Unsupervised Sound Separation Using Mixtures of Mixtures

Scott Wisdom
Google Research
scottwisdom@google.com

Efthymios Tzinis[*]
University of Illinois
at Urbana-Champaign

Hakan Erdogan
Google Research
hakanerdogan@google.com

Ron J. Weiss
Google Research
ronw@google.com

Kevin Wilson
Google Research
kwwilson@google.com

John R. Hershey
Google Research
johnhershey@google.com

Abstract

In recent years, rapid progress has been made on the problem of single-channel sound separation using supervised training of deep neural networks. In such supervised approaches, the model is trained to predict the component sources from synthetic mixtures created by adding up isolated ground-truth sources. The reliance on this synthetic training data is problematic because good performance depends upon the degree of match between the training data and real-world audio, especially in terms of the acoustic conditions and distribution of sources. The acoustic properties can be challenging to accurately simulate, and the distribution of sound types may be hard to replicate. In this paper, we propose a completely unsupervised method, mixture invariant training (MixIT), that requires only single-channel acoustic mixtures. In MixIT, training examples are constructed by mixing together existing mixtures, and the model separates them into a variable number of latent sources, such that the separated sources can be remixed to approximate the original mixtures. We show that MixIT can achieve competitive performance compared to supervised methods on speech separation. Using MixIT in a semi-supervised learning setting enables unsupervised domain adaptation and learning from large amounts of real-world data without ground-truth source waveforms. In particular, we significantly improve reverberant speech separation performance by incorporating reverberant mixtures, train a speech enhancement system from noisy mixtures, and improve universal sound separation by incorporating a large amount of in-the-wild data.

1 Introduction

Audio perception is fraught with a fundamental problem: individual sounds are convolved with unknown acoustic reverberation functions and mixed together at the acoustic sensor in a way that is impossible to disentangle without prior knowledge of the source characteristics. It is a hallmark of human hearing that we are able to hear the nuances of different sources, even when presented with a monaural mixture of sounds. In recent years significant progress has been made on extracting estimates of each source from single-channel recordings, using supervised deep learning methods. These techniques have been applied to important tasks such as speaker-independent enhancement (separation of speech from nonspeech interference) [17, 42] and speech separation (separation of

[*] Work done during an internship at Google.

Preprint. Under review.

speech from speech) [15, 18, 46]. The more general "universal sound separation" problem of separating arbitrary classes of sound from each other has also recently been addressed [19, 39].

These approaches have used supervised training, in which ground-truth source waveforms are considered targets for various loss functions including mask-based deep clustering [15] and permutation-invariant signal-level losses [18, 46]. Deep clustering is an embedding-based approach that implicitly represents the assignment of elements of a mixture, such as time-frequency bins of a spectrogram, to sources in a way that is independent of any ordering of the sources. In permutation-invariant training [18, 46], the model explicitly outputs the signals in an arbitrary order, and the loss function finds the permutation of that order that best matches the estimated signals to the references, i.e. treating the problem as a set prediction task. In both cases the ground-truth signals are inherently part of the loss.

A major problem with supervised training for source separation is that it is not feasible to record both the mixture signal and the individual ground-truth source signals in an acoustic environment, because source recordings are contaminated by cross-talk. Therefore supervised training has relied on synthetic mixtures created by adding up isolated ground-truth sources, with or without a simulation of the acoustic environment. Although supervised training has been effective in training models that perform well on data that match the same distribution of mixtures, they fare poorly when there is mismatch in the distribution of sound types [28], or in acoustic conditions such as reverberation [27]. It is difficult to match the characteristics of a real dataset because the distribution of source types and room characteristics may be unknown and difficult to estimate, data of every source type in isolation may not be readily available, and accurately simulating realistic acoustics is challenging.

One approach to avoiding these difficulties is to use acoustic mixtures from the target domain, without references, directly in training. To that end, weakly supervised training has been proposed to substitute the strong labels of source references with another modality such as class labels, visual features, or spatial information. In [32] class labels were used as a substitute for signal-level losses. The spatial locations of individual sources, which can be inferred from multichannel audio, has also been used to guide learning of single-channel separation [38, 35, 8]. Visual input corresponding to each source has been used to supervise the extraction of the corresponding sources in [12], where the targets included mixtures of sources, and the mapping between source estimates and mixture references was given by the video correspondence. Because these approaches rely on multimodal training data containing extra input in the form of labels, video, or multichannel signals, they cannot be used in settings where only single-channel audio is available.

In this paper, we propose a novel unsupervised training framework that requires only single-channel acoustic mixtures. This framework is related to permutation-invariant training (PIT) [46], in which the permutation used to match source estimates to source references is relaxed to allow summation over some of the sources. In our proposed *mixture invariant training* (MixIT), instead of single-source references, we use mixtures from the target domain as references, and the input to the separation model is formed by summing together these mixtures to form a mixture of mixtures. The model is trained to separate this input into a variable number of latent sources, such that the separated sources can be remixed to approximate the original mixtures.

Contributions: (1) we propose the first purely unsupervised learning method that is effective for audio-only single-channel separation tasks such as speech separation and find that it can achieve competitive performance compared to supervised methods; (2) we provide extensive experiments with cross-domain adaptation to show the effectiveness of MixIT for domain adaptation to different reverberation characteristics in semi-supervised settings; (3) the proposed method opens up the use of a wider variety of data, such as training speech enhancement models from noisy mixtures by only using speech activity labels, or improving performance universal sound separation models by training on large amounts of unlabeled, in-the-wild data.

2 Relation to previous work

Early separation approaches used hidden Markov models [33, 14, 21] and non-negative matrix factorization [36, 34] trained on isolated single-source data. These generative models incorporated the signal combination model into the likelihood function. The explicit generative construction enabled maximum likelihood inference and unsupervised adaptation [41]. However, the difficulty of discriminative training, restrictive modeling assumptions, and the need for approximation methods for tractable inference were liabilities. MixIT avoids these issues by performing self-supervised discriminative training of unrestricted deep networks, while still enabling unsupervised adaptation.

Discriminative source separation models generate synthetic mixtures form isolated sources which are also used as targets for training. Considering synthesis as part of the learning framework, such approaches can be described as self-supervised in that they start with single-source data. Early methods posed the problem in terms of time-frequency mask estimation, and considered restrictive cases such as speaker-dependent models, and class-specific separation, e.g. speech versus music [17], or noise [42]. However, more general speaker-independent speech separation [15, 46], and class-independent universal sound separation [19] are now addressed using class-independent methods such as deep clustering [15] and PIT [46]. These frameworks handle the output permutation problem caused by the lack of a unique source class for each output. Recent state-of-the-art models have shifted from mask-based recurrent networks to time-domain convolutional networks for speech separation [26], speech enhancement [19], and universal sound separation [19, 39] tasks.

