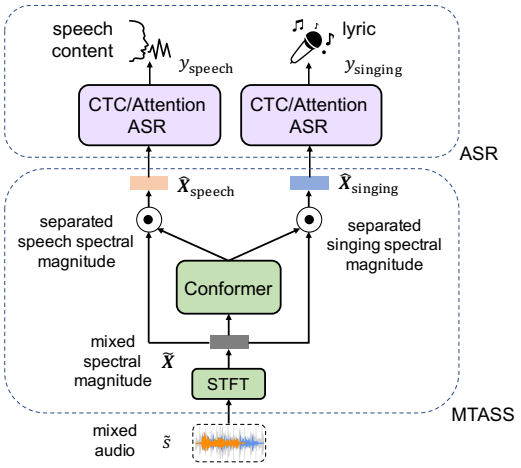


Figure 1: An overview of JRSV system. $\tilde{\mathbf{s}}$ denotes the mixed audio. $\widetilde{\mathbf{X}}$ denotes the mixed spectral magnitude. $\hat{\mathbf{X}}_{\text{speech}}$ and $\hat{\mathbf{X}}_{\text{singing}}$ denote the separated spectral magnitudes of speech and the singing voice. $y_{\text{speech}}$ and $y_{\text{singing}}$ denote the text sequence of speech and singing voice.

encoded representations of separated tracks approximate the representations of the clean audio track to improve the robustness of the model. (3) To evaluate the proposed methods, we build and release a benchmark dataset called Dual-Track Speech and singing Voice Dataset (DTSVD). The experimental results demonstrate that JRSV outperforms the cascade system by achieving a relative reduction of 41% in character error rates (CERs) for speech and 57% in CERs for singing voices.

## 2 Proposed methods

### 2.1 The MTASS module

We adopt the Conformer-based [13, 12] MTASS network to separate speech and singing voices. In detail, after a short-time Fourier transform (STFT), the spectral magnitude is fed into the Conformer-based separation network. In the last layer, the separated spectral magnitude of speech and singing voices are mapped out using two output layers. Three kinds of losses are applied to MTASS: magnitude-based mean absolute error (MAE) loss, discriminate separation loss [12], and consistency loss [14]. The detailed loss function is formulated as follows:

$$L_{\text{MTASS}} = L_{\text{mag}} - \lambda L_{\text{dis}} + \gamma L_{\text{cst}}, \tag{1}$$

where $\lambda$ and $\gamma$ are scale parameters to balance each loss. We set $\lambda$ to 0.1 and $\gamma$ to 0.3.

### 2.2 The ASR module

We employ the U2 (CTC/attention rescoring) structure [15, 16, 17] as the ASR model. The model consists of three parts. Given the separated spectral magnitude, the Conformer-based encoder encodes the acoustic representations. Then, the CTC decoder uses prefix beam search to generate candidates. And then, the attention decoder rescores the candidates to find the best hypothesis.

The ASR module is trained with CTC/attention joint loss:

$$L_{\text{ASR}} = \alpha L_{\text{CTC}} + (1-\alpha) L_{\text{att}}, \tag{2}$$

where $L_{\text{CTC}}$ is computed with a forward-backward algorithm [18] for the CTC decoder, and $L_{\text{att}}$ is computed with token-wise cross-entropy for the attention rescorer.

Inspired by [19, 20, 21], we propose online distillation to further improve the representation ability of the conformer encoder. Specifically, we attempt to make the encoded acoustic representations of the separated spectral magnitude approximate the encoded representations of the original clean spectral magnitude. The final loss is:

$$L_{\text{ASR}} = \alpha L_{\text{CTC}} + (1-\alpha) L_{\text{att}} + \beta L_{\text{distil}}, \tag{3}$$

where the $\alpha$ ($= 0.3$) and $\beta$ ($= 0.3$) are the hyperparameters to balance the values during training.

Table 1: Preliminary experiment: ASR models directly recognize the test sets of DTSSVD without separation. `speech` means the model trained with only speech data. `singing` means the model trained with only singing data. `multi` means the modeled trained with both speech and singing data.

| | CER% (Speech / Singing Voices) | | | | | |
|---|---|---|---|---|---|---|
| | Unmixed | Overlap 0.0 | Overlap 0.1 | Overlap 0.3 | Overlap 0.5 | Overlap 1.0 |
| `speech` | 6.3 / 88.2 | 48.1 / 245.7 | 49.4 / 233.5 | 58.5 /186.2 | 69.1 / 169.1 | 75.3 / 159.7 |
| `singing` | 86.6 / 9.7 | 185.5 / 95.6 | 206.3 / 94.3 | 167.6 / 89.5 | 150.8 / 90.6 | 145.1 / 91.6 |
| `multi` | 6.3 / 9.4 | 57.9 / 222.4 | 60.3 / 236.9 | 67.9 / 190.9 | 76.2 / 171.8 | 80.6 / 163.8 |

