Conv-TasNet: Surpassing Ideal Time–Frequency Magnitude Masking for Speech Separation

Yi Luo and Nima Mesgarani

Abstract—**Single-channel, speaker-independent speech separation methods have recently seen great progress. However, the accuracy, latency, and computational cost of such methods remain insufficient. The majority of the previous methods have formulated the separation problem through the time–frequency representation of the mixed signal, which has several drawbacks, including the decoupling of the phase and magnitude of the signal, the suboptimality of time–frequency representation for speech separation, and the long latency in calculating the spectrograms. To address these shortcomings, we propose a fully convolutional time-domain audio separation network (Conv-TasNet), a deep learning framework for end-to-end time-domain speech separation. Conv-TasNet uses a linear encoder to generate a representation of the speech waveform optimized for separating individual speakers. Speaker separation is achieved by applying a set of weighting functions (masks) to the encoder output. The modified encoder representations are then inverted back to the waveforms using a linear decoder. The masks are found using a temporal convolutional network consisting of stacked one-dimensional dilated convolutional blocks, which allows the network to model the long-term dependencies of the speech signal while maintaining a small model size. The proposed Conv-TasNet system significantly outperforms previous time–frequency masking methods in separating two- and three-speaker mixtures. Additionally, Conv-TasNet surpasses several ideal time–frequency magnitude masks in two-speaker speech separation as evaluated by both objective distortion measures and subjective quality assessment by human listeners. Finally, Conv-TasNet has a significantly smaller model size and a shorter minimum latency, making it a suitable solution for both offline and real-time speech separation applications. This study, therefore, represents a major step toward the realization of speech separation systems for real-world speech processing technologies.**

Index Terms—**Source separation, single-channel, time-domain, deep learning, real-time.**

I. Introduction

ROBUST speech processing in real-world acoustic environments often requires automatic speech separation. Because of the importance of this research topic for speech processing technologies, numerous methods have been proposed for solving this problem. However, the accuracy of speech separation, particularly for new speakers, remains inadequate.

Most previous speech separation approaches have been formulated in the time-frequency (T-F, or spectrogram) representation of the mixture signal, which is estimated from the waveform using the short-time Fourier transform (STFT) [1]. Speech separation methods in the T-F domain aim to approximate the clean spectrogram of the individual sources from the mixture spectrogram. This process can be performed by directly approximating the spectrogram representation of each source from the mixture using nonlinear regression techniques, where the clean source spectrograms are used as the training target [2]–[4]. Alternatively, a weighting function (mask) can be estimated for each source to multiply each T-F bin in the mixture spectrogram to recover the individual sources. In recent years, deep learning has greatly advanced the performance of time-frequency masking methods by increasing the accuracy of the mask estimation [5]–[12]. In both the direct method and the mask estimation method, the waveform of each source is calculated using the inverse short-time Fourier transform (iSTFT) of the estimated magnitude spectrogram of each source together with either the original or the modified phase of the mixture sound.

While time-frequency masking remains the most commonly used method for speech separation, this method has several shortcomings. First, STFT is a generic signal transformation that is not necessarily optimal for speech separation. Second, accurate reconstruction of the phase of the clean sources is a nontrivial problem, and the erroneous estimation of the phase introduces an upper bound on the accuracy of the reconstructed audio. This issue is evident by the imperfect reconstruction accuracy of the sources even when the ideal clean magnitude spectrograms are applied to the mixture. Although methods for phase reconstruction can be applied to alleviate this issue [11], [13], [14], the performance of the method remains suboptimal. Third, successful separation from the time-frequency representation requires a high-resolution frequency decomposition of the mixture signal, which requires a long temporal window for the calculation of STFT. This requirement increases the minimum latency of the system, which limits its applicability in real-time, low-latency applications such as in telecommunication and hearable devices. For example, the window length of STFT in most speech separation systems is at least 32 ms [5], [7], [8] and is even greater in music separation applications, which require an even higher resolution spectrogram (higher than 90 ms) [15], [16].

Because these issues arise from formulating the separation problem in the time-frequency domain, a logical approach is to avoid decoupling the magnitude and the phase of the sound by directly formulating the separation in the time domain. Previous

Manuscript received September 24, 2018; revised February 25, 2019 and April 21, 2019; accepted April 23, 2019. Date of publication May 6, 2019; date of current version May 21, 2019. This work was supported in part by a grant from the National Institute of Health, NIDCD, DC014279; in part by a National Science Foundation CAREER Award; and in part by the Pew Charitable Trusts. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Tan Lee. (*Corresponding author: Nima Mesgarani.*)

The authors are with the Department of Electrical Engineering, Columbia University, New York, NY 10027 USA (e-mail: yl3364@columbia.edu; nima@ee.columbia.edu).

Digital Object Identifier 10.1109/TASLP.2019.2915167

2329-9290 © 2019 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

studies have explored the feasibility of time-domain speech separation through methods such as independent component analysis (ICA) [17] and time-domain non-negative matrix factorization (NMF) [18]. However, the performance of these systems has not been comparable with the performance of time-frequency approaches, particularly in terms of their ability to scale and generalize to large data. On the other hand, a few recent studies have explored deep learning for time-domain audio separation [19]–[21]. The shared idea in all these systems is to replace the STFT step for feature extraction with a data-driven representation that is jointly optimized with an end-to-end training paradigm. These representations and their inverse transforms can be explicitly designed to replace STFT and iSTFT. Alternatively, feature extraction together with separation can be implicitly incorporated into the network architecture, for example by using an end-to-end convolutional neural network (CNN) [22], [23]. These methods are different in how they extract features from the waveform and in terms of the design of the separation module. In [19], a convolutional encoder motivated by discrete cosine transform (DCT) is used as the front-end. The separation is then performed by passing the encoder features to a multi-layer perceptron (MLP). The reconstruction of the waveforms is achieved by inverting the encoder operation. In [20], the separation is incorporated into a U-Net 1-D CNN architecture [24] without explicitly transforming the input into a spectrogram-like representation. However, the performance of these methods on a large speech corpus such as the benchmark introduced in [25] has not been tested. Another such method is the time-domain audio separation network (TasNet) [21], [26]. In TasNet, the mixture waveform is modeled with a convolutional encoder-decoder architecture, which consists of an encoder with a non-negativity constraint on its output and a linear decoder for inverting the encoder output back to the sound waveform. This framework is similar to the ICA method when a non-negative mixing matrix is used [27] and to the semi-nonnegative matrix factorization method (semi-NMF) [28], where the basis signals are the parameters of the decoder. The separation step in TasNet is done by finding a weighting function for each source (similar to time-frequency masking) for the encoder output at each time step. It has been shown that TasNet has achieved better or comparable performance with various previous T-F domain systems, showing its effectiveness and potential.

While TasNet outperformed previous time-frequency speech separation methods in both causal and non-causal implementations, the use of a deep long short-term memory (LSTM) network as the separation module in the original TasNet significantly limited its applicability. First, choosing smaller kernel size (i.e. length of the waveform segments) in the encoder increases the length of the encoder output, which makes the training of the LSTMs unmanageable. Second, the large number of parameters in deep LSTM network significantly increases its computational cost and limits its applicability to low-resource, low-power platforms such as wearable hearing devices. The third problem which we will illustrate in this paper is caused by the long temporal dependencies of LSTM networks which often results in inconsistent separation accuracy, for example, when changing the starting point of the mixture. To alleviate the limitations of the previous TasNet, we propose the fully-convolutional TasNet (Conv-TasNet) that uses only convolutional layers in all stages of processing. Motivated by the success of temporal convolutional network (TCN) models [29]–[31], Conv-TasNet uses stacked dilated 1-D convolutional blocks to replace the deep LSTM networks for the separation step. The use of convolution allows parallel processing on consecutive frames or segments to greatly speed up the separation process and also significantly reduces the model size. To further decrease the number of parameters and the computational cost, we substitute the original convolution operation with depthwise separable convolution [32], [33]. We show that with these modifications, Conv-TasNet significantly increases the separation accuracy over the previous LSTM-TasNet in both causal and non-causal implementations. Moreover, the separation accuracy of Conv-TasNet surpasses the performance of ideal time-frequency magnitude masks, including the ideal binary mask (IBM [34]), ideal ratio mask (IRM [35], [36]), and Winener filter-like mask (WFM [37]) in both signal-to-distortion ratio (SDR) and subjective (mean opinion score, MOS) measures.