MixIT follows this trend and uses a signal-level discriminative loss. The framework can be used with any architecture; in this paper we use a modern time-convolutional network. Unlike previous supervised approaches, MixIT can use a database of only mixtures as references, enabling training directly on target-domain mixtures for which ground-truth source signals cannot be obtained.

Similar to MixIT, [12] uses *mixtures of mixtures* (MoMs) as input, and sums over estimated sources to match the target mixtures, using the *co-separation loss*. However, the co-separation loss does not identify correspondence between sources and mixtures, since that is established by the supervising video inputs, each of which is assumed to correspond to one source. In MixIT this is handled in an unsupervised manner, by finding the best correspondence between sums of sources and the reference mixtures without using other modalities, making the proposed methods the first fully unsupervised separation work using MoMs.

Also related is *adversarial unmix-and-remix* [16], which separates linear image mixtures in a GAN framework, with the discriminator operating on mixtures rather than single sources. Mixtures are separated, and the resulting sources are recombined to form new mixtures. Adversarial training encourages new mixtures to match the distribution of the original inputs. A cycle consistency loss is also used by separating and remixing the new mixtures. In contrast, MixIT avoids the difficulty of saddle-point optimization associated with GANs. An advantage of [16] is that it is trained with only the original mixtures as input, while MixIT uses MoMs, relying on generalization to work on single mixtures. Unmix-and-remix was reported to work well on image mixtures, but failed on audio mixtures [16]. We show that MixIT works well on several audio tasks. However, these unmix-and-remix is complementary, and could be combined with MixIT in future work.

Mixing inputs and outputs as in MixIT is reminiscent of MixUp regularization [48], which has been a useful component of recent techniques for semi-supervised classification [4, 3]. Our approach differs from these in that the sound separation problem is regression rather than classification, and thus training targets are not discrete labels, but waveforms in the same domain as the model inputs. Moreover, our approach is unsupervised, whereas MixUp is a regularization for a supervised task.

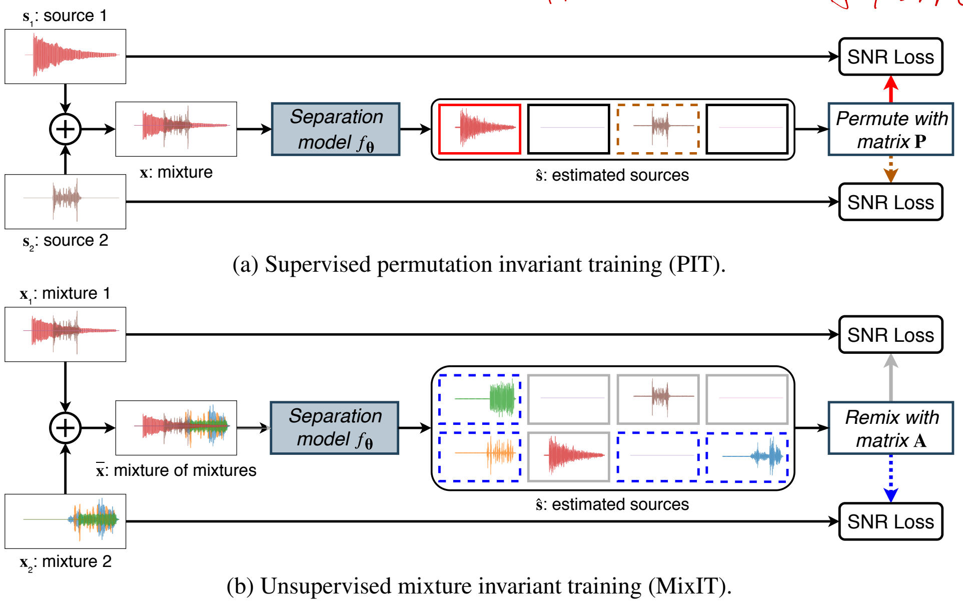
In our experiments we explore adapting separation models to domains for which it is difficult to obtain reference source signals. Recent approaches to such unsupervised domain adaptation have used adversarial training to learn domain-invariant intermediate network activations [11, 6, 37], learn to translate synthetic inputs to the target domain [5], or train student and teacher models to predict consistent separated estimates from supervised and unsupervised mixtures [22]. In contrast, we take a semi-supervised learning approach and jointly train the same network using both supervised and unsupervised losses, without making explicit use of domain labels. A related approach was proposed for speech enhancement in [1], inspired by [24], which uses a self-supervised loss that requires a second uncorrelated realization of the same noisy input signal, obtained from a specialized mid-side microphone. In contrast, the proposed unsupervised loss only requires single-channel mixture recordings with minimal assumptions.

3 Method

We generalize the permutation-invariant training framework to operate directly on unsupervised mixtures, as illustrated in Figure 1. Formally, a supervised separation dataset is comprised of pairs of

ปกติจะทำตามรูป A → ~~[illegible]~~ เราต้องมี clean

→ สิ่งที่ต่อไป → ไม่ต้อง source → แต่แยกจาก mixture และ



(a) Supervised permutation invariant training (PIT).

(b) Unsupervised mixture invariant training (MixIT).

Figure 1: Schematic comparing (a) PIT separating a two source mixture into up to four constituent sources to (b) MixIT separating a mixture of mixtures into up to eight constituent sources. Arrow color indicates the best match between estimated sources and the ground-truth.

input mixtures $\mathbf{x} = \sum_{n=1}^{N} \mathbf{s}_n$ and their constituent sources $\mathbf{s}_n \in \mathbb{R}^T$, where each mixture contains up to N sources with T time samples each. Without loss of generality, for the mixtures that contain only $N' < N$ sources we assume that $\mathbf{s}_n = \mathbf{0}$ for $N' < n \leq N$. An unsupervised dataset only contains input mixtures without underlying reference sources. However we assume that the maximum number of sources which may be present in the mixtures is known.

3.1 Permutation invariant training (PIT)

In the supervised case we are given a mixture $\mathbf{x}$ and its corresponding sources $\mathbf{s}$ to train on. The input mixture $\mathbf{x}$ is fed through a separation model $f_{\boldsymbol{\theta}}$ with parameters $\boldsymbol{\theta}$. The model predicts M sources: $\hat{\mathbf{s}} = f_{\boldsymbol{\theta}}(\mathbf{x})$, where $M = N$ is the maximum number of sources co-existing in any given mixture drawn from the supervised dataset. Consequently, the supervised separation loss can be written as:

$$\mathcal{L}_{\text{PIT}}\left(\mathbf{s}, \hat{\mathbf{s}}\right) = \min_{\mathbf{P}} \sum_{m=1}^{M} \mathcal{L}\left(\mathbf{s}_m, [\mathbf{P}\hat{\mathbf{s}}]_m\right), \tag{1}$$

where $\mathbf{P}$ is an $M \times M$ permutation matrix and $\mathcal{L}$ is a signal-level loss function such as negative signal-to-noise ratio (SNR). There is no predefined ordering of the source signals. Instead, the loss is computed using the permutation which gives the best match between ground-truth reference sources $\mathbf{s}$ and estimated sources $\hat{\mathbf{s}}$.