Table 2: The SDR improvments of MTASS models on the test sets with different overlap ratios.

| | | SDRi | | | | |
|---|---|---|---|---|---|---|
| Overlap Ratios | | 0.0 | 0.1 | 0.3 | 0.5 | 1.0 |
| `MTASS` | speech | 46.0 | 26.7 | 18.0 | 15.9 | 14.6 |
| | singing | 19.4 | 18.0 | 15.4 | 14.4 | 13.7 |
| + $L_{dis}$ | speech | 44.8 | 27.0 | 18.5 | 16.5 | 15.1 |
| | singing | 19.1 | 17.6 | 15.0 | 14.1 | 13.3 |
| + $L_{cst}$ | speech | **49.0** | **27.3** | **18.7** | **16.6** | **15.4** |
| | singing | **19.7** | **18.2** | **15.6** | **14.6** | **14.0** |

### 2.3 Two-stage training

Counter-intuitively, we find that jointly optimizing the $L_{\text{MTASS}}$ and $L_{\text{ASR}}$ does not improve the performance. Even the system cannot converge when trained from scratch. We analyze two reasons: (1) The spectral magnitude of the mixed audio contains ambiguous information, which obstacles the optimization procedure. (2) the high-level $L_{\text{ASR}}$ is not compatible with the low-level $L_{\text{MTASS}}$. Therefore, we propose a curriculum learning [22]-based two-stage training procedure: 1) we train the MTASS module; 2) the ASR model is trained, and the MTASS module is fixed during training. We will discuss the reason why joint training does not improve performance in subsection 4.4.

## 3 DTSSV dataset

DTSSV dataset is built based on AISHELL-1 [23], OpenSinger [24], and MusDB18 [25]. In MusDB18, only background music is adopted. We randomly sample speech audio, a singing voice, and a background music segment and mix them with different overlaps. In detail: (1) Normalize the amplitude of $\mathbf{s}_{\text{speech}}$, $\mathbf{s}_{\text{sing}}$, and $\mathbf{s}_{\text{music}}$. (2) Draw signal-to-noise ratios (SNRs) in decibels from uniform distributions: $\mathcal{U}(-10, 2)$ for $\mathbf{s}_{\text{speech}}$ and $\mathbf{s}_{\text{sing}}$, and $\mathcal{U}(-15, 2)$ for $\mathbf{s}_{\text{music}}$. (3) Mix the $\mathbf{s}_{\text{sing}}$ and the $\mathbf{s}_{\text{music}}$. Then randomly sample an overlap ratio from {1.0, 0.5, 0.3, 0.1, 0.0} and mix $\mathbf{s}_{\text{speech}}$. For the development set and the test set, we randomly select a singing voice and a music segment for each speech audio. For the training set, the three audio sources are mixed on-the-fly during training.

## 4 Experiments and analysis

### 4.1 Setup

For MTASS, the Conformer-based model [13, 12] consists of 16 Conformer blocks. The dimension of each block ($d_{\text{model}}$) is 256. The number of attention heads is 8. Relative positional encodings are used. The inner dimension of MLP ($d_{\text{ffn}}$) is 1024. The kernel size of the convolution is 33. The window length is 1024, with the FFT is 256. We use Adam optimizer [26] with a learning rate $= 1e-3$. All the MTASS models are trained for 200 epochs.

For CTC/attention hybrid ASR, the model first uses a 2 dimensional CNN for subsampling. 12 Conformer blocks as the encoder, and 6 Transformer decoder blocks as the rescorer. The dimension of each block is 256. The number of attention heads is 4. The inner dimension of MLP ($d_{\text{ffn}}$) is 2048. The kernel size of the convolution is 15. For the cascade system, we use the 80-dim FBANK features. For JRSV, the input dimension is 513. The optimizer is Adam [26], and we use the Noam learning rate schedule [27] with 10000 warm-up steps. All the ASR models are trained for 100 epochs. For the two-stage training for JRSV, we first load the pre-trained MTASS and then train the ASR module.

