# CoSSD - An end-to-end framework for multi-instance source separation and detection

Shrishail Baligar
*Electrical Engineering and Computer Science*
*University of California, Merced*

Shawn Newsam
*Electrical Engineering and Computer Science*
*University of California, Merced*

***Abstract*—We address the problem of separating classes of sound sources given 1) an audio mixture and 2) a conditional that represents the desired source to be separated. The proposed Conditional Source Separation and Detection (CoSSD) network operates on a defined set of classes during training and inference. We show that our model is versatile in that it can be repurposed for various separation tasks. We demonstrate its capability for conditional audio source separation on the NIGENS dataset of general sound events. We also show that, without modification, the model can perform speech and non-speech component separation using mixtures from the LibriSpeech and FSD50k datasets. Finally, a key feature of the proposed CoSSD is that it performs *detection* in addition to separation, making it a practical and unified solution for query-based audio analysis.**

***Index Terms*—Source separation, detection, audio scene analysis, speech, deep learning**

## I. Introduction

Rustling leaves, AC compressors, and rubbing hands all sound very similar. Likewise, a monophonic phone ringtone and a smoke alarm, or footsteps and knocking on a door sound similar. These examples are from different "classes" of sources and events which highlights the challenge in separating and identifying sources that sound similar due to inter-class similarity. Conversely, a fan in air sounds different than in water. This is the same source, fan, but different events, where the mechanisms involved, air vs. water, are different. While the fan-in-water might be considered as the class *propeller*, such fine class labelling can lead to class-explosion which brings about challenges especially for supervised learning. Sounds with very different events may have same source labels, and different sources may sound similar. Indeed, the space of sound events and sources is very large and any form of meaningful analysis like source separation or detection in practical applications requires constraining this space to a tractable set of classes.

This paper investigates jointly separating and detecting specific sources of sound from mixtures. The proposed supervised method is trained and performs inference on a defined set of sound sources with the kinds of inter-class similarity and intra-class variation discussed above. The model also *detects* whether the specific query sound source is present in the mixture. This allows the model to separate a finite set of sources in mixtures from an open world since the separated output can just be ignored if the source is not detected. We believe this is a unique feature of our model and represents progress towards open-world source separation. We make the following novel contributions:

- We propose a novel problem setting of jointly performing conditional separation and detection of multiple sources of interest in an audio mixture for non-speech and speech problems.
- We present an end-to-end multiple-instance Conditional Source Separation and Detection (CoSSD) Network to solve the proposed problem and compare it with two approaches.
- Our CoSSD model can be conditioned in two ways, one-hot vectors and audio waveforms, to separate and detect specific sound sources at varying levels of SNR.
- CoSSD is available in three model sizes that represent the trade-off between performance and resource needs when considering real-time application.

## II. Related Work

CoSSD has a highly modular design, partly inspired by Conv-TasNet, a blind source separation framework proposed by Luo and Mesgarani [1]. CoSSD consists of a conditional embedding, an encoder and decoder, and a detection network. It also has a Temporal Convolutional Network based masker component from the Conv-TasNet that produces a single mask. The encoder and decoder for CoSSD are STFT filterbanks that perform better than trainable 1D-Conv layers as described later in Table 3.

While we are unaware of work on joint separation and detection, especially in the context of inter-class similarity and intra-class variation, we identified two related conditional source separation models. Gfeller et al. [2] propose a one-shot learning based model to separate sound classes that the model has never seen before using a U-Net [3] and a FiLM [4] based conditional in the waveform domain. The strength of this approach is that it does not need labeled training data and it can separate multiple instances of sound classes in the mixture. That is, it can separate either sound $A$ or sound $B$ in a mixture of $A$ and $B$ depending on the conditional. However, the paper does not discuss or address inter-class similarity and intra-class variance since, like most one-shot methods, the framework has no concept of class. Thus, it might be challenged if the conditional is from the same class as the