The signal-level loss function between a reference $\mathbf{y} \in \mathbb{R}^T$ and estimate $\hat{\mathbf{y}} \in \mathbb{R}^T$ from a model with trainable parameters $\boldsymbol{\theta}$ is the negative thresholded SNR:

$$\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) = -10 \log_{10} \frac{\|\mathbf{y}\|^2}{\|\mathbf{y} - \hat{\mathbf{y}}\|^2 + \tau\|\mathbf{y}\|^2} = 10 \log_{10}\left(\|\mathbf{y} - \hat{\mathbf{y}}\|^2 + \tau\|\mathbf{y}\|^2\right) \underbrace{-10 \log_{10} \|\mathbf{y}\|^2}_{\text{const. w.r.t. } \boldsymbol{\theta}}, \tag{2}$$

where $\tau = 10^{-\text{SNR}_{\text{max}}/10}$ acts as a soft threshold that clamps the loss at SNR_{max}. This threshold prevents examples that are already well-separated from dominating the gradients within a training batch. We found $\text{SNR}_{\text{max}} = 30$ dB to be a good value, as shown in the Appendix.

3.2 Mixture invariant training (MixIT)

The main limitation of PIT is that it requires knowledge of the ground truth source signals $\mathbf{s}$, and therefore cannot directly leverage unsupervised data where only mixtures $\mathbf{x}$ are observed. MixIT overcomes this problem as follows. Consider two mixtures $\mathbf{x}_1$ and $\mathbf{x}_2$, each comprised of up to N underlying sources (any number of mixtures can be used, but here we use two for simplicity). The mixtures are drawn at random without replacement from an unsupervised dataset and a mixture of mixtures is formed by adding them together: $\bar{\mathbf{x}} = \mathbf{x}_1 + \mathbf{x}_2$. The separation model $f_{\boldsymbol{\theta}}$ takes $\bar{\mathbf{x}}$ as input, and predicts $M = 2N$ source signals. In this way we make sure that the model is always capable of predicting enough sources for any $\bar{\mathbf{x}}$. The unsupervised MixIT loss is computed between the estimated sources $\hat{\mathbf{s}}$ and the input mixtures $\mathbf{x}_1$, $\mathbf{x}_2$ as follows:

ความต่าง => คล้าย p => แต่จริง
คือการเลือกเสียง

$$\mathcal{L}_{\mathrm{MixIT}}\left(\mathbf{x}_1, \mathbf{x}_2, \hat{\mathbf{s}}\right) = \min_{\mathbf{A}} \sum_{i=1}^{2} \mathcal{L}\left(\mathbf{x}_i, [\mathbf{A}\hat{\mathbf{s}}]_i\right), \tag{3}$$

where $\mathcal{L}$ is the same signal-level loss used in PIT (2) and the *mixing matrix* $\mathbf{A} \in \mathbb{B}^{2\times M}$ is constrained to the set of $2 \times M$ binary matrices where each column sums to 1, i.e. the set of matrices which assign each source $\hat{\mathbf{s}}$ to either $\mathbf{x}_1$ or $\mathbf{x}_2$. MixIT minimizes the total loss between mixtures $\mathbf{x}$ and remixed separated sources $\hat{\mathbf{x}} = \mathbf{A}\hat{\mathbf{s}}$ by choosing the best match between sources and mixtures, in a process analogous to PIT. Due to the constraints on $\mathbf{A}$, each separated source $\hat{\mathbf{s}}_m$ can only be used once. In practice, we optimize over $\mathbf{A}$ using an exhaustive $\mathcal{O}(2^M)$ search. The tasks considered in this paper only require M up to 8, which we empirically found to not take a significant amount of time during training. To scale to larger values of M, it should be possible to perform this optimization more efficiently, but we defer this to future work.

There is an implicit assumption in MixIT that the sources are additive, and that they are independent of each other in the original mixtures $\mathbf{x}_1$ and $\mathbf{x}_2$, in the sense that there is no information in $\overline{\mathbf{x}}$ about which sources belong to which mixtures. The two mixtures $\mathbf{x} \in \mathbb{R}^{2\times T}$ are assumed to result from mixing unknown sources $\mathbf{s}^* \in \mathbb{R}^{P\times T}$ using an unknown $2 \times P$ mixing matrix $\mathbf{A}^*$: $\mathbf{x} = \mathbf{A}^*\mathbf{s}^*$. If the network could infer which sources belong together in the references, and hence knew the mixing matrix $\mathbf{A}^*$ (up to a left permutation), then the M source estimates, $\hat{\mathbf{s}} \in \mathbb{R}^{M\times T}$ could minimize the loss (3) without separating all the sources (i.e., by under-separating). That is, for a known mixing matrix $\mathbf{A}^*$, the loss (3) could be minimized, for example, by the estimate $\hat{\mathbf{s}} = \mathbf{C}^+\mathbf{A}^*\mathbf{s}^*$, with $\mathbf{C}^+$ the pseudoinverse of a $2 \times M$ mixing matrix $\mathbf{C}$ such that $\mathbf{C}\mathbf{C}^+ = \mathbf{I}$, at $\mathbf{A} = \mathbf{C}$, since $\mathbf{C}\hat{\mathbf{s}} = \mathbf{C}\mathbf{C}^+\mathbf{A}^*\mathbf{s}^* = \mathbf{x}$. However, if the sources are independent, then the network cannot infer the mixing matrix that produced the reference mixtures. Nevertheless, the loss can be minimized with a single set of estimates, regardless of the mixing matrix $\mathbf{A}^*$, by separating all of the sources. That is, the estimated sources must be within a mixing matrix $\mathbf{B} \in \mathbb{B}^{P\times M}$ of the original sources, $\mathbf{s}^* = \mathbf{B}\hat{\mathbf{s}}$, so that (3) is minimized at $\mathbf{A} = \mathbf{A}^*\mathbf{B}$, for any $\mathbf{A}^*$. Hence, lack of knowledge about which sources belong to which mixtures encourages the network to separate the sources as much as possible.