Reasons not to use LSTM

The rest of the paper is organized as follows. We introduce the proposed Conv-TasNet in Section II, describe the experimental procedures in Section III, and show the experimental results and analysis in Section IV.

II. Convolutional Time-Domain Audio Separation Network

The fully-convolutional time-domain audio separation network (Conv-TasNet) consists of three processing stages, as shown in Fig. 1(A): encoder, separation, and decoder. First, an encoder module is used to transform short segments of the mixture waveform into their corresponding representations in an intermediate feature space. This representation is then used to estimate a multiplicative function (mask) for each source at each time step. The source waveforms are then reconstructed by transforming the masked encoder features using a decoder module. We describe the details of each stage in this section.

A. *Time-Domain Speech Separation*

The problem of single-channel speech separation can be formulated in terms of estimating C sources $s_1(t), \ldots, s_c(t) \in \mathbb{R}^{1\times T}$, given the discrete waveform of the mixture $x(t) \in \mathbb{R}^{1\times T}$, where

$$x(t) = \sum_{i=1}^{C} s_i(t). \tag{1}$$

In time-domain audio separation, we aim to directly estimate $s_i(t), i = 1, \ldots, C$, from $x(t)$.

B. *Convolutional Encoder-Decoder*

The input mixture sound can be divided into overlapping segments of length L, represented by $\mathbf{x}_k \in \mathbb{R}^{1\times L}$, where $k = 1, \ldots, \hat{T}$ denotes the segment index and $\hat{T}$ denotes the total number of segments in the input. $\mathbf{x}_k$ is transformed into a N-dimensional representation, $\mathbf{w} \in \mathbb{R}^{1\times N}$ by a 1-D convolution

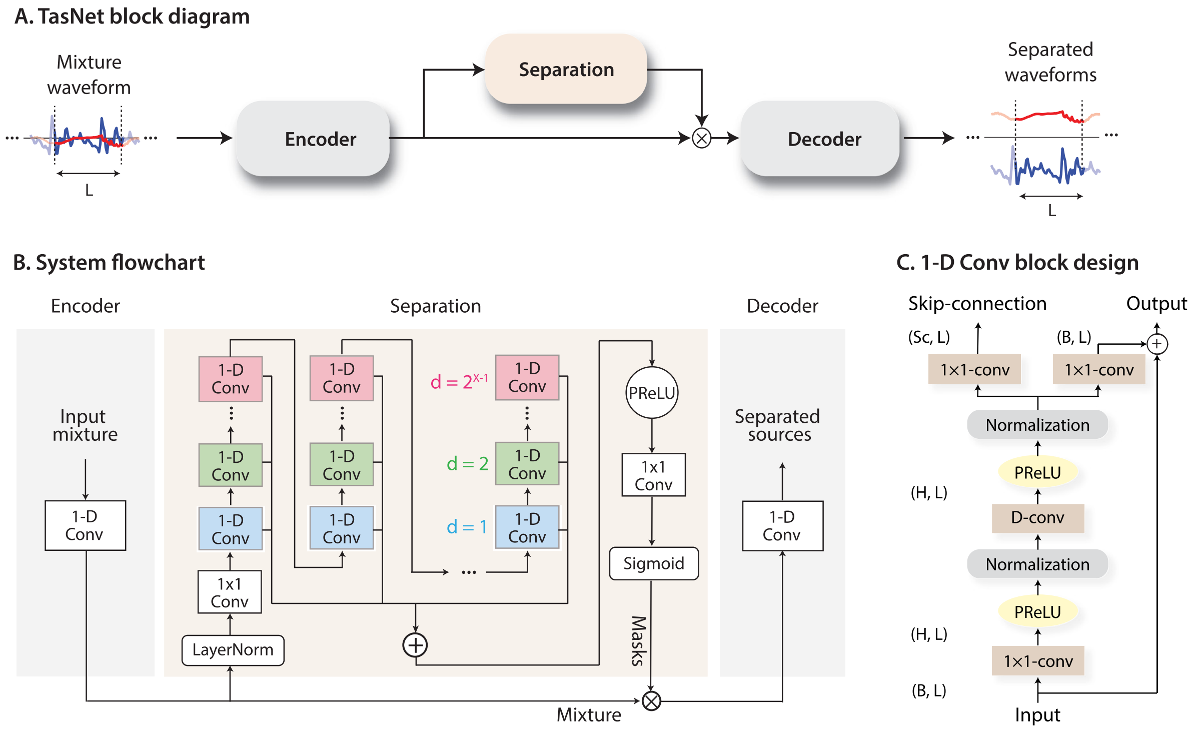

Fig. 1. (A) The block diagram of the TasNet system. An encoder maps a segment of the mixture waveform to a high-dimensional representation and a separation module calculates a multiplicative function (i.e., a mask) for each of the target sources. A decoder reconstructs the source waveforms from the masked features. (B) A flowchart of the proposed system. A 1-D convolutional autoencoder models the waveforms and a temporal convolutional network (TCN) separation module estimates the masks based on the encoder output. Different colors in the 1-D convolutional blocks in TCN denote different dilation factors. (C) The design of 1-D convolutional block. Each block consists of a 1×1-*conv* operation followed by a depthwise convolution ($D - conv$) operation, with nonlinear activation function and normalization added between each two convolution operations. Two linear $1 \times 1 - conv$ blocks serve as the residual path and the skip-connection path respectively.

operation, which is reformulated as a matrix multiplication (the index k is dropped from now on):

$$\mathbf{w} = \mathcal{H}(\mathbf{x}\mathbf{U}^{\mathrm{T}}) \tag{2}$$

where $\mathbf{U} \in \mathbb{R}^{N \times L}$ contains N vectors (encoder basis functions) with length L each, and $\mathcal{H}(\cdot)$ is an optional nonlinear function. In [21], [26], $\mathcal{H}(\cdot)$ was the rectified linear unit (ReLU) to ensure that the representation is non-negative. The decoder reconstructs the waveform from this representation using a 1-D transposed convolution operation, which can be reformulated as another matrix multiplication:

$$\hat{\mathbf{x}} = \mathbf{w}\mathbf{V} \tag{3}$$

where $\hat{\mathbf{x}} \in \mathbb{R}^{1 \times L}$ is the reconstruction of $\mathbf{x}$, and the rows in $\mathbf{V} \in \mathbb{R}^{N \times L}$ are the decoder basis functions, each with length L. The overlapping reconstructed segments are summed together to generate the final waveforms.

Although we reformulate the encoder/decoder operations as matrix multiplication, the term "convolutional autoencoder" is used because in actual model implementation, convolutional and transposed convolutional layers can more easily handle the overlap between segments and thus enable faster training and better convergence.[1]

C. Estimating the Separation Masks

The separation for each frame is performed by estimating C vectors (masks) $\mathbf{m}_i \in \mathbb{R}^{1 \times N}, i = 1, \ldots, C$ where C is the number of speakers in the mixture that is multiplied by the encoder output $\mathbf{w}$. The mask vectors $\mathbf{m}_i$ have the constraint that $\mathbf{m}_i \in [0, 1]$. The representation of each source, $\mathbf{d}_i \in \mathbb{R}^{1 \times N}$, is then calculated by applying the corresponding mask, $\mathbf{m}_i$, to the mixture representation $\mathbf{w}$:

$$\mathbf{d}_i = \mathbf{w} \odot \mathbf{m}_i \tag{4}$$

where $\odot$ denotes element-wise multiplication. The waveform of each source $\hat{\mathbf{s}}_i, i = 1, \ldots, C$ is then reconstructed by the decoder:

$$\hat{\mathbf{s}}_i = \mathbf{d}_i \mathbf{V}. \tag{5}$$

The unit summation constraint in [21], [26], $\sum_{i=1}^{C} \mathbf{m}_i = \mathbf{1}$, was applied based on the assumption that the encoder-encoder architecture can perfectly reconstruct the input mixture. In Section IV-A, we will examine the consequence of relaxing this unity summation constraint on separation accuracy.