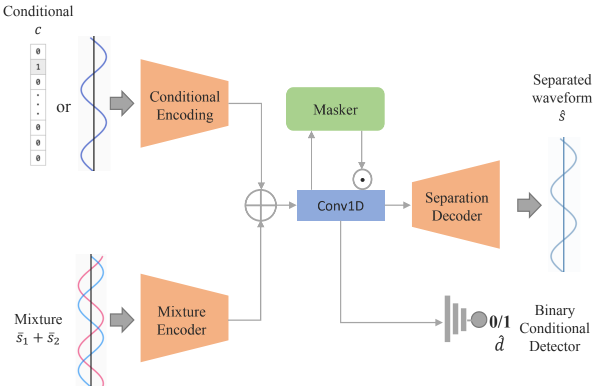

Fig. 1: Overview of CoSSD. The input conditional is a one-hot vector or an audio waveform representing the class.

source but sounds a bit different. This is one of one-shot learning's main drawbacks thus keeping supervised methods and large datasets relevant [5], [6]. The problem proposed by Gfeller et al. [2] is also different from ours. We perform separation and detection jointly while being robust to unseen class mixtures, with applications in both general sound source and speech foreground and background separation.

Kong et al. [7] is another related work which applies a Sound Event Detection (SED) multi-class classifier before the separation model in order to separate specific sounds from a mixture. This model works in the time-frequency domain through spectrograms. Again, this work focuses on a different problem in that it only seeks to separate one source in a mixture, the one detected by the classifier. In contrast, CoSSD can separate multiple instances of sound sources in a mixture which makes it suitable for a wide range of separation tasks in the general sound and speech domains. Our model is further distinguished in that it is trained in an end-to-end fashion and operates in the waveform domain. Other related work includes Ochiai et al. [8] who present event separation models albeit with less noisy mixtures; Turpault et al. [9] who propose sound separation as a pre-processing step to improve SED; and Tzinis et al. [10] who show improved separation performance by conditioning on the predictions of SED.

## III. Method

CoSSD takes two inputs and provides two outputs. One input is the mixture $m = \bar{s}_1 + \bar{s}_2$ that may or may not contain the source of interest to separate. The second input is a conditional vector $c$ that informs or queries the model which source to separate. This conditional can be a one-hot vector or an audio sample representing a specific sound-source class. The outputs are the binary output $\hat{d}$ of a presence/absence detector and the separated audio waveform $\hat{s}$. Only when $\hat{d}$ is 1 do we consider $\hat{s}$. During training, $\hat{s}$ is set to a *0*-vector when the conditional is absent in the mixture $m$. During inference, the network will output $\hat{s}$ with very low amplitude $\mathcal{O}(10^{-3})$ noise when the conditional is absent which can be ignored since the detector's output $\hat{d}$ will be *0*.

| | | | | |
|---|---|---|---|---|
| Encoder | *n_filters* | 256 | 256 | 192 |
| | *kernel_size* | 128 | 128 | 128 |
| | *stride* | 64 | 64 | 64 |
| Decoder | *n_filters* | 514 | 514 | 386 |
| | *kernel_size* | 128 | 128 | 128 |
| | *stride* | 64 | 64 | 64 |
| Masker | *n_blocks* | 8 | 4 | 3 |
| | *n_repeats* | 3 | 2 | 2 |
| | *hid_ch* | 512 | 128 | 64 |
| | *skip_ch* | 128 | 128 | 64 |
| | *kernel_size* | 3 | 3 | 3 |
| One-hot Encoding | *fc1* | $14 \rightarrow 64$ | $14 \rightarrow 64$ | $14 \rightarrow 64$ |
| | *fc2* | $64 \rightarrow 499$ | $64 \rightarrow 499$ | $64 \rightarrow 499$ |
| | *Conv1D_ch* | $1 \rightarrow 258$ | $1 \rightarrow 258$ | $1 \rightarrow 258$ |
| Binary Detector | *fc_d1* | $499 \rightarrow 32$ | $499 \rightarrow 32$ | $499 \rightarrow 32$ |
| | *fc_d2* | $32 \rightarrow 1$ | $32 \rightarrow 1$ | $32 \rightarrow 1$ |
| Model Size: | | 5.8M | 1.3M | 600k |

TABLE I: Various hyperparameters of CoSSD leading to three model sizes. The hyperparameters for 1D-Conv vs. STFT based encoder-decoders translate directly as: *n_filters*: Determines the length of the STFT filters before windowing, *kernel_size*: Length of the filters (i.e the window), *stride*: Stride of the convolution (hop size).