$\hat{s} = C^{+}A^{*}$

Note that when $M > P$, the network can produce more estimates than there are sources (i.e., over-separate). There is no penalty in (3) for over-separating the sources; in this work semi-supervised training may help with this, and future work will address methods to discourage over-separation in the fully unsupervised case.

3.3 Semi-supervised training

When trained on M isolated sources, i.e. with full supervision, the MixIT loss is equivalent to PIT. Specifically, input mixtures $\mathbf{x}_i$ are replaced with ground-truth reference sources $\mathbf{s}_i$ and the mixing matrix $\mathbf{A}$ becomes an $M \times M$ permutation matrix $\mathbf{P}$. This makes it straightforward to combine both losses to perform semi-supervised learning. In essence, each training batch contains $p\%$ supervised data, for which we use the PIT loss (1), and the remaining contains unsupervised mixtures, for which we do not know their constituent sources and use the MixIT loss (3).

4 Experiments

Our separation model $f_{\boldsymbol{\theta}}$ consists of a learnable convolutional basis transform that produces mixture basis coefficients. These are processed by an improved time-domain convolutional network (TDCN++) [19], similar to ConvTasNet [26]. This network predicts M masks, each of which contains values between 0 and 1 and are the same size as the basis coefficients. These masks are multiplied elementwise with the basis coefficients, and M separated waveforms are produced by overlapping and adding the masked coefficients. A mixture consistency projection layer [45] is applied to constrain separated sources to add up to the input mixture. The architecture is described in more detail in the Appendix. All models are trained on 4 Google Cloud TPUs (16 chips) with the Adam optimizer [20], a batch size of 256, and learning rate of 10^{-3}.

Separation performance is measured using scale-invariant signal-to-noise ratio (SI-SNR) [23]. SI-SNR measures fidelity between a signal $\mathbf{y}$ and its estimate $\hat{\mathbf{y}}$ within an arbitrary scale factor:

$$\text{SI-SNR}(\mathbf{y}, \hat{\mathbf{y}}) = 10 \log_{10} \frac{\|\alpha \mathbf{y}\|^2}{\|\alpha \mathbf{y} - \hat{\mathbf{y}}\|^2}, \tag{4}$$

where $\alpha = \text{argmin}_a \|a\mathbf{y} - \hat{\mathbf{y}}\|^2 = \mathbf{y}^T \hat{\mathbf{y}} / \|\mathbf{y}\|^2$. Generally we measure SI-SNR improvement (SI-SNRi), which is the difference between the SI-SNR of each source estimate after processing, and the SI-SNR obtained using the input mixture as the estimate for each source. For mixtures that contain only a single source, SI-SNRi is not meaningful, because the mixture SI-SNR is infinite. In this case we measure performance using single-source absolute SI-SNR (SS). In real-world separation tasks, mixtures can contain a variable number of sources, or fewer sources than are produced by the separation model. To handle these cases during evaluation, we compute a multi-source SI-SNRi (MSi) metric by zero-padding the references to M sources, aligning them to the separated sources with a permutation that maximizes SI-SNR, and averaging the resulting SI-SNRi over non-zero references. Audio demos for all tasks are provided online[2].

4.1 Speech separation

For speech separation experiments, we use the WSJ0-2mix [15] and Libri2Mix [7] datasets, sampled at 8 kHz or 16 kHz. We also employ the reverberant spatialized version of WSJ0-2mix [40] and a reverberant version of Libri2Mix we created. Both datasets consist of utterances from male and female speakers drawn from either the Wall Street Journal (WSJ0) corpus or from LibriSpeech [30]. Reverberant versions are created by convolving utterances with room impulse responses generated by a room simulator employing the image method [2]. WSJ0-2mix provides 30 hours of training mixtures with individual source utterances drawn with replacement, and the train-360-clean split of Libri2Mix provides 364 hours of mixtures where source utterances are drawn without replacement.

For our experiment, we sweep the amount of supervised versus unsupervised data for both the anechoic and reverberant versions of WSJ0-2mix. The proportion p of unsupervised data from the same domain is swept from 0% to 100% where supervised training uses PIT with the separation loss (2) between ground-truth references and separated sources, and unsupervised training only uses the mixtures using MixIT (3) with the same separation loss (2) between mixtures and remixed separated sources. In both cases, the input to the separation model is a mixture of two mixtures. For training, 3 second clips are used for WSJ0-2mix, and 10 second clips for Libri2Mix.

We try two variants of this task: mixtures that always contain two speakers (2-source) such that MoMs always contain four sources, and mixtures containing either one or two speakers (1-or-2-source) such that MoMs contain two to four sources. Note that the network always has four outputs. Evaluation always uses single mixtures of two sources. To determine if unsupervised data can help with domain mismatch, we also consider using supervised data from a mismatched domain, which simulates the realistic scenario faced by practitioners training sound separation systems, where real acoustic mixtures from a target domain are available without references and synthetic supervised data must be created to match the distribution of the real data. It is difficult to perfectly match the distribution of real data, so synthetic supervised data will inevitably have some mismatch to the target domain.

The results on anechoic and reverberant WSJ0-2mix and Libri2Mix are shown in Figure 2. First, notice that reverberant data is more challenging to separate because reverberation smears out the spectral energy of sources over time, and thus all models achieve lower performance on reverberant data. Two-source mixture trained models tends to do less well compared to the 1-or-2-source variants. One difference with the 1-or-2-source setup is that the model observes some inputs that have two sources, which matches the evaluation. Another difference is that as references, the 1-source mixtures act as supervised examples.

Notice that for both anechoic and reverberant data, even completely unsupervised training with MixIT (rightmost points) achieves performance on par with supervised training (leftmost points) with 1-or-2-source mixtures. For 2-source mixtures, totally unsupervised performance is generally worse by up to

[2] https://universal-sound-separation.github.io/unsupervised_sound_separation

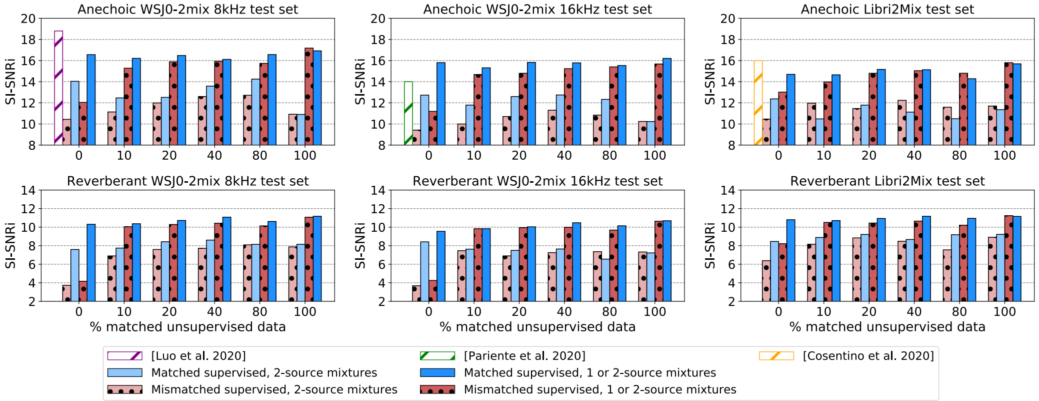


Figure 2: Sweeping proportion of matched unsupervised training examples with matched or mismatched supervised examples on WSJ0-2mix 8kHz (left), WSJ0-2mix 16kHz (middle), and Libri2Mix (right). The leftmost bars in each plot correspond to 100% supervision using PIT, and the rightmost bars are fully unsupervised using MixIT.