D. Convolutional Separation Module

Motivated by the temporal convolutional network (TCN) [29]–[31], we propose a fully-convolutional separation module that consists of stacked 1-D dilated convolutional blocks, as shown in Fig. 1(B). TCN was proposed as a replacement for

[1] With our Pytorch implementation, this is possibly due to the different *autograd* mechanisms in fully-connected layer and 1-D (transposed) convolutional layers.

RNNs in various sequence modeling tasks. Each layer in a TCN consists of 1-D convolutional blocks with increasing dilation factors. The dilation factors increase exponentially to ensure a sufficiently large temporal context window to take advantage of the long-range dependencies of the speech signal, as denoted with different colors in Fig. 1(B). In Conv-TasNet, M convolutional blocks with dilation factors $1, 2, 4, \ldots, 2^{M-1}$ are repeated R times. The input to each block is zero padded accordingly to ensure the output length is the same as the input. The output of the TCN is passed to a convolutional block with kernel size 1 ($1 \times 1-conv$ block, also known as *pointwise* convolution) for mask estimation. The $1 \times 1-conv$ block together with a nonlinear activation function estimates C mask vectors for the C target sources.

Fig. 1(C) shows the design of each 1-D convolutional block. The design of the 1-D convolutional blocks follows [38], where a residual path and a skip-connection path are applied: the residual path of a block serves as the input to the next block, and the skip-connection paths for all blocks are summed up and used as the output of the TCN. To further decrease the number of parameters, depthwise separable convolution ($S\text{-}conv(\cdot)$) is used to replace standard convolution in each convolutional block. Depthwise separable convolution (also referred to as separable convolution) has proven effective in image processing tasks [32], [33] and neural machine translation tasks [39]. The depthwise separable convolution operator decouples the standard convolution operation into two consecutive operations, a depthwise convolution ($D\text{-}conv(\cdot)$) followed by pointwise convolution ($1 \times 1-conv(\cdot)$):

$$D\text{-}conv(\mathbf{Y}, \mathbf{K}) = concat(\mathbf{y}_j \circledast \mathbf{k}_j), j = 1, \ldots, N \tag{6}$$

$$S\text{-}conv(\mathbf{Y}, \mathbf{K}, \mathbf{L}) = D\text{-}conv(\mathbf{Y}, \mathbf{K}) \circledast \mathbf{L} \tag{7}$$

where $\mathbf{Y} \in \mathbb{R}^{G \times M}$ is the input to $S\text{-}conv(\cdot)$, $\mathbf{K} \in \mathbb{R}^{G \times P}$ is the convolution kernel with size P, $\mathbf{y}_j \in \mathbb{R}^{1 \times M}$ and $\mathbf{k}_j \in \mathbb{R}^{1 \times P}$ are the rows of matrices $\mathbf{Y}$ and $\mathbf{K}$, respectively, $\mathbf{L} \in \mathbb{R}^{G \times H \times 1}$ is the convolution kernel with size 1, and $\circledast$ denotes the convolution operation. In other words, the $D\text{-}conv(\cdot)$ operation convolves each row of the input Y with the corresponding row of matrix K, and the $1 \times 1-conv$ block linearly transforms the feature space. In comparison with the standard convolution with kernel size $\hat{\mathbf{K}} \in \mathbb{R}^{G \times H \times P}$, depthwise separable convolution only contains $G \times P + G \times H$ parameters, which decreases the model size by a factor of $\frac{H \times P}{H + P} \approx P$ when $H \gg P$.

A nonlinear activation function and a normalization operation are added after both the first $1 \times 1\text{-}conv$ and $D\text{-}conv$ blocks respectively. The nonlinear activation function is the parametric rectified linear unit (PReLU) [40]:

$$PReLU(x) = \begin{cases} x, & \text{if } x \geq 0 \\ \alpha x, & \text{otherwise} \end{cases} \tag{8}$$

where $\alpha \in \mathbb{R}$ is a trainable scalar controlling the negative slope of the rectifier. The choice of the normalization method in the network depends on the causality requirement. For noncausal configuration, we found empirically that global layer normalization (gLN) outperforms all other normalization methods. In gLN, the feature is normalized over both the channel and the time dimensions:

$$gLN(\mathbf{F}) = \frac{\mathbf{F} - E[\mathbf{F}]}{\sqrt{Var[\mathbf{F}] + \epsilon}} \odot \gamma + \beta \tag{9}$$

$$E[\mathbf{F}] = \frac{1}{NT} \sum_{NT} \mathbf{F} \tag{10}$$

$$Var[\mathbf{F}] = \frac{1}{NT} \sum_{NT} (\mathbf{F} - E[\mathbf{F}])^2 \tag{11}$$

where $\mathbf{F} \in \mathbb{R}^{N \times T}$ is the feature, $\gamma, \beta \in \mathbb{R}^{N \times 1}$ are trainable parameters, and ϵ is a small constant for numerical stability. This is identical to the standard layer normalization applied in computer vision models where the channel and time dimension correspond to the width and height dimension in an image [41]. In causal configuration, gLN cannot be applied since it relies on the future values of the signal at any time step. Instead, we designed a cumulative layer normalization (cLN) operation to perform step-wise normalization in the causal system:

$$cLN(\mathbf{f}_k) = \frac{\mathbf{f}_k - E[\mathbf{f}_{t \leq k}]}{\sqrt{Var[\mathbf{f}_{t \leq k}] + \epsilon}} \odot \gamma + \beta \tag{12}$$

$$E[\mathbf{f}_{t \leq k}] = \frac{1}{Nk} \sum_{Nk} \mathbf{f}_{t \leq k} \tag{13}$$

$$Var[\mathbf{f}_{t \leq k}] = \frac{1}{Nk} \sum_{Nk} (\mathbf{f}_{t \leq k} - E[\mathbf{f}_{t \leq k}])^2 \tag{14}$$

where $\mathbf{f}_k \in \mathbb{R}^{N \times 1}$ is the k-th frame of the entire feature $\mathbf{F}$, $\mathbf{f}_{t \leq k} \in \mathbb{R}^{N \times k}$ corresponds to the feature of k frames $[\mathbf{f}_1, \ldots, \mathbf{f}_k]$, and $\gamma, \beta \in \mathbb{R}^{N \times 1}$ are trainable parameters applied to all frames. To ensure that the separation module is invariant to the scaling of the input, the selected normalization method is applied to the encoder output $\mathbf{w}$ before it is passed to the separation module.

At the beginning of the separation module, a linear $1 \times 1\text{-}conv$ block is added as a bottleneck layer. This block determines the number of channels in the input and residual path of the subsequent convolutional blocks. For instance, if the linear bottleneck layer has B channels, then for a 1-D convolutional block with H channels and kernel size P, the size of the kernel in the first $1 \times 1\text{-}conv$ block and the first $D\text{-}conv$ block should be $\mathbf{O} \in \mathbb{R}^{B \times H \times 1}$ and $\mathbf{K} \in \mathbb{R}^{H \times P}$ respectively, and the size of the kernel in the residual paths should be $\mathbf{L}_{Rs} \in \mathbb{R}^{H \times B \times 1}$. The number of output channels in the skip-connection path can be different than B, and we denote the size of kernels in that path as $\mathbf{L}_{Sc} \in \mathbb{R}^{H \times Sc \times 1}$.