### A. Data Pre-Processing

The input audio mixture $m$ is a 2-second audio waveform sampled at 16kHz. During training, $m$ is a pair of disjoint classes of sound sources with SNR = 0, assuming the target class to separate is signal and the other is noise. The conditional $c$ is a one-hot vector or another 2-sec audio waveform representing the class we are looking to separate in the mixture. (The networks for these two cases differ only in the conditional encoder.) The output $\hat{s}$ is the expected separation which is a 2-sec audio waveform at 16kHz. For the detection component, we train on both present $\hat{d}$=*1* and absent $\hat{d}$=*0* scenarios, depending on whether the conditional class is present in the mixture or not.

### B. Model Architecture

The architecture of CoSSD is illustrated in Figure 1. In our initial attempts, the encoder, decoder, and masker aspects of CoSSD were inspired by the Conv-TasNet model. However, we replaced the 1D Conv layers in the encoder and decoder with STFT filterbanks after noticing a significant improvement in the detection accuracy using the latter as seen in Table 3. Further, we add our conditional embedding layer for the one-hot conditional vector using fully connected layers and a 1D Conv layer. The weights of the waveform-conditional variant are shared with the input mixture's encoder. Finally, the detector component of the architecture is a binary classifier with sigmoid activation. Table 1 contains details of various hyperparameters in the different modules of CoSSD. The modular design of CoSSD allows for a highly interpretable and flexible architecture. This makes it easy to identify aspects that influence the separation quality and detection accuracy. The modularity also helps target specific parts of CoSSD to manipulate the number of model parameters. Table 1 shows three model sizes of CoSSD: 5.8M, 1.3 M, and 600k parameters. The smaller architectures may be favored by resource-constrained environments with real-time applications. All CoSSD models can be used for real-time application by staggering the input with tolerable latency-delta and overlap.

The encoder and decoder consist of Short-time Fourier Transform (STFT) filterbanks. The encoder performs STFT over the input and conditional waveform (for wavform as a conditional only). The decoder performs I-STFT over the intermediate hidden representation involving the masker and Conv1D layers. The output of the encoder is concatenated with the conditional embedding before being passed into a Conv1D layer. The output of this Conv1D layer is fed to the masker, which then outputs a single mask to filter the appropriate sound source in the latent space. This filtering is an element-wise multiplication of the mask and the output of the Conv2D layer. The hyperparameters for the encoder and decoder are interchangeable for the CoSSD with and without learnable encoder-decoders. For STFT-based encoder-decoders, which is the main model in this paper, the hyperparameters translate directly as seen in Table 1.

The major hyperparameters in the masker that influence the trade-off between model-parameter size and performance are the number of TCN blocks, their repetition, and the hidden channels.

The detector module predicts the presence or absence of the class represented by the conditional vector in the input mixture. This module is a fully connected layer that sits on top of the element-wise multiplication Conv1D layer and the mask, as shown in Figure 1. In the results below, we report the accuracy of the detector module in addition to the quality of the separation. This gives an indication of what happens when the mixture is an arbitrary audio clip in the wild that might not contain the target source.

## IV. Experiments

We use the same process for generating the training and evaluation samples for our experiments. We standardize 2-sec audio clips from disjoint classes to -12 peak dBFS before before pairing them to form the mixtures. This standardization is necessary since some of the datasets used in our experiments are from crowd-sourced repositories like Freesound [11]. Peak normalization also allows us to create SNR-specific mixtures of target source and other sounds to evaluate how our model performs at different noise levels. Our work does not aim at denoising tasks, but when focusing on extracting one source from a mixture of sounds, we treat the other source(s) as noise. We create training sample mixtures at SNR=0, which means the sources have equal levels. During the evaluation, we test CoSSD over three different SNRs $\{0, 6, 12\}$ as seen in Table 2. Such evaluation helps understand the model's strengths and weaknesses over tasks of varying difficulty.