3 dB compared to fully or semi-supervised on anechoic data, while performance is more comparable on reverberant data. However, even a small amount of supervision (80% unsupervised) dramatically improves separation performance on anechoic data. When the supervised data is mismatched, adding only a small amount of unsupervised data (10%) from a matched domain drastically improves performance: using mismatched anechoic supervised data and matched reverberant unsupervised data, we observe boosts of 2-3 dB for 2-source mixtures on all datasets. For 1-to-2-source mixtures, performance increases by about 6 dB on WSJ0-2mix datasets and 2.5 dB for Libri2Mix.

Though our goal is to explore MixIT for less supervised learning, our models are competitive on anechoic datasets with state-of-the-art approaches that do not exploit additional information such as speaker identity. Figure 2 includes the best reported numbers for 8 kHz WSJ0-2mix [25], 16 kHz WSJ0-2mix [31], and Libri2Mix [7]. For WSJ0-2mix 8 kHz, other recent supervised approaches further improve over [25] by incorporating additional speaker identity information, including [29] with 20.1 dB and WaveSplit [47] with 20.4 dB. Note that MixIT is compatible with any of these network architectures, and could also be used in combination with auxiliary information.

4.2 Speech enhancement

MixIT can also be useful for tasks where sources are drawn from different classes. An example of such a task is speech enhancement, where the goal is to remove all nonspeech sounds from a mixture. We prepared a speech enhancement dataset using speech from LibriSpeech [30] and non-speech sounds from `freesound.org`. Based on user tags, we filtered out synthetic sounds (e.g. synthesizers). We also used a sound classification model [13] to avoid clips likely containing speech.

Using the clean speech data and filtered nonspeech data, we construct two splits of data: speech-plus-noise audio and noise-only audio. These two types of audio simulate annotating the speech activity of a large amount of data, which can easily be done automatically with commonly-available speech detection models. To ensure that the speech signal is always output as the first separated source, we added constraints to the possible mixings in MixIT. The first mixture $\mathbf{x}_1$ is always drawn from the speech-plus-noise split, and the second mixture $\mathbf{x}_2$ is always drawn from the noise-only split. A three-output separation model is trained, where the optimization over the mixing matrix $\mathbf{A}$ (3) is constrained such that only outputs 1 and 3 or 1 and 2 can be used to reconstruct the speech-plus-noise mixture $\mathbf{x}_1$, and only separated sources 2 or 3 can be used to match the noise-only mixture $\mathbf{x}_2$.

As a baseline, we also trained a supervised two-output separation model with our signal-level loss (2) on both separated speech and noise outputs. On a held out test set, the supervised model achieves 15.0 dB SI-SNRi for speech, and the unsupervised MixIT model achieves 11.4 dB SI-SNRi. Thus, by only using labels about speech presence, which is easy to automatically annotate with a speech activity detector, we can train a speech enhancement model only from mixtures with MixIT that achieves 76% of the performance of a fully-supervised model. Such a fully-supervised model is potentially

Table 1: Multi-source SI-SNR improvement (MSi) and single-source SI-SNR (SS) in dB on FUSS validation and test sets, with or without additional reverberation, for different combinations of supervised and unsupervised training data and probability of zeroing out one supervised mixture p_0.

			No reverb				Reverb			
			Validation		Test		Validation		Test	
Supervised	p_0	Unsupervised	MSi	SS	MSi	SS	MSi	SS	MSi	SS
FUSS (16 hr)	0.2	Freesound (120 hr)	**13.9**	32.1	13.9	29.0	13.6	33.4	13.3	30.7
FUSS (16 hr)	0.0	Freesound (120 hr)	**13.9**	14.8	**14.5**	12.0	**13.9**	16.9	**13.9**	13.1
FUSS (16 hr)	0.2	–	13.8	**35.5**	12.8	**35.9**	13.7	35.6	12.2	**36.3**
FUSS (16 hr)	0.0	–	12.6	11.7	11.8	15.9	12.4	11.9	11.4	16.4
FUSS (8 hr)	0.2	FUSS (8 hr)	13.5	30.8	12.4	33.5	13.2	30.9	11.6	34.0
FUSS (8 hr)	0.0	FUSS (8 hr)	12.8	11.4	12.1	14.4	12.7	11.4	11.4	14.8
–	–	FUSS (16 hr)	12.4	3.7	12.4	4.3	11.9	3.6	11.4	3.6
–	–	Freesound (120 hr)	10.6	8.4	11.0	8.3	10.8	8.3	10.7	8.1

vulnerable to domain mismatch, and also requires substantial effort to construct a large synthetic training set. In contrast, MixIT makes it possible to easily leverage vast amounts of unsupervised noisy data that also matches the real-world distribution, which we intend to explore in future work.

4.3 Universal sound separation

Universal sound separation is the task of separating arbitrary sounds from an acoustic mixture [19, 39]. For our experiments, we use the recently released Free Universal Sound Separation (FUSS) dataset [43, 44],which consists of sounds drawn from `freesound.org`. Using labels from a prerelease of FSD50k [9], gathered through the Freesound Annotator [10], source clips have been screened such that they likely only contain a single sound class. The 10 second mixtures contain one to four sources, and unprocessed and reverberant versions of the dataset are available, where the reverberant version uses the image method to simulate room impulse responses with frequency-dependent walls.

Table 1 shows performance on FUSS datasets in a variety of supervised, semi-supervised and purely unsupervised settings. The Freesound dataset is the same as used for our speech enhancement experiment in Section 4.2, and consists of about 120 hours of mixture data. In contrast, the FUSS training datasets consist of about 55 hours of mixtures constructed from about 16 hours of isolated sources. We also experiment with randomly zeroing out one of the supervised mixtures with probability p_0 during training. In particular, we found that using $p_0 = 0.2$ helps all semi-supervised and fully supervised models greatly improve SS, though sometimes with a slight loss of MSi.