III. Experimental Procedures

A. Dataset

We evaluated our system on two-speaker and three-speaker speech separation problems using the WSJ0-2mix and WSJ0-3mix datasets [25]. 30 hours of training and 10 hours of validation data are generated from speakers in si_tr_s from the datasets. The speech mixtures are generated by randomly selecting utterances from different speakers in the Wall Street Journal dataset (WSJ0) and mixing them at random signal-to-noise ratios (SNR) between −5 dB and 5 dB. 5 hours of evaluation set is generated

TABLE I
HYPERPARAMETERS OF THE NETWORK

Symbol	Description
N	Number of filters in autoencoder
L	Length of the filters (in samples)
B	Number of channels in bottleneck and the residual paths' 1×1-$conv$ blocks
Sc	Number of channels in skip-connection paths' 1×1-$conv$ blocks
H	Number of channels in convolutional blocks
P	Kernel size in convolutional blocks
X	Number of convolutional blocks in each repeat
R	Number of repeats

in the same way using utterances from 16 unseen speakers in si_dt_05 and si_et_05. The scripts for creating the dataset can be found at [42]. All the waveforms are resampled at 8 kHz.

B. Experiment Configurations

The networks are trained for 100 epochs on 4-second long segments. The initial learning rate is set to $1e^{-3}$. The learning rate is halved if the accuracy of validation set is not improved in 3 consecutive epochs. Adam [43] is used as the optimizer. A 50% stride size is used in the convolutional autoencoder (i.e. 50% overlap between consecutive frames). Gradient clipping with maximum L_2-norm of 5 is applied during training. The hyperparameters of the network are shown in Table I. A Pytorch implementation of the Conv-TasNet model can be found at.[2]

C. Training Objective

The objective of training the end-to-end system is maximizing the scale-invariant source-to-noise ratio (SI-SNR), which has commonly been used as the evaluation metric for source separation replacing the standard source-to-distortion ratio (SDR) [5], [9], [44]. SI-SNR is defined as:

$$\begin{cases} \mathbf{s}_{target} := \frac{\langle \hat{\mathbf{s}}, \mathbf{s} \rangle \mathbf{s}}{\|\mathbf{s}\|^2} \\ \mathbf{e}_{noise} := \hat{\mathbf{s}} - \mathbf{s}_{target} \\ \text{SI-SNR} := 10\, log_{10} \frac{\|\mathbf{s}_{target}\|^2}{\|\mathbf{e}_{noise}\|^2} \end{cases} \tag{15}$$

where $\hat{\mathbf{s}} \in \mathbb{R}^{1 \times T}$ and $\mathbf{s} \in \mathbb{R}^{1 \times T}$ are the estimated and original clean sources, respectively, and $\|\mathbf{s}\|^2 = \langle \mathbf{s}, \mathbf{s} \rangle$ denotes the signal power. Scale invariance is ensured by normalizing $\hat{\mathbf{s}}$ and $\mathbf{s}$ to zero-mean prior to the calculation. Utterance-level permutation invariant training (uPIT) is applied during training to address the source permutation problem [7].

D. Evaluation Metrics

We report the scale-invariant signal-to-noise ratio improvement (SI-SNRi) and signal-to-distortion ratio improvement (SDRi) [44] as objective measures of separation accuracy. SI-SNR is defined in equation 15. The reported improvements in Tables III to V indicate the additive values over the original mixture. In addition to the distortion metrics, we also evaluated the quality of the separated mixtures using both the perceptual evaluation of subjective quality (PESQ, [45]) and the mean opinion score (MOS) [46] by asking 40 normal hearing subjects to rate the quality of the separated mixtures. All human testing procedures were approved by the local institutional review board (IRB) at Columbia University in the City of New York.

E. Comparison With Ideal Time-Frequency Masks

Following the common configurations in [5], [7], [9], the ideal time-frequency masks were calculated using STFT with a 32 ms window size and 8 ms hop size with a Hanning window. The ideal masks include the ideal binary mask (IBM), ideal ratio mask (IRM), and Wiener filter-like mask (WFM), which are defined for source i as:

$$IBM_i(f,t) = \begin{cases} 1, \ |\mathcal{S}_i(f,t)| > |\mathcal{S}_{j \neq i}(f,t)| \\ 0, \ \text{otherwise} \end{cases} \tag{16}$$

$$IRM_i(f,t) = \frac{|\mathcal{S}_i(f,t)|}{\sum_{j=1}^{C} |\mathcal{S}_j(f,t)|} \tag{17}$$

$$WFM_i(f,t) = \frac{|\mathcal{S}_i(f,t)|^2}{\sum_{j=1}^{C} |\mathcal{S}_j(f,t)|^2} \tag{18}$$

where $\mathcal{S}_i(f,t) \in \mathbb{C}^{F \times T}$ are the complex-valued spectrograms of clean sources $i = 1, \ldots, C$.

IV. RESULTS

Fig. 2 visualizes all the internal variables of Conv-TasNet for one example mixture sound with two overlapping speakers (denoted by red and blue). The encoder and decoder basis functions are sorted by the similarity of the Euclidean distance of the basis functions found using the unweighted pair group method with arithmetic mean (UPGMA) method [47]. The basis functions show a diversity of frequency and phase tuning. The representation of the encoder is colored according to the power of each speaker at the corresponding basis output at each time point, demonstrating the sparsity of the encoder representation. As can be seen in Fig. 2, the estimated masks for the two speakers highly resemble their encoder representations, which allows for the suppression of the encoder outputs that correspond to the interfering speaker and the extraction of the target speaker in each mask. The separated waveforms for the two speakers are estimated by the linear decoder, whose basis functions are shown in Fig. 2. The separated waveforms are shown on the right.

A. Non-Negativity of the Encoder Output

The non-negativity of the encoder output was enforced in [21], [26] using a rectified-linear nonlinearity (ReLU) function. This constraint was based on the assumption that the masking operation on the encoder output is only meaningful when the mixture and speaker waveforms can be represented with a non-negative combination of the basis functions, since an unbounded

[2] https://github.com/naplab/Conv-TasNet

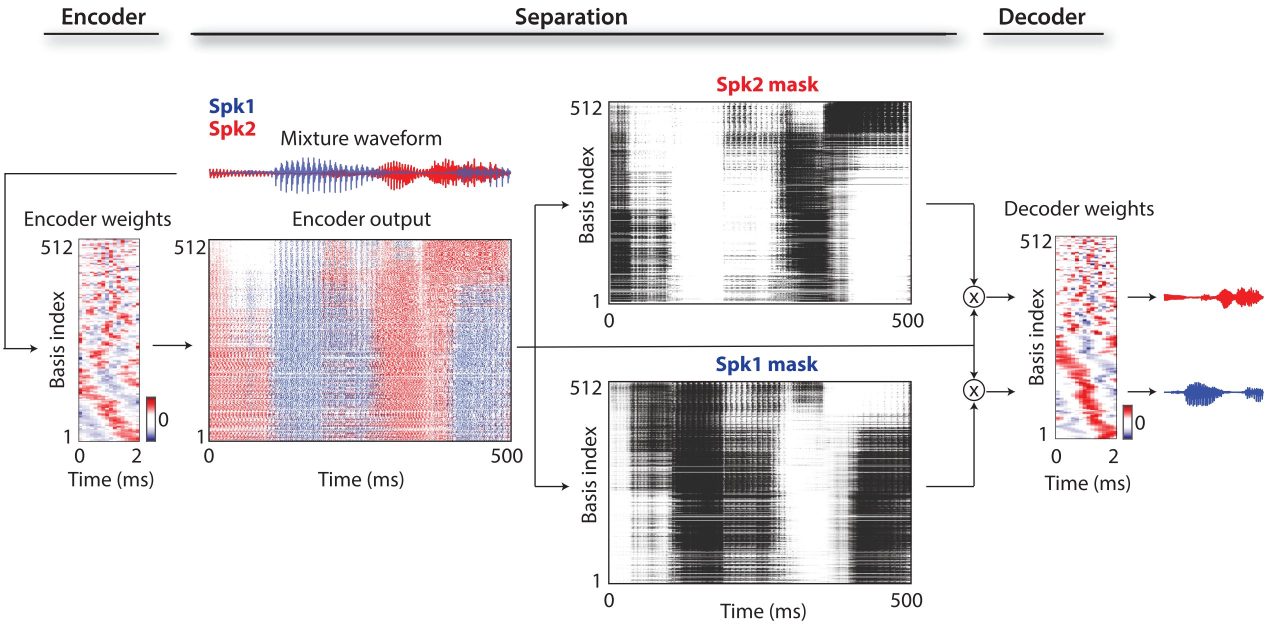

Fig. 2. Visualization of the encoder and decoder basis functions, encoder representation, and source masks for a sample 2-speaker mixture. The speakers are shown in red and blue. The encoder representation is colored according to the power of each speaker at each basis function and point in time. The basis functions are sorted according to their Euclidean similarity and show diversity in frequency and phase tuning.