Table 3: The CERs of the cascade system on the test sets with different overlap ratios.

| | CER % (Speech / Singing Voices) | | | | | |
|---|---|---|---|---|---|---|
| Overlap Ratios | 0.0 | 0.1 | 0.3 | 0.5 | 1.0 | Avg. |
| `ASR` / `MTASS` | 9.2 / 28.4 | 12.8 / 29.6 | 24.3 / 32.5 | 29.3 / 35.9 | 31.3 / 33.8 | 21.4 / 32.1 |
| + $L_{dis}$ | 9.3 / 27.3 | 13.0 / 28.3 | 24.5 / 30.8 | 29.1 / 33.9 | 31.5 / 31.4 | 21.5 / 30.3 |
| + $L_{dis}$ + $L_{cst}$ | 9.0 / 25.2 | 12.7 / 26.5 | 23.8 / 29.1 | 28.4 / 31.9 | 30.6 / 29.2 | 20.9 / 28.4 |

Table 4: The CERs of the JRSV on the test sets with different overlap ratios.

| | CER % (Speech / Singing Voices) | | | | | |
|---|---|---|---|---|---|---|
| Overlap Ratios | 0.0 | 0.1 | 0.3 | 0.5 | 1.0 | Avg. |
| `JRSV-t` | 10.1 / 22.0 | 12.1 / 22.6 | 17.9 / 25.2 | 20.2 / 26.7 | 20.4 / 28.1 | 16.1 / 24.9 |
| `JRSV-f` | 8.2 / 13.1 | 9.7 / 13.8 | 14.1 / 14.7 | 15.9 / 15.5 | 16.6 / 16.5 | 12.9 / 14.7 |
| `JRSV-f-d` | **7.6 / 10.8** | **9.3 / 11.6** | **13.6 / 12.0** | **15.3 / 12.8** | **15.9 / 13.1** | **12.3 / 12.1** |

We evaluate the system in two aspects: separation performance and recognition accuracy. Signal-to-distortion ratio [28] improvements (SDRi) are used to evaluate the separation performance. For recognition accuracy, we compute the CERs for the speech and singing voice tracks.

### 4.2 Preliminary experiment: directly recognizing the mixture

The performance of the ASR model without MTASS tells the impacts of the complex acoustic conditions on ASR. In Table 1. `ASR-speech`, `ASR-sing`, and `ASR-multi` denote the ASR model trained on the speech data, singing data, and both speech and singing data, respectively. Models trained on the matched data perform well compared to the corresponding unmixed data. `ASR-multi` performs best on the unmixed speech and singing voices. However, all models fail to recognize the mixed audio. Because many speech contents are recognized and inserted, which causes insertion errors, the CERs of the singing voice are larger than 100%. This experiment demonstrates that the ASR model fails to recognize mixed audio in complex acoustic conditions directly.

### 4.3 Recognizing with the cascade system

The cascade system first separates the mixed audio into the speech track and the singing voice track and then recognizes with `ASR-multi`. First, we evaluate the performance of the MTASS models in Table 2. `MTASS` achieves a significant SDR improvement. The discriminative loss $L_{dis}$ and the consistent loss $L_{cst}$ can further improve the SDR. Then we evaluate the recognition performance in Table 3. $L_{dis}$ brings an improvement for sing voices. When using both $L_{dis}$ and $L_{cst}$, the system achieves the best performance.

### 4.4 Recognizing with JRSV

Comparing Table 4 to Table 3, `JSRV-f` with the frozen MTASS achieves significantly better performances than `ASR-multi` / `MTASS` cascade models. With online distillation, the performance is further improved. On average, the `JRSV-f-d` achieves a 41%/57% relative CER reduction compared with the best cascade system. Counter-intuitively, `JRSV-t` (trainable MTASS) does not achieve a good performance. We have searched many weights for $L_{\text{MTASS}}$ and $L_{\text{ASR}}$ but do not achieve a positive result. We analyze a possible reason that the MTASS module and the ASR module play different roles: the MTASS module processes the low-level features, and the ASR module processes the high-level semantic representations. The two objects are not compatible. The ASR loss influences the low-level feature extraction, which affects the performance.

## 5 Conclusions and future work

We propose JRSV to jointly recognize speech and singing voices. The MTASS module separates the mixed audio into distinct speech and singing voice tracks while removing background music. The CTC/attention hybrid recognition module recognizes both tracks. Online distillation is proposed to further improve recognition accuracy. A benchmark dataset is constructed. Experimental results demonstrate that the proposed methods can significantly improve recognition accuracy on each track of the mixed audio. In the future, we will analyze the reason why the joint optimization of $L_{\text{MTASS}}$ and $L_{\text{ASR}}$ does not bring better performance in more detail.

## References


[1] 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 *IEEE ICASSP*, pages 7284–7288, 2020.

[2] John R Hershey, Zhuo Chen, Jonathan Le Roux, and Shinji Watanabe. Deep clustering: Discriminative embeddings for segmentation and separation. In *IEEE ICASSP*, pages 31–35, 2016.

[3] Morten Kolbæk, Dong Yu, Zheng-Hua Tan, and Jesper Jensen. Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(10):1901–1913, 2017.

[4] Yi Luo and Nima Mesgarani. Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation. *IEEE/ACM transactions on audio, speech, and language processing*, 27(8):1256–1266, 2019.