Notably, MixIT is able to leverage the variety of Freesound mixtures and yields an MSi of 13.9 dB on the test set, an improvement of 1.1 dB compared to the purely supervised setting for similar SS (35.9 dB for purely supervised versus 29.0 dB for semi-supervised). Moreover, it is also clear that providing the model with raw audio from Freesound leads to better generalization on unseen mixtures with reverb. For reverberant FUSS, this semi-supervised approach also outperforms the purely-supervised approach by a significant margin in terms of MSi: 12.2 dB $\rightarrow$ 13.3 dB for comparable SS (36.3 dB versus 30.7 dB). Furthermore, adding Freesound mixtures improves generalization and boosts the performance on the test set even more compared to the validation set.

In purely unsupervised settings (last two rows), we see that MixIT is able to perform adequately using only mixtures from the same non-reverberant FUSS dataset (12.4 dB MSi test without reverb, 11.4 dB MSi test with reverb) but interestingly even when using mixtures from unmatched Freesound data (11.0 dB MSi test without reverb, 10.7 dB MSi test with reverb). SS scores are lower for these purely unsupervised approaches since the models are never provided with supervised single-source inputs.

5 Discussion

The experiments show that MixIT works well for speech separation, speech enhancement, and universal sound separation. In the speech separation experiments, Unsupervised domain adaptation always helps: matched fully unsupervised training is always better than mismatched fully supervised training, often by a significant margin. To the best of our knowledge, this is the first single-channel purely unsupervised method which obtains comparable performance to state-of-the-art fully-supervised approaches on sound separation tasks. For universal sound separation and speech enhancement, the unsupervised training does not help as much, presumably because the synthetic test sets are well-matched to the supervised training domain. However, for universal sound separation, unsupervised training does seem to help slightly with generalization to the test set, relative to the supervised-only training, which tends to do better on the validation set. In the fully unsupervised case, performance on speech enhancement and FUSS is not at supervised levels as it is in the speech separation experiments, but the performance it achieves with no supervision remains unprecedented. Unsupervised performance is at its worst in the single-source mixture case of the FUSS task. This may be because MixIT does not discourage further separation of single sources. A reasonable approach to this may be to impose an additional "separation consistency" loss to ensure consistency between sources separated from the mixture of mixtures and those separated from the individual mixtures (including single-source mixtures), which we leave for future work.

In some of the experiments reported here, the data preparation has some limitations. The WSJ0-2mix and FUSS data have the property that each unique source may be repeated across multiple mixture examples, whereas Libri2Mix and presumably Freesound, both use unique sources in every mixture. Such re-use of source signals is not a problem for ordinary supervised separation, but in the context of MixIT, there is a possibility that the model may abuse this redundancy. In particular in the 1-or-2 source case, this raises the chance that each source appears as a reference, which could make the unsupervised training act more like supervised training. However, the unsupervised performance on Libri2Mix, which does not contain redundant sources, parallels the WSJ0-2mix results and shows that if there is a redundancy loophole to be exploited in some cases, it is not needed for good performance.

Another caveat is that the MixIT method is best suited to cases where sources occur together independently, so that the network cannot determine which sources in the mixture of mixtures came from the same reference. The fact that unsupervised training works less well on the Freesound data, when evaluated on the artificially mixed FUSS data, may be partly because of the non-uniform co-occurrance statistics in Freesound, for two reasons. First, the evaluation is done on FUSS, which has uniform co-occurrence statistics, so there is a potential mismatch. Second, in cases where there are strong co-occurrence dependencies, the network could avoid separating the sources within each component mixture, treating them as a single source. It is a future area of research to understand when this occurs and what the remedies may be. A first step may be to impose a non-uniform co-occurrence distribution in synthetic data to understand how the method behaves as the strength of the co-occurrence is varied. An ultimate goal is to evaluate separation on real mixture data; however, this remains challenging because of the lack of ground truth. As a proxy, future experiments may use recognition or human listening as a measure of separation, depending on the application.

6 Conclusion

We have presented MixIT, a new paradigm for training sound separation models in a completely unsupervised manner where ground-truth source references are not required. Across several tasks including speech separation, speech enhancement, and universal sound separation, we demonstrated that MixIT can approach the performance of supervised PIT, and is especially helpful in a semi-supervised setup to adapt to mismatched domains. More broadly, MixIT opens new lines of research where massive amounts of previously untapped in-the-wild data can be leveraged to train sound separation systems.

Acknowledgments

The idea of using mixtures of mixtures came about during the 2015 JHU CLSP Jelinek Summer Workshop at University of Washington. Special thanks goes to Jonathan Le Roux for prior discussions of other methods using MoMs, and to Aren Jansen for helpful comments on the manuscript.

References

[1] N. Alamdari, A. Azarang, and N. Kehtarnavaz. Self-supervised deep learning-based speech denoising. *arXiv preprint arXiv:1904.12069*, 2019.

[2] J. B. Allen and D. A. Berkley. Image method for efficiently simulating small-room acoustics. *The Journal of the Acoustical Society of America*, 65(4):943–950, 1979.

[3] D. Berthelot, N. Carlini, E. D. Cubuk, A. Kurakin, K. Sohn, H. Zhang, and C. Raffel. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *arXiv preprint arXiv:1911.09785*, 2019.

[4] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel. MixMatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, pages 5050–5060, 2019.

[5] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 3722–3731, 2017.

[6] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan. Domain separation networks. In *Advances in Neural Information Processing Systems*, pages 343–351, 2016.

[7] J. Cosentino, M. Pariente, S. Cornell, A. Deleforge, and E. Vincent. LibriMix: An open-source dataset for generalizable speech separation. *arXiv preprint arXiv:2005.11262*, 2020.

[8] L. Drude, D. Hasenklever, and R. Haeb-Umbach. Unsupervised training of a deep clustering model for multichannel blind source separation. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 695–699, 2019.

[9] E. Fonseca, X. Favory, J. Pons, F. Font, and X. Serra. FSD50k: an open dataset of human-labeled sound events. In *arXiv*, 2020.

[10] E. Fonseca, J. Pons Puig, X. Favory, F. Font Corbera, D. Bogdanov, A. Ferraro, S. Oramas, A. Porter, and X. Serra. Freesound datasets: a platform for the creation of open audio datasets. In *Proc. International Society for Music Information Retrieval ISMIR*, pages 486–493, 2017.

[11] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[12] R. Gao and K. Grauman. Co-separating sounds of visual objects. In *Proc. IEEE International Conference on Computer Vision*, pages 3879–3888, 2019.

[13] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 776–780, 2017.