encoder representation may result in unbounded masks. However, by removing the nonlinear function $\mathcal{H}$, another assumption can be made: with an unbounded but highly overcomplete representation of the mixture, a set of non-negative masks can still be found to reconstruct the clean sources. In this case, the overcompleteness of the representation is crucial. If there exist only a unique weight feature for the mixture as well as for the sources, the non-negativity of the mask cannot be guaranteed. Also note that in both assumptions, we put no constraint on the relationship between the encoder and decoder basis functions $\mathbf{U}$ and $\mathbf{V}$, meaning that they are not forced to reconstruct the mixture signal perfectly. One way to explicitly ensure the autoencoder property is by choosing $\mathbf{V}$ to be the pseudo-inverse of $\mathbf{U}$ (i.e. least square reconstruction). The choice of encoder/decoder design affects the mask estimation: in the case of an autoencoder, the unit summation constraint must be satisfied; otherwise, the unit summation constraint is not strictly required. To illustrate this point, we compared five different encoder-decoder configurations:

1) Linear encoder with its pseudo-inverse (Pinv) as decoder, i.e. $\mathbf{w} = \mathbf{x}(\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T$ and $\hat{\mathbf{x}} = \mathbf{w}\mathbf{V}$, with Softmax function for mask estimation.
2) Linear encoder and decoder where $\mathbf{w} = \mathbf{x}\mathbf{U}$ and $\hat{\mathbf{x}} = \mathbf{w}\mathbf{V}$, with Softmax or Sigmoid function for mask estimation.
3) Encoder with ReLU activation and linear decoder where $\mathbf{w} = ReLU(\mathbf{x}\mathbf{U})$ and $\hat{\mathbf{x}} = \mathbf{w}\mathbf{V}$, with Softmax or Sigmoid function for mask estimation.

Separation accuracy of different configurations in Table III shows that pseudo-inverse autoencoder leads to the worst performance, indicating that an explicit autoencoder configuration does not necessarily improve the separation score in this framework. The performance of all other configurations is comparable. Because linear encoder and decoder with Sigmoid function achieves a slightly better accuracy over other methods, we used this configuration in all the following experiments.

B. *Optimizing the Network Parameters*

We evaluate the performance of Conv-TasNet on two speaker separation tasks as a function of different network parameters. Table II shows the performance of the systems with different parameters, from which we can conclude the following statements:

i) Encoder/decoder: Increasing the number of basis signals in the encoder/decoder increases the overcompleteness of the basis signals and improves the performance.
ii) Hyperparameters in the 1-D convolutional blocks: A possible configuration consists of a small bottleneck size B and a large number of channels in the convolutional blocks H. This matches the observation in [48], where the ratio between the convolutional block and the bottleneck H/B was found to be best around 5. Increasing the number of channels in the skip-connection block improves the performance while greatly increases the model size. Therefore, we selected a small skip-connection block as a trade-off between performance and model size.
iii) Number of 1-D convolutional blocks: When the receptive field is the same, deeper networks lead to better performance, possibly due to the increased model capacity.
iv) Size of receptive field: Increasing the size of receptive field leads to better performance, which shows the importance of modeling the temporal dependencies in the speech signal.

TABLE II
THE EFFECT OF DIFFERENT CONFIGURATIONS IN CONV-TASNET

N	L	B	H	Sc	P	X	R	Normalization	Causal	Receptive field (s)	Model size	SI-SNRi (dB)	SDRi (dB)
128	40	128	256	128	3	7	2	gLN	×	1.28	1.5M	13.0	13.3
256	40	128	256	128	3	7	2	gLN	×	1.28	1.5M	13.1	13.4
512	40	128	256	128	3	7	2	gLN	×	1.28	1.7M	13.3	13.6
512	40	128	256	256	3	7	2	gLN	×	1.28	2.4M	13.0	13.3
512	40	128	512	128	3	7	2	gLN	×	1.28	3.1M	13.3	13.6
512	40	128	512	512	3	7	2	gLN	×	1.28	6.2M	13.5	13.8
512	40	256	256	256	3	7	2	gLN	×	1.28	3.2M	13.0	13.3
512	40	256	512	256	3	7	2	gLN	×	1.28	6.0M	13.4	13.7
512	40	256	512	512	3	7	2	gLN	×	1.28	8.1M	13.2	13.5
512	40	128	512	128	3	6	4	gLN	×	1.27	5.1M	14.1	14.4
512	40	128	512	128	3	4	6	gLN	×	0.46	5.1M	13.9	14.2
512	40	128	512	128	3	8	3	gLN	×	3.83	5.1M	14.5	14.8
512	32	128	512	128	3	8	3	gLN	×	3.06	5.1M	14.7	15.0
512	16	128	512	128	3	8	3	gLN	×	1.53	5.1M	**15.3**	**15.6**
512	16	128	512	128	3	8	3	cLN	✓	1.53	5.1M	10.6	11.0

TABLE III
SEPARATION SCORE FOR DIFFERENT SYSTEM CONFIGURATIONS

Encoder	Mask	Model size	SI-SNRi (dB)	SDRi (dB)
Pinv	Softmax	1.5M	12.1	12.4
Linear	Softmax		12.9	13.2
	Sigmoid		**13.1**	**13.4**
ReLU	Softmax		13.0	13.3
	Sigmoid		12.9	13.2

TABLE IV
COMPARISON WITH OTHER METHODS ON WSJ0-2MIX DATASET

Method	Model size	Causal	SI-SNRi (dB)	SDRi (dB)
DPCL++ [5]	13.6M	×	10.8	–
uPIT-BLSTM-ST [7]	92.7M	×	–	10.0
DANet [8]	9.1M	×	10.5	–
ADANet [9]	9.1M	×	10.4	10.8
cuPIT-Grid-RD [50]	47.2M	×	–	10.2
CBLDNN-GAT [12]	39.5M	×	–	11.0
Chimera++ [10]	32.9M	×	11.5	12.0
WA-MISI-5 [11]	32.9M	×	12.6	13.1
BLSTM-TasNet [26]	23.6M	×	13.2	13.6
Conv-TasNet-gLN	**5.1M**	×	**15.3**	**15.6**
uPIT-LSTM [7]	46.3M	✓	–	7.0
LSTM-TasNet [26]	32.0M	✓	**10.8**	**11.2**
Conv-TasNet-cLN	**5.1M**	✓	10.6	11.0
IRM	–	–	12.2	12.6
IBM	–	–	13.0	13.5
WFM	–	–	13.4	13.8

TABLE V
COMPARISON WITH OTHER SYSTEMS ON WSJ0-3MIX DATASET

Method	Model size	Causal	SI-SNRi (dB)	SDRi (dB)
DPCL++ [5]	13.6M	×	7.1	–
uPIT-BLSTM-ST [7]	92.7M	×	–	7.7
DANet [8]	9.1M	×	8.6	8.9
ADANet [9]	9.1M	×	9.1	9.4
Conv-TasNet-gLN	**5.1M**	×	**12.7**	**13.1**
Conv-TasNet-cLN	**5.1M**	✓	**7.8**	**8.2**
IRM	–	–	12.5	13.0
IBM	–	–	13.2	13.6
WFM	–	–	13.6	14.0

v) Length of each segment: Shorter segment length consistently improves performance. Note that the best system uses a filter length of only 2 ms ($\frac{L}{fs} = \frac{16}{8000} = 0.002s$), which makes it very difficult to train a deep LSTM network with the same L due to the large number of time steps in the encoder output.

vi) Causality: Using a causal configuration leads to a significant drop in the performance. This drop could be due to the causal convolution and/or the layer normalization operations.