[5] Dong Yu, Xuankai Chang, and Yanmin Qian. Recognizing Multi-Talker Speech with Permutation Invariant Training. In *Proc. Interspeech 2017*, pages 2456–2460, 2017.

[6] Shane Settle, Jonathan Le Roux, Takaaki Hori, Shinji Watanabe, and John R Hershey. End-to-end multi-speaker speech recognition. In *IEEE ICASSP*, pages 4819–4823, 2018.

[7] Xuankai Chang, Yanmin Qian, Kai Yu, and Shinji Watanabe. End-to-end monaural multi-speaker asr system without pretraining. In *IEEE ICASSP*, pages 6256–6260, 2019.

[8] Liang Lu, Naoyuki Kanda, Jinyu Li, and Yifan Gong. Streaming end-to-end multi-talker speech recognition. *IEEE Signal Processing Letters*, 28:803–807, 2021.

[9] Desh Raj, Liang Lu, Zhuo Chen, Yashesh Gaur, and Jinyu Li. Continuous streaming multi-talker asr with dual-path transducers. In *IEEE ICASSP*, pages 7317–7321, 2022.

[10] Naoyuki Kanda, Yashesh Gaur, Xiaofei Wang, Zhong Meng, and Takuya Yoshioka. Serialized Output Training for End-to-End Overlapped Speech Recognition. In *Proc. Interspeech 2020*, pages 2797–2801, 2020.

[11] Lu Zhang, Chenxing Li, Feng Deng, and Xiaorui Wang. Multi-task audio source separation. In *IEEE ASRU*, pages 671–678, 2021.

[12] Chenxing Li, Yang Wang, Feng Deng, Zhuo Zhang, Xiaorui Wang, and Zhongyuan Wang. Ead-conformer: a conformer-based encoder-attention-decoder-network for multi-task audio source separation. In *IEEE ICASSP*, pages 521–525, 2022.

[13] Sanyuan Chen, Yu Wu, Zhuo Chen, Jinyu Li, Chengyi Wang, Shujie Liu, and M. Zhou. Continuous speech separation with conformer. *IEEE ICASSP*, pages 5749–5753, 2021.

[14] Scott Wisdom, John R Hershey, Kevin Wilson, Jeremy Thorpe, Michael Chinen, Brian Patton, and Rif A Saurous. Differentiable consistency constraints for improved deep speech enhancement. In *IEEE ICASSP*, pages 900–904, 2019.

[15] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang. Conformer: Convolution-augmented Transformer for Speech Recognition. In *Proc. Interspeech 2020*, pages 5036–5040, 2020.

[16] Zhuoyuan Yao, Di Wu, Xiong Wang, Binbin Zhang, Fan Yu, Chao Yang, Zhendong Peng, Xiaoyu Chen, Lei Xie, and Xin Lei. WeNet: Production Oriented Streaming and Non-Streaming End-to-End Speech Recognition Toolkit. In *Proc. Interspeech 2021*, pages 4054–4058, 2021.

[17] Zhengkun Tian, Jiangyan Yi, Ye Bai, Jianhua Tao, Shuai Zhang, and Zhengqi Wen. One in a hundred: Selecting the best predicted sequence from numerous candidates for speech recognition. In *IEEE APSIPA ASC*, pages 454–459, 2021.

[18] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *ICML*, pages 369–376, 2006.

[19] Jinyu Li, Michael L. Seltzer, Xi Wang, Rui Zhao, and Yifan Gong. Large-Scale Domain Adaptation via Teacher-Student Learning. In *Proc. Interspeech 2017*, pages 2386–2390, 2017.

[20] Jiangyan Yi and Jianhua Tao. Distilling knowledge for distant speech recognition via parallel data. In *IEEE APSIPA ASC*, pages 170–175, 2019.

[21] Arun Narayanan, James Walker, Sankaran Panchapagesan, Nathan Howard, and Yuma Koizumi. Learning mask scalars for improved robust automatic speech recognition. In *IEEE SLT*, pages 317–323, 2023.

[22] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, pages 41–48, 2009.

[23] Hui Bu, Jiayu Du, Xingyu Na, Bengu Wu, and Hao Zheng. Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline. In *IEEE O-COCOSDA*, pages 1–5, 2017.

[24] Rongjie Huang, Feiyang Chen, Yi Ren, Jinglin Liu, Chenye Cui, and Zhou Zhao. Multi-singer: Fast multi-singer singing voice vocoder with a large-scale corpus. In *ACM MM*, pages 3945–3954, 2021.

[25] Zafar Rafii, Antoine Liutkus, Fabian-Robert Stöter, Stylianos Ioannis Mimilakis, and Rachel Bittner. The MUSDB18 corpus for music separation, December 2017.

[26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[28] E. Vincent, R. Gribonval, and C. Fevotte. Performance measurement in blind audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(4):1462–1469, 2006.