[14] J. R. Hershey and M. Casey. Audio-visual sound separation via hidden markov models. In *Advances in Neural Information Processing Systems*, pages 1173–1180, 2002.

[15] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe. Deep clustering: Discriminative embeddings for segmentation and separation. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 31–35, 2016.

[16] Y. Hoshen. Towards unsupervised single-channel blind source separation using adversarial pair unmix-and-remix. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 3272–3276, 2019.

[17] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis. Deep learning for monaural speech separation. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 1562–1566, 2014.

[18] Y. Isik, J. L. Roux, Z. Chen, S. Watanabe, and J. R. Hershey. Single-channel multi-speaker separation using deep clustering. *arXiv preprint arXiv:1607.02173*, 2016.

[19] I. Kavalerov, S. Wisdom, H. Erdogan, B. Patton, K. Wilson, J. Le Roux, and J. R. Hershey. Universal sound separation. In *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2019.

[20] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proc. International Conference on Learning Representations (ICLR)*, 2015.

[21] T. Kristjansson, J. Hershey, P. Olsen, S. Rennie, and R. Gopinath. Super-human multi-talker speech recognition: The IBM 2006 speech separation challenge system. In *Proc. Ninth International Conference on Spoken Language Processing*, 2006.

[22] M. W. Lam, J. Wang, D. Su, and D. Yu. Mixup-breakdown: a consistency training method for improving generalization of speech separation models. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 6374–6378. IEEE, 2020.

[23] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey. SDR–half-baked or well done? In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 626–630, 2019.

[24] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila. Noise2noise: Learning image restoration without clean data. In *Proc. International Conference on Machine Learning (ICML)*, pages 2965–2974, 2018.

[25] Y. Luo, Z. Chen, and T. Yoshioka. Dual-path RNN: efficient long sequence modeling for time-domain single-channel speech separation. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 46–50, 2020.

[26] Y. Luo and N. 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.

[27] M. Maciejewski, G. Sell, L. P. Garcia-Perera, S. Watanabe, and S. Khudanpur. Building corpora for single-channel speech separation across multiple domains. *arXiv preprint arXiv:1811.02641*, 2018.

[28] E. Manilow, G. Wichern, P. Seetharaman, and J. Le Roux. Cutting music source separation some slakh: A dataset to study the impact of training data quality and quantity. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 45–49, 2019.

[29] E. Nachmani, Y. Adi, and L. Wolf. Voice separation with an unknown number of multiple speakers. *arXiv preprint arXiv:2003.01531*, 2020.

[30] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur. LibriSpeech: an ASR corpus based on public domain audio books. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 5206–5210, Apr. 2015.

[31] M. Pariente, S. Cornell, A. Deleforge, and E. Vincent. Filterbank design for end-to-end speech separation. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 6364–6368, 2020.

[32] F. Pishdadian, G. Wichern, and J. Le Roux. Finding strength in weakness: Learning to separate sounds with weak supervision. *arXiv preprint arXiv:1911.02182*, 2019.

[33] S. T. Roweis. One microphone source separation. In *Advances in Neural Information Processing Systems*, pages 793–799, 2001.

[34] M. N. Schmidt and R. K. Olsson. Single-channel speech separation using sparse non-negative matrix factorization. In *Proc. Ninth International Conference on Spoken Language Processing*, 2006.

[35] P. Seetharaman, G. Wichern, J. Le Roux, and B. Pardo. Bootstrapping single-channel source separation via unsupervised spatial clustering on stereo mixtures. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 356–360, 2019.

[36] P. Smaragdis. Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs. In *Proc. International Conference on Independent Component Analysis and Signal Separation*, pages 494–499, 2004.

[37] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176, 2017.

[38] E. Tzinis, S. Venkataramani, and P. Smaragdis. Unsupervised deep clustering for source separation: Direct learning from mixtures using spatial information. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 81–85, 2019.

[39] E. Tzinis, S. Wisdom, J. R. Hershey, A. Jansen, and D. P. W. Ellis. Improving universal sound separation using sound classification. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 96–100, 2020.

[40] Z.-Q. Wang, J. Le Roux, and J. R. Hershey. Multi-channel deep clustering: Discriminative spectral and spatial embeddings for speaker-independent speech separation. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2018.

[41] R. J. Weiss and D. P. W. Ellis. Speech separation using speaker-adapted eigenvoice speech models. *Computer Speech & Language*, 24(1):16–29, 2010.

[42] F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. Le Roux, J. R. Hershey, and B. Schuller. Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 91–99, 2015.

[43] S. Wisdom, H. Erdogan, D. P. W. Ellis, and J. R. Hershey. *Free Universal Sound Separation (FUSS) dataset*, 2020. `https://github.com/google-research/sound-separation/tree/master/datasets/fuss`.

[44] S. Wisdom, H. Erdogan, D. P. W. Ellis, R. Serizel, N. Turpault, E. Fonseca, J. Salamon, P. Seetharaman, and J. R. Hershey. What's all the FUSS about free universal sound separation data? *In preparation*, 2020.

[45] S. Wisdom, J. R. Hershey, K. Wilson, J. Thorpe, M. Chinen, B. Patton, and R. A. Saurous. Differentiable consistency constraints for improved deep speech enhancement. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 900–904, 2019.

[46] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen. Permutation invariant training of deep models for speaker-independent multi-talker speech separation. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 241–245, 2017.

[47] N. Zeghidour and D. Grangier. Wavesplit: End-to-end speech separation by speaker clustering. *arXiv preprint arXiv:2002.08933*, 2020.

[48] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. In *Proc. International Conference on Learning Representations (ICLR)*, 2018.

A Separation model architecture

In Table 2, we describe the separation network architecture using a TDCN++ [19]. As compared to the original ConvTasNet method [26], the changes to the model include the following:

- Instead of global layer norm, which averages statistics over frames and channels, the TDCN++ uses instance norm, also known as feature-wise global layer norm [19]. This mean-and-variance normalization is performed separately for each convolution channel across frames, with trainable scalar bias and scale parameters.
- The second difference is skip-residual connections from the outputs of earlier residual blocks to form the inputs of the later residual blocks. A skip-residual connection includes a transformation in the form of a dense layer with bias of the block outputs and all paths from residual connections are summed with the regular block input coming from the previous block. Note that all dense layers in the model include bias terms.
- Finally, a scalar scale parameter is applied after each dense layer stage, which is an over-parametrization trick that improves convergence. The scale parameters for the second dense layer in layer i are initialized using exponential decay in the form of 0.9^i. All other scales are initialized to 1.0. This initial scaling controls the contribution of each block into the residual sum. It also causes the initial blocks train faster and the later blocks to train slower, which is reminiscent of layer-wise training.