C. Comparison of Conv-TasNet With Previous Methods

We compared the separation accuracy of Conv-TasNet with previous methods using SDRi and SI-SNRi. Table IV compares the performance of Conv-TasNet with other state-of-the-art methods on the same WSJ0-2mix dataset. For all systems, we list the best results that have been reported in the literature. The numbers of parameters in different methods are based on our implementations, except for [12] which is provided by the authors. The missing values in the table are either because the numbers were not reported in the study or because the results were calculated with a different STFT configuration. The previous TasNet in [26] is denoted by the (B)LSTM-TasNet. While the BLSTM-TasNet already outperformed IRM and IBM, the non-causal Conv-TasNet significantly surpasses the performance of all three ideal T-F masks in SI-SNRi and SDRi metrics with a significantly smaller model size comparing with all previous methods.

Table V compares the performance of Conv-TasNet with those of other systems on a three-speaker speech separation task involving the WSJ0-3mix dataset. The non-causal Conv-TasNet system significantly outperforms all previous STFT-based systems in SDRi. While there is no prior result on a causal algorithm for three-speaker separation, the causal Conv-TasNet significantly outperforms even the other two non-causal STFT-based systems [5], [7]. Examples of separated audio for two and three speaker mixtures from both causal and non-causal implementations of Conv-TasNet are available online [49].

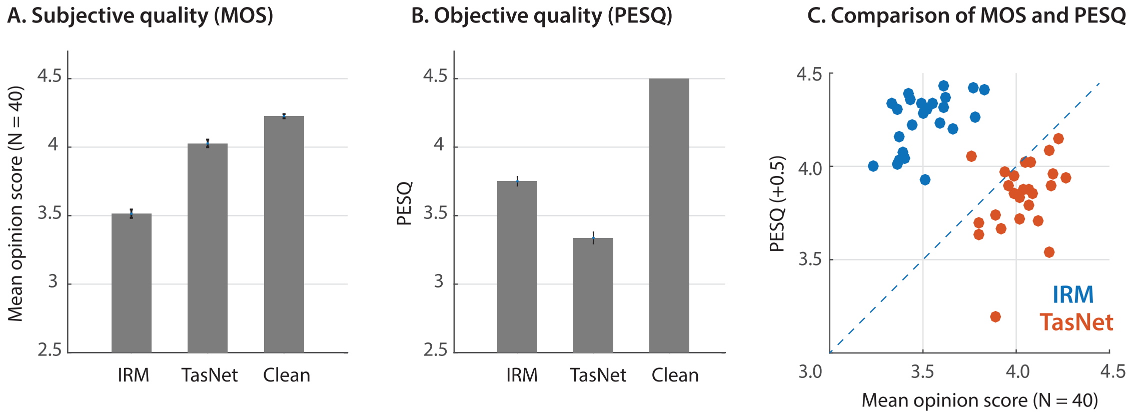

Fig. 3. Subjective and objective quality evaluation of separated utterances in WSJ0-2mix. (A) The mean opinion scores (MOS, N = 40) for IRM, Conv-TasNet and the clean utterance. Conv-TasNet significantly outperforms IRM ($p < 1e - 16$, t-test). (B) PESQ scores are higher for IRM compared to the Conv-TasNet ($p < 1e - 16$, t-test). Error bars indicate standard error (STE) (C) MOS versus PESQ for individual utterances. Each dot denotes one mixture utterance, separated using the IRM (blue) or Conv-TasNet (red). The subjective ratings of almost all utterances for Conv-TasNet are higher than their corresponding PESQ scores.

TABLE VI
PESQ Scores for the Ideal T-F Masks and Conv-TasNet on the Entire WSJ0-2mix and WSJ0-3mix Test Sets

Dataset	PESQ			
	IRM	IBM	WFM	Conv-TasNet
WSJ0-2mix	**3.74**	3.33	3.70	3.24
WSJ0-3mix	**3.52**	2.91	3.45	2.61

TABLE VII
Mean Opinion Score (MOS, N = 40) and PESQ for the 25 Selected Utterances From the WSJ0-2mix Test Set

Method	MOS	PESQ
Conv-TasNet-gLN	**4.03**	3.22
IRM	3.51	**3.74**
Clean	4.23	4.5

TABLE VIII
Processing Time for Causal LSTM-TasNet and Conv-TasNet. The Speed is Evaluated as the Average Time Required to Separate a Frame (Time per Frame, TPF)

Method	CPU/GPU TPF (ms)
LSTM-TasNet	4.3/0.2
Conv-TasNet-cLN	**0.4/0.02**

D. Subjective and Objective Quality Evaluation Of Conv-TasNet

In addition to SDRi and SI-SNRi, we evaluated the subjective and objective quality of the separated speech and compared with three ideal time-frequency magnitude masks. Table VI shows the PESQ score for Conv-TasNet and IRM, IBM, and WFM, where IRM has the highest score for both WSJ0-2mix and WSJ0-3mix dataset. However, since PESQ aims to predict the subjective quality of speech, human quality evaluation can be considered as the ground truth. Therefore, we conducted a psychophysics experiment in which we asked 40 normal hearing subjects to listen and rate the quality of the separated speech sounds. Because of the practical limitations of human psychophysics experiments, we restricted the subjective comparison of Conv-TasNet to the ideal ratio mask (IRM) which has the highest PESQ score among the three ideal masks (Table VI). We randomly chose 25 two-speaker mixture sounds from the two-speaker test set (WSJ0-2mix). We avoided a possible selection bias by ensuring that the average PESQ scores for the IRM and Conv-TasNet separated sounds for the selected 25 samples were equal to the average PESQ scores over the entire test set (comparison of Tables VI and VII). The length of each utterance was constrained to be within 0.5 standard deviation of the mean of the entire test set. The subjects were asked to rate the quality of the clean utterances, the IRM-separated utterances, and the Conv-TasNet separated utterances on the scale of 1 to 5 (1: bad, 2: poor, 3: fair, 4: good, 5: excellent). A clean utterance was first given as the reference for the highest possible score (i.e., 5). Then the clean, IRM, and Conv-TasNet samples were presented to the subjects in random order. The mean opinion score (MOS) of each of the 25 utterances was then averaged over the 40 subjects.

Fig. 3 and Table VII show the result of the human subjective quality test, where the MOS for Conv-TasNet is significantly higher than the MOS for the IRM ($p < 1e - 16$, t-test). In addition, the superior subjective quality of Conv-TasNet over IRM is consistent across most of the 25 test utterances as shown in Fig. 3(C). This observation shows that PESQ consistently underestimates MOS for Conv-TasNet separated utterances, which may be due to the dependence of PESQ on the magnitude spectrogram of speech [45] which could produce lower scores for time-domain approaches.

E. Processing Speed Comparison

Table VIII compares the processing speed of LSTM-TasNet and causal Conv-TasNet. The speed is evaluated as the average processing time for the systems to separate each frame in the mixtures, which we refer to as time per frame (TPF). TPF determines whether a system can be implemented in real time, which requires a TPF that is smaller than the frame length.

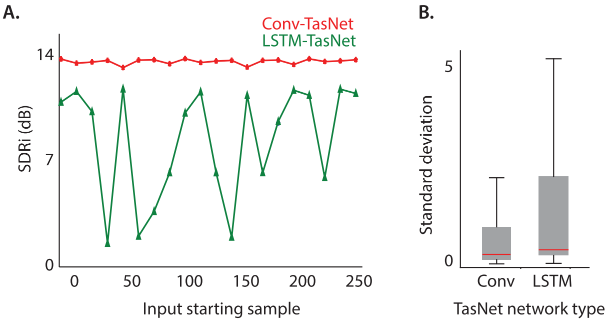

Fig. 4. (A) SDRi of an example mixture separated using LSTM-TasNet and causal Conv-TasNet as a function of the starting point in the mixture. The performance of Conv-TasNet is considerably more consistent and insensitive to the start point. (B) Standard deviation of SDRi across all the mixtures in the WSJ0-2mix test set with varying starting points.