### A. Training

For the one-hot conditional model, training samples are a 2-sec mixture, a one-hot vector representing a source class, the expected 2-sec audio separation, and a binary detection, 0 or 1. For the waveform conditional model, the one-hot vector is replaced by a waveform that represents the source class. Given that the model performs two tasks, separation and detection, we consider the optimization problem as multi-task learning with a loss function $L$ that has two components, a synthesis loss $L_{synth}$ for separation, and binary cross-entropy loss for detection $L_{det}$. We initially investigated an MSE loss for $L_{synth}$ since we expected the phase information to carry-over. However, we later found the SI-SDR loss [12] performs slightly better and used it to train the models in the experiments below. The final loss function for the multi-task learning is $L = L_{synth} + \lambda\ L_{det}$ We experimented with values of $\lambda$ between $\{0.5,1\}$ but did not find any performance difference and so we set it to 1. We train the models for up to 50 epochs with a learning rate of 0.0001 and a batch size of 24.

### B. Datasets

We use NIGENS (Neural Information processing group GENeral Sounds) [13] dataset to develop CoSSD for separating and detecting specific sound sources. This dataset has the advantage of finer clip-level annotation. The NIGENS dataset has fourteen classes $\{$alarm, crying baby, crash, barking dog, running engine, burning fire, footsteps, knocking on the door, female and male speech, female and male scream, ringing phone, piano$\}$. This dataset has a small average number of samples per class; therefore, we decide to ignore the finer annotation and go ahead with slightly noisy labeling by assuming the label's association to the entire clip as a trade-off. This gives an average of 450 2-sec clips per class. We create a training and evaluation set by randomly sampling and pairing disjoint sound source classes. While this allows for a large number of training samples, we limit our training set to 550,000 pairs for the NIGENS dataset, and 25,500 validation pairs.

Finally, as a special case, we use CoSSD for separating foreground and background in a mixture of speech and non-speech sounds. This case should not be confused with the traditional multi-source noise or white noise and speech. We create these mixtures for training using 1000-hours of the LibriSpeech ASR speech corpus [14] and specific sounds from the FSD50k [15] dataset.

### C. Results

We do not compare the performance of CoSSD to other methods since we are not aware of any that perform joint separation and detection in specific-sound sources and speech tasks. However, focusing just on separation, we observe that CoSSD exhibits the improved performance expected of supervised methods over one-shot learning [2]. We evaluate CoSSD on the NIGENS dataset for both the one-hot and waveform conditioned cases and on the LibriSpeech+FSD50k datasets for the one-hot case. In Table 2 and Table 3, we show results of the proposed CoSSD framework and compare it with two alternative approaches. We use three quantitative metrics to measure the quality of the separation: SDR (Signal to Distortion Ratio), SI-SDR (Scale Invariant Signal to Distortion Ratio), and STOI (Short-Time Objective Intelligibility) [12], [16]. Higher values correspond to better performance. To make our investigation comprehensive, we present the results for three model sizes and for three SNR values $\{0, 6, 12\}$. Note that the results provide comprehensive detection accuracies