Table 2: Separation network with TDCN++ architecture configuration. Variables are number of encoder basis coefficients $N = 256$, encoder basis kernel size L, which is 40 for 16 kHz data and 20 for 8 kHz data, number of waveform samples T, number of coefficient frames F, and number of separated sources M.

Module name	Operation	Output shape	Kernel size	Dilation	Stride
Waveform	Input	$T \times 1$	–	–	–
Encoder	Conv	$F \times N$	$1 \times L \times N$	1	$L/2$
Coeffs	Intermediate	$F \times N$	–	–	–
Initial bottleneck	ReLU	$F \times N$	–	–	–
	Dense	$F \times 256$	$N \times 256$	1	1
i-th separable dilated	Input	$F \times 256$	Previous block output		
conv block (x32)			+ sum of skip-residual inputs		
	Dense	$F \times 512$	256×512	–	–
with skip-residual	Scale	$F \times 512$	1×1	–	–
connections b/w blocks:	PReLU	$F \times 512$	–	–	–
$i \to i+1$,	Instance norm	$F \times 512$	–	–	–
$0 \to 8$, $0 \to 16$, $0 \to 24$,	Depthwise conv	$F \times 512$	$512 \times 3 \times 1$	$2^{\mathrm{mod}(i,8)}$	1
$8 \to 16$, $8 \to 24$,	PReLU	$F \times 512$	–	–	–
$16 \to 24$,	Instance norm	$F \times 512$	–	–	–
	Dense	$F \times 256$	512×256	–	–
	Scale	$F \times 512$	1×1	–	–
Final bottleneck	Dense	$F \times 256$	512×256	–	–
Perform masking	Dense	$F \times M \cdot N$	$256 \times M \cdot N$	–	–
	Sigmoid	$F \times M \cdot N$	–	–	–
	Reshape	$F \times M \times N$	–	–	–
	Multiply	$F \times M \times N$	Multiply with $F \times 1 \times N$ coeffs		
Decoder	Transposed conv	$T \times M$	$L \times N \times 1$	1	$L/2$
Separated waveforms	Output	$T \times M$	–	–	–

As mentioned in the text, we also apply a mixture consistency projection [45] to the resulting separated waveforms, which projects them such that they sum up to the original mixture. This projection solves the following optimization problem to find mixture consistency separated sources $\hat{\mathbf{s}}$ given initial separated sources $\underline{\mathbf{s}}$ separated by the model from a mixture $\mathbf{x}$:

$$\begin{aligned} \underset{\hat{\mathbf{s}} \in \mathbb{R}^{M \times T}}{\text{minimize}} \quad & \frac{1}{2} \sum_m \|\hat{\mathbf{s}}_m - \underline{\mathbf{s}}_m\|^2 \\ \text{subject to} \quad & \sum_m \hat{\mathbf{s}}_m = \mathbf{x}. \end{aligned} \tag{5}$$

The projection operation is the closed-form solution of this problem:

$$\hat{\mathbf{s}}_m = \underline{\mathbf{s}}_m + \frac{1}{M}(\mathbf{x} - \sum_{m'} \underline{\mathbf{s}}_{m'}), \tag{6}$$

which is differentiable and can simply be applied as a final layer to the initial separated sources $\underline{\mathbf{s}}$.

B Training details

For each task, we train all models to 200k steps, evaluating a checkpoint every 10 minutes. For evaluation on the test set, we select the checkpoint with the highest validation score. As mentioned in the text, all models are trained with batch size 256 with learning rate 10^{-3} on 4 Google Cloud TPUs (16 chips).

C Ablations

In order to evaluate the contribution of different components of the proposed model we compare several variations trained on WSJ0-2mix with two-source mixtures: disabling mixture consistency, and varying $\mathrm{SNR}_{\mathrm{max}}$. Performance is reported on the validation set after 200k training steps.

Mixture consistency We observed modest improvement of 0.5 dB SI-SNRi by incorporating mixture consistency (6) versus not.

SNR threshold Performance is not very sensitive to $\mathrm{SNR}_{\mathrm{max}}$ as long as it is 20 dB or larger, as shown in Table 3.

Table 3: SI-SNRi in dB as a function of $\mathrm{SNR}_{\mathrm{max}}$ for unsupervised MixIT on WSJ0-2mix 2-source mixtures.

$\mathrm{SNR}_{\mathrm{max}}$	10	20	30	40	50
SI-SNRi	13.1	13.8	13.7	13.6	13.7

Zero source loss For speech separation tasks using 1-to-2-source mixtures and for universal sound separation on FUSS, the separation model needs to be able to output near-zero signals for "inactive" source slots. Following the implementation of the baseline FUSS separation model [43], we experimented with using explicit losses on separated signals that align to all-zeros reference sources for supervised training examples. Following the baseline FUSS implementation, we chose to use the prescribed modification of the negative SNR loss function (2), where the mixture signal $\mathbf{x}$ instead of the source signal $\mathbf{s}$ is used to determine the soft-thresholding, where we still set τ corresponding to $\mathrm{SNR}_{\mathrm{max}}$ of 30 dB:

$$\mathcal{L}_0(\mathbf{s}=\mathbf{0},\hat{\mathbf{s}},\mathbf{x}) = 10\log_{10}\left(\|\hat{\mathbf{s}}\|^2 + \tau\|\mathbf{x}\|^2\right), \tag{7}$$

which means the loss will be clipped when the power of the separated signal drops 30 dB below the power of the mixture signal. We also experimented with changing the value of p_0, the probability of zeroing out one of the mixtures and its corresponding reference signals for supervised examples.

Table 4: Effect of incorporating the zero loss $\mathcal{L}_0$ (7) on supervised separation performance on the WSJ0-2mix and FUSS validation sets without additional reverb after 200k training steps.

Dataset	p_0	$\mathcal{L}_0$	SI-SNRi	MSi	SS
WSJ0-2mix 2-source mixtures	0.0	✗	15.9		
	0.0	✓	14.3		
FUSS	0.0	✗		12.3	10.9
	0.0	✓		12.0	12.0
	0.2	✗		12.6	29.5
	0.2	✓		12.3	34.7

The results are shown in Table 4 for WSJ0-2mix and FUSS, where the models are trained on mixtures of mixtures, and evaluated on single mixtures from the validation set. Notice that incorporating $\mathcal{L}_0$ decreases SI-SNRi on WSJ0-2mix and MSi on FUSS; however, it boosts SS performance, which is probably due to better suppression of inactive source power. Because we also use mixture consistency, reduction in inactive source power should allow the network to allocate more power to the reconstructed single source. For FUSS, using a p_0 greater than 0 slightly improves MSi, and greatly improves SS. This is because the network is presented with actual single mixtures during training, which improves the match between train and test.