For the CPU configuration, we tested the system with one processor on an Intel Core i7-5820K CPU. For the GPU configuration, we preloaded both the systems and the data to a Nvidia Titan Xp GPU. LSTM-TasNet with CPU configuration has a TPF close to its frame length (5 ms), which is only marginally acceptable in applications where only a slower CPU is available. Moreover, the processing in LSTM-TasNet is done sequentially, which means that the processing of each time frame must wait for the completion of the previous time frame, further increasing the total processing time of the entire utterance. Since Conv-TasNet decouples the processing of consecutive frames, the processing of subsequent frames does not have to wait until the completion of the current frame and allows the possibility of parallel computing. This process leads to a TPF that is 5 times smaller than the frame length (2 ms) in our CPU configuration. Therefore, even with slower CPUs, Conv-TasNet can still perform real-time separation.

F. *Sensitivity of LSTM-TasNet to the Mixture Starting Point*

Unlike language processing tasks where sentences have determined starting words, it is difficult to define a general starting sample or frame for speech separation and enhancement tasks. A robust audio processing system should therefore be insensitive to the starting point of the mixture. However, we empirically found that the performance of the causal LSTM-TasNet is very sensitive to the exact starting point of the mixture, which means that shifting the input mixture by several samples may adversely affect the separation accuracy. We systematically examined the robustness of LSTM-TasNet and causal Conv-TasNet to the starting point of the mixture by evaluating the separation accuracy for each mixture in the WSJ0-2mix test set with different sample shifts of the input. A shift of s samples corresponds to starting the separation at sample s instead of the first sample. Fig. 4(A) shows the performance of both systems on the same example mixture with different values of input shift. We observe that, unlike LSTM-TasNet, the causal Conv-TasNet performs consistently well for all shift values of the input mixture. We further tested the overall robustness for the entire test set by calculating the standard deviation of SDRi in each mixture with shifted mixture inputs similar to Fig. 4(A). The box plots of all the mixtures in the WSJ0-2mix test set in Fig. 4(B) show that causal Conv-TasNet performs consistently better across the entire test set, which confirms the robustness of Conv-TasNet to variations in the starting point of the mixture. One explanation for this inconsistency may be due to the sequential processing constraint in LSTM-TasNet which means that failures in previous frames can accumulate and affect the separation performance in all following frames, while the decoupled processing of consecutive frames in Conv-TasNet alleviates the effect of occasional error.

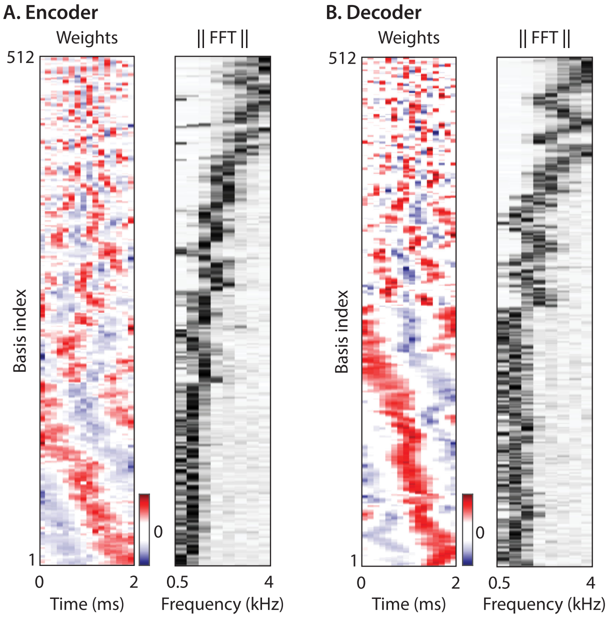

Fig. 5. Visualization of encoder and decoder basis functions and the magnitudes of their FFTs. The basis functions are sorted based on their pairwise Euclidean similarity.

G. *Properties of the Basis Functions*

One of the motivations for replacing the STFT representation of the mixture signal with the convolutional encoder in TasNet was to construct a representation of the audio that is optimized for speech separation. To shed light on the properties of the encoder and decoder representations, we examine the basis functions of the encoder and decoder (rows of the matrices $\mathbf{U}$ and $\mathbf{V}$). The basis functions are shown in Fig. 5 for the best noncausal Conv-TasNet, sorted in the same way as Fig. 2. The magnitudes of the FFTs for each filter are also shown in the same order. As seen in the figure, the majority of the filters are tuned to lower frequencies. In addition, it shows that filters with the same frequency tuning express various phase values for that frequency. This observation can be seen by the circular shift of the low-frequency basis functions. This result suggests an important role for low-frequency features of speech such as pitch as well as explicit encoding of the phase information to achieve superior speech separation performance.

V. Discussion

In this paper, we introduced the fully-convolutional time-domain audio separation network (Conv-TasNet), a deep learning framework for time-domain speech separation. This framework addresses the shortcomings of speech separation in the

STFT domain, including the decoupling of phase and magnitude, the suboptimal representation of the mixture audio for separation, and the high latency of calculating the STFT. The improvements are accomplished by replacing the STFT with a convolutional encoder-decoder architecture. The separation in Conv-TasNet is done using a temporal convolutional network (TCN) architecture together with a depthwise separable convolution operation to address the challenges of deep LSTM networks. Our evaluations showed that Conv-TasNet significantly outperforms STFT speech separation systems even when the ideal time-frequency masks for the target speakers are used. In addition, Conv-TasNet has a smaller model size and a shorter minimum latency, which makes it suitable for low-resource, low latency applications.

Unlike STFT which has a well-defined inverse transform that can perfectly reconstruct the input, best performance in the proposed model is achieved by an overcomplete linear convolutional encoder-decoder framework without guaranteeing the perfect reconstruction of the input. This observation motivates rethinking of autoencoder and overcompleteness in the source separation problem which may share similarities to the studies of overcomplete dictionary and sparse coding [51], [52]. Moreover, the analysis of the encoder/decoder basis functions in Section IV-G revealed two interesting properties. First, most of the filters are tuned to low acoustic frequencies (more than 60% tuned to frequencies below 1 kHz). This pattern of frequency representation, which we found using a data-driven method, roughly resembles the well-known mel-frequency scale [53] as well as the tonotopic organization of the frequencies in the mammalian auditory system [54], [55]. In addition, the overexpression of lower frequencies may indicate the importance of accurate pitch tracking in speech separation, similar to what has been reported in human multitalker perception studies [56]. In addition, we found that filters with the same frequency tuning explicitly express various phase information. In contrast, this information is implicit in the STFT operations, where the real and imaginary parts only represent symmetric (cosine) and asymmetric (sine) phases, respectively. This explicit encoding of signal phase values may be the key reason for the superior performance of TasNet over the STFT-based separation methods.

The combination of high accuracy, short latency, and small model size makes Conv-TasNet a suitable choice for both offline and real-time, low-latency speech processing applications such as embedded systems and wearable hearing and telecommunication devices. Conv-TasNet can also serve as a front-end module for tandem systems in other audio processing tasks, such as multitalker speech recognition [57]–[60] and speaker identification [61], [62]. On the other hand, several limitations of Conv-TasNet must be addressed before it can be actualized, including the long-term tracking of speakers and generalization to noisy and reverberant environments. Because Conv-TasNet uses a fixed temporal context length, the long-term tracking of an individual speaker may fail, particularly when there is a long pause in the mixture audio. In addition, the generalization of Conv-TasNet to noisy and reverberant conditions must be further tested [26], as time-domain approaches are more prone to temporal distortions which are particularly severe in reverberant acoustic environments. In such conditions, extending the Conv-TasNet framework to incorporate multiple input audio channels may prove advantageous when more than one microphone is available. Previous studies have shown the benefit of extending speech separation to multichannel inputs [63]–[65], particularly in adverse acoustic conditions and when the number of interfering speakers is large (e.g., more than 3).