|  | SNR | Model Parameters |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
|  |  | 5.8M |  |  |  |  | 1.3M |  |  |  |  | 600k |  |  |  |  |
|  |  | SDR | SI-SDR | STOI | Det Acc |  | SDR | SI-SDR | STOI | Det Acc |  | SDR | SI-SDR | STOI | Det Acc |  |
| (A) | 0 | 12.31 | 10.81 | 0.67 | 0.84p | 0.94a | 10.44 | 8.70 | 0.66 | 0.85p | 0.94a | 7.70 | 5.94 | 0.62 | 0.78p | 0.90a |
|  | 6 | 15.12 | 13.69 | 0.72 | 0.87p | 0.93a | 13.16 | 11.66 | 0.71 | 0.87p | 0.95a | 10.62 | 9.15 | 0.68 | 0.86p | 0.91a |
|  | 12 | 17.32 | 15.94 | 0.76 | 0.87p | 0.92a | 15.60 | 14.40 | 0.75 | 0.90p | 0.94a | 13.05 | 11.84 | 0.72 | 0.86p | 0.91a |
| (B) | 0 | 12.17 | 10.58 | 0.66 | 0.83p | 0.95a | 10.36 | 8.99 | 0.64 | 0.81p | 0.88a | 7.67 | 5.68 | 0.62 | 0.79p | 0.76a |
|  | 6 | 15.03 | 13.46 | 0.71 | 0.88p | 0.94a | 12.92 | 11.64 | 0.70 | 0.86p | 0.88a | 10.45 | 8.76 | 0.68 | 0.87p | 0.78a |
|  | 12 | 17.18 | 15.56 | 0.75 | 0.90p | 0.93a | 15.09 | 14.05 | 0.74 | 0.89p | 0.88a | 13.05 | 11.74 | 0.73 | 0.875p | 0.79a |
| (C) | 0 | 20.14 | 19.61 | 0.94 | 1.00p | 0.993a | 19.14 | 18.62 | 0.93 | 1.00p | 0.93a | 17.57 | 17.04 | 0.92 | 1.00p | 0.99a |
|  | 6 | 23.08 | 22.66 | 0.96 | 1.00p | 0.991a | 22.15 | 21.77 | 0.96 | 1.00p | 0.991a | 20.70 | 20.24 | 0.95 | 1.00p | 0.99a |
|  | 12 | 25.66 | 25.31 | 0.98 | 1.00p | 0.988a | 24.79 | 24.49 | 0.97 | 1.00p | 0.988a | 23.29 | 22.85 | 0.97 | 1.00p | 0.99a |
| (D) | 0 | 5.28 | 3.65 | 0.45 | 1.00p | 0.993a | 4.37 | 2.55 | 0.43 | 1.00p | 0.96a | 2.31 | 0.50 | 0.39 | 1.00p | 0.99a |
|  | 6 | 8.03 | 6.62 | 0.54 | 1.00p | 0.993a | 7.07 | 5.53 | 0.51 | 1.00p | 0.95a | 5.11 | 3.48 | 0.47 | 1.00p | 0.99a |
|  | 12 | 10.54 | 10.39 | 0.82 | 1.00p | 0.993a | 9.35 | 7.93 | 0.59 | 1.00p | 0.955a | 7.37 | 5.74 | 0.54 | 1.00p | 0.99a |

TABLE II: Separation performance (SDR, SI-SDR, and STOI) and detection accuracy of CoSSD for different model sizes and SNR levels. Higher is better for all metrics. (A) Mixtures from the NIGENS dataset with one-hot conditional. (B) Mixtures from the NIGENS dataset with waveform conditional. (C) Mixtures from LibriSpeech and FSD50K with speech as the target. (D) Mixtures from LibriSpeech and FSD50K with non-speech as the target. (C) and (D) are with one-hot conditional only.

where *p* denotes the detection accuracy when the source of interest is present and is accurately predicted as being present, and *a* denotes detection accuracy when the source of interest is absent and is accurately predicted as being absent.

*1) NIGENS Dataset:* The NIGENS dataset exhibits the kinds of inter-class similarity and intra-class variation discussed earlier. For instance, classes like footsteps and door-knock sound similar, and the classes phone-ringtone and alarm have significant intra-class variation. Given these challenges, CoSSD performs well at separating specific sound classes in both conditional cases. Section (A) in Table 2 shows the performance of the one-hot conditional and section (B) shows the performance of the waveform conditional. Both achieve high separation performance and high detection accuracy. It is interesting that the performance difference between the two conditionals is not significant. As expected, the separation performance improves as the SNR increases, meaning the target audio source is louder than the other source making separation easier. Also as expected, the performance worsens as the model size decreases which is common in deep learning. The performance decrease from 1.3M to 600k parameters is greater than from 5.8M to 1.3M due to the limited capacity of the masker module in the smallest model.

*2) Special Case: Speech and Non-Speech Separation:* This experiment highlights the versatility of CoSSD where we apply it to speech and non-speech separation, with only a slight modification how the target source is coded in the one-hot conditional vector. Separating speech and non-speech is important in cinematography, podcasts, speech-denoising, and comfort-noise extraction applications. For example, given a mixture of bird songs and background (human) chatter, some applications might want to separate the bird songs while others might want to separate the chatter. CoSSD can do either by simply changing the conditional. Adapting to the binary class problem does not require any changes to the model architecture. Instead, the one-hot conditional vector [1,1,1,1,1,1,1,0,0,0,0,0,0,0] is used to denote speech and [0,0,0,0,0,0,0,1,1,1,1,1,1,1] to denote non-speech. We form a set of over 3.5M 2-sec mixtures combining speech from Librispeech and specific sounds from FSD50k. Similar to the NIGENS dataset, we present the results for different SNR values and different model sizes.

Section (C) of Table 2 shows the results for when speech is the target source and section (D) shows the results for when non-speech is the target. We see the model achieves nearly 100% accuracy at the detection task. This may be due to the abundance of training data and to the easier task of discriminating speech from non-speech. We see that the model is better able to separate speech than non-speech. This is probably due to the relative homogeneity of the speech clips from Librispeech compared to the non-speech ones from FSD50k.

*3) Comparison with baseline and other approaches:* To the best of our knowledge, there is no existing work that can separate multiple sources of interest in a mixture, using two types of conditionals, one-hot and waveform, and can be applied successfully to non-speech and speech separation and detection tasks, in a single end-to-end trained framework.

| SNR | (A) Cond Sep Only |  |  | (B) Cond Det Only |  | (C) CoSSD with Learned Enc-Dec |  |  |  |  | (D) CoSSD |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
|  | 5M |  |  | 800k |  | 5.91M |  |  |  |  | 5.8M |  |  |  |  |
|  | SDR | SI-SDR | STOI | Det Acc |  | SDR | SI-SDR | STOI | Det Acc |  | SDR | SI-SDR | STOI | Det Acc |  |
| 0 | 11.66 | 10.36 | 0.66 | 0.50p | 0.76a | 12.93 | 11.38 | 0.66 | 0.74p | 0.92a | 12.31 | 10.81 | 0.67 | 0.84p | 0.94a |
| 6 | 14.63 | 13.43 | 0.71 | 0.47p | 0.76a | 15.42 | 13.96 | 0.71 | 0.78p | 0.92a | 15.12 | 13.69 | 0.72 | 0.87p | 0.93a |
| 12 | 17.07 | 15.97 | 0.75 | 0.45p | 0.76a | 17.34 | 15.99 | 0.74 | 0.80p | 0.92a | 17.32 | 15.94 | 0.76 | 0.87p | 0.92a |

TABLE III: Comparing the proposed CoSSD with other methods and baselines on NIGENS Dataset. Results in (A) and (B) show performance of Separation and Detection Networks when trained and applied separately to the proposed problem. (C) and (D) illustrate the motivation behind using STFT based Encoder Decoders instead of trainable 1D Conv Encoder and Decoders. Note that the results in Table 3(D) are the same as those in Table 2(A) for the 5.8M parameter model.

Therefore, it is difficult to have a fair comparison with existing work. However, we compare CoSSD with two other techniques to solve the proposed problem. First, we consider performing conditional separation and detection separately. Table 3, (A) and (B) show the results of this approach, where we trained these two models separately and applied them in succession. Performing conditional separation and detection separately deteriorates the separation quality by a small extent and the detection accuracy by a very large margin. Second, (C) and (D) in Table 3 support our motivation behind using STFT filterbanks in the encoder and decoder of CoSSD instead of the learned 1D Convolution filters that are used in Conv-TasNet. We find that the detection performance is significantly better for CoSSD with STFT filters, and the separation quality is similar. Note that learned filters can result in extra parameters through the encoders and decoders that can prove significant in smaller networks.

## V. Discussion

Given that CoSSD performs conditional separation and detection jointly using a common representation, it begs whether there is always a positive correlation between the separation performance vs. detection performance. For most of the 35 training epochs, the detection and separation performance positively correlate. However, we start seeing a slight negative correlation only when the validation performance starts plateauing. Hence, choosing the correct checkpoint during training becomes key in finding a desirable separation vs. detection performance balance.

When considering other approaches for comparison, we found that existing frameworks cannot handle the case where there are potentially multiple sources of interest in the mixture, but we want to select which one to separate. Therefore, we compare the proposed model with a CoSSD with learned 1D-Conv encoders and decoders that can be viewed as a highly augmented blind source separation framework - Conv-TasNet. Table 3 (C) shows the results of this setting of CoSSD where it performs poorly on detection tasks compared to the learned STFT filterbank based CoSSD.

We compare the performance of different model sizes in the context of potentially deploying the framework in real-time or in resource-constrained environments. The main contribution of the paper is a solution to a new problem. We expect that the performance of our CoSSD framework would further improve given more parameters and compute time.

## VI. Conclusion

We proposed an end-to-end single-channel waveform-based conditional source separation and detection model, and performed extensive experiments comparing with other potential approaches using the NIGENS, LibriSpeech, and FSD50k datasets. CoSSD operates on a finite set of sound source classes while being robust to mixtures from an open-world. Future work includes converting the modules fully or partly [17] to transformer networks.

See baligar.github.io/CoSSD/ for audio samples.

## References

[1] 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, vol. 27, no. 8, pp. 1256–1266,2019.

[2] Beat Gfeller, Dominik Roblek, and Marco Tagliasac-chi, “One-shot conditional audio filtering of arbitrary sounds,”in ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2021, pp. 501–505.

[3] Olaf Ronneberger, Philipp Fischer, and Thomas Brox,“U-net: Convolutional networks for biomedical image segmentation,” in International Conference on Medical image computing and computer-assisted intervention. Springer, 2015, pp. 234–241.

[4] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville, “Film: Visual reasoning with a general conditioning layer,” in Proceedings of the AAAI Conference on Artificial Intelligence, 2018,vol. 32.

[5] Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi ZelnikManor, “Imagenet-21k pre-training for the masses,”arXiv preprint arXiv:2104.10972, 2021.

[6] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta, “Revisiting unreasonable effectiveness of data in deep learning era,” in Proceedings of the IEEE international conference on computer vision, 2017, pp.843–852.

[7] Qiuqiang Kong, Yuxuan Wang, Xuchen Song, Yin Cao,Wenwu Wang, and Mark D Plumbley, “Source separation with weakly labelled data: An approach to computational auditory scene analysis,” in ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020, pp. 101–105.

[8] Tsubasa Ochiai, Marc Delcroix, Yuma Koizumi, Hiroaki Ito, Keisuke Kinoshita, and Shoko Araki, “Lis-ten to what you want: Neural network-based universal sound selector,”arXiv preprint arXiv:2006.05712,2020.

[9] Nicolas Turpault, Scott Wisdom, Hakan Erdogan, JohnHershey, Romain Serizel, Eduardo Fonseca, PremSeetharaman, and Justin Salamon, “Improving sound event detection in domestic environments using soundseparation,”arXiv preprint arXiv:2007.03932, 2020.

[10] Efthymios Tzinis, Scott Wisdom, John R Hershey, ArenJansen, and Daniel PW Ellis,“Improving universal sound separation using sound classification,” in ICASSP2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE,2020, pp. 96–100.

[11] Eduardo Fonseca, Jordi Pons Puig, Xavier Favory,Frederic Font Corbera, Dmitry Bogdanov, Andres Ferraro, Sergio Oramas, Alastair Porter, and Xavier Serra,“Freesound datasets: a platform for the creation of openaudio datasets,” in Hu X, Cunningham SJ, Turnbull D,Duan Z, editors. Proceedings of the 18th ISMIR Conference; 2017 oct 23-27; Suzhou, China.[Canada]: International Society for Music Information Retrieval; 2017.p. 486-93. International Society for Music Information Retrieval (ISMIR), 2017.

[12] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan,and John R Hershey, “SDR–half-baked or well done?,”in ICASSP 2019-2019 IEEE International Conferenceon Acoustics, Speech and Signal Processing (ICASSP).IEEE, 2019, pp. 626–630.

[13] Ivo Trowitzsch, Jalil Taghia, Youssef Kashef, andKlaus Obermayer, “The NIGENS general sound eventsdatabase,”arXiv preprint arXiv:1902.08314, 2019.

[14] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, “Librispeech: an ASR corpus based on public domain audio books,” in 2015 IEEE internationalconference on acoustics, speech and signal processing(ICASSP). IEEE, 2015, pp. 5206–5210.

[15] Eduardo Fonseca, Xavier Favory, Jordi Pons, Frederic Font, and Xavier Serra,“FSD50k:an opendataset of human-labeled sound events,”arXiv preprintarXiv:2010.00475, 2020.

[16] Cees H Taal, Richard C Hendriks, Richard Heusdens,and Jesper Jensen, “A short-time objective intelligibility measure for time-frequency weighted noisy speech,”in 2010 IEEE international conference on acoustics,speech and signal processing. IEEE, 2010, pp. 4214–4217.

[17] Jingjing Chen, Qirong Mao, and Dong Liu,“Dual-path transformer network: Direct context-aware modeling for end-to-end monaural speech separation,”arXivpreprint arXiv:2007.13975, 2020.