In summary, Conv-TasNet represents a significant step toward the realization of speech separation algorithms and opens many future research directions that would further improve its accuracy, speed, and computational cost, which could eventually make automatic speech separation a common and necessary feature of every speech processing technology designed for real-world applications.

REFERENCES

[1] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 1, pp. 1702–1726, Oct. 2018.

[2] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Speech enhancement based on deep denoising autoencoder." in *Proc. Interspeech*, 2013, pp. 436–440.

[3] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Process. Lett.*, vol. 21, no. 1, pp. 65–68, Jan. 2014.

[4] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 1, pp. 7–19, Jan. 2015.

[5] Y. Isik, J. Le Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-channel multi-speaker separation using deep clustering," in *Proc. Interspeech*, 2016, pp. 545–549.

[6] 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 Int. Conf. Acoust., Speech, Signal Process.*, 2017, pp. 241–245.

[7] M. Kolbæk, D. Yu, Z.-H. Tan, and J. Jensen, "Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 10, pp. 1901–1913, Oct. 2017.

[8] Z. Chen, Y. Luo, and N. Mesgarani, "Deep attractor network for single-microphone speaker separation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2017, pp. 246–250.

[9] Y. Luo, Z. Chen, and N. Mesgarani, "Speaker-independent speech separation with deep attractor network," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 4, pp. 787–796, Apr. 2018. [Online]. Available: http://dx.doi.org/10.1109/TASLP.2018.2795749

[10] Z.-Q. Wang, J. Le Roux, and J. R. Hershey, "Alternative objective functions for deep clustering," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 686–690.

[11] Z.-Q. Wang, J. L. Roux, D. Wang, and J. R. Hershey, "End-to-end speech separation with unfolded iterative phase reconstruction," *Interspeech*, pp. 2708–2712, 2017.

[12] C. Li, L. Zhu, S. Xu, P. Gao, and B. Xu, "CBLDNN-based speaker-independent speech separation via generative adversarial training," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 711–715.

[13] D. Griffin and J. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-32, no. 2, pp. 236–243, Apr. 1984.

[14] J. Le Roux, N. Ono, and S. Sagayama, "Explicit consistency constraints for STFT spectrograms and their application to phase reconstruction." in *Proc. INTERSPEECH*, 2008, pp. 23–28.

[15] Y. Luo, Z. Chen, J. R. Hershey, J. Le Roux, and N. Mesgarani, "Deep clustering and conventional networks for music separation: Stronger together," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2017, pp. 61–65.

[16] A. Jansson, E. Humphrey, N. Montecchio, R. Bittner, A. Kumar, and T. Weyde, "Singing voice separation with deep u-net convolutional networks," in *Proc. 18th Int. Soc. Music Inf. Retrieval Conf.*, 2017, pp. 23–27.

[17] S. Choi, A. Cichocki, H.-M. Park, and S.-Y. Lee, "Blind source separation and independent component analysis: A review," *Neural Inf. Process.—Lett. Rev.*, vol. 6, no. 1, pp. 1–57, 2005.

[18] K. Yoshii, R. Tomioka, D. Mochihashi, and M. Goto, "Beyond NMF: Time-domain audio source separation without phase reconstruction," in *Proc. Int. Soc. Music Inf. Retrieval*, 2013, pp. 369–374.

[19] S. Venkataramani, J. Casebeer, and P. Smaragdis, "End-to-end source separation with adaptive front-ends," in *Proc. IEEE 52nd Asilomar Conf. Signals, Syst., Comput.*, 2018, pp. 684–688.

[20] D. Stoller, S. Ewert, and S. Dixon, "Wave-u-net: A multi-scale neural network for end-to-end audio source separation," in *Proc. Int. Soc. Music Inf. Retrieval*, 2018, pp. 334–340.

[21] Y. Luo and N. Mesgarani, "TasNet: Time-domain audio separation network for real-time, single-channel speech separation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 696–700.

[22] S.-W. Fu, T.-W. Wang, Y. Tsao, X. Lu, and H. Kawai, "End-to-end waveform utterance enhancement for direct evaluation metrics optimization by fully convolutional neural networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 9, pp. 1570–1584, Sep. 2018.

[23] S. Pascual, A. Bonafonte, and J. Serrà, "SEGAN: Speech enhancement generative adversarial network," in *Proc. Interspeech*, 2017, pp. 3642–3646.

[24] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assisted Intervention*, 2015, pp. 234–241.

[25] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2016, pp. 31–35.

[26] Y. Luo and N. Mesgarani, "Real-time single-channel dereverberation and separation with time-domain audio separation network," in *Proc. Interspeech*, 2018, pp. 342–346.

[27] F.-Y. Wang, C.-Y. Chi, T.-H. Chan, and Y. Wang, "Nonnegative least-correlated component analysis for separation of dependent sources by volume maximization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 875–888, May 2010.

[28] C. H. Ding, T. Li, and M. I. Jordan, "Convex and semi-nonnegative matrix factorizations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 1, pp. 45–55, Jan. 2010.

[29] C. Lea, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks: A unified approach to action segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 47–54.

[30] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 156–165.

[31] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, arXiv:1803.01271.

[32] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1251–1258.

[33] A. G. Howard *et al.*, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," 2017, arXiv:1704.04861.

[34] D. Wang, "On ideal binary mask as the computational goal of auditory scene analysis," in *Speech Separation by Humans and Machines*. Boston, MA, USA: Springer, 2005, pp. 181–197.

[35] Y. Li and D. Wang, "On the optimality of ideal binary time–frequency masks," *Speech Commun.*, vol. 51, no. 3, pp. 230–239, 2009.

[36] Y. Wang, A. Narayanan, and D. Wang, "On training targets for supervised speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 1849–1858, Dec. 2014.

[37] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 708–712.

[38] A. Van Den Oord *et al.*, "Wavenet: A generative model for raw audio," in *Proc. 9th ISCA Speech Syn. Workshop*, 2016, p. 125.

[39] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1251–1258.

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1026–1034.

[41] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," 2016, arXiv:1607.06450.

[42] "Script to generate the multi-speaker dataset using wsj0," 2016. [Online]. Available: http://www.merl.com/demos/deep-clustering

[43] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Lear. Represent.*, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[44] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 4, pp. 1462–1469, Jul. 2006.

[45] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)—A new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2001, vol. 2, pp. 749–752.

[46] *Vocabulary for Performance and Quality of Service*, International Telecommunication Union (ITU), Geneva, Switzerland, ITU-T Rec. P.10, 2006.

[47] R. R. Sokal, "A statistical method for evaluating systematic relationship," *Univ. Kansas Sci. Bull.*, vol. 28, pp. 1409–1438, 1958.

[48] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 4510–4520.

[49] "Audio samples for Conv-TasNet," 2018. [Online]. Available: http://naplab.ee.columbia.edu/tasnet.html

[50] C. Xu, X. Xiao, and H. Li, "Single channel speech separation with constrained utterance level permutation invariant training using grid LSTM," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 6–10.

[51] T.-W. Lee, M. S. Lewicki, M. Girolami, and T. J. Sejnowski, "Blind source separation of more sources than mixtures using overcomplete representations," *IEEE Signal Process. Lett.*, vol. 6, no. 4, pp. 87–90, Apr. 1999.

[52] M. Zibulevsky and B. A. Pearlmutter, "Blind source separation by sparse decomposition in a signal dictionary," *Neural Comput.*, vol. 13, no. 4, pp. 863–882, 2001.

[53] S. Imai, "Cepstral analysis synthesis on the mel frequency scale," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1983, vol. 8, pp. 93–96.

[54] G. L. Romani, S. J. Williamson, and L. Kaufman, "Tonotopic organization of the human auditory cortex," *Science*, vol. 216, no. 4552, pp. 1339–1340, 1982.

[55] C. Pantev, M. Hoke, B. Lutkenhoner, and K. Lehnertz, "Tonotopic organization of the auditory cortex: Pitch versus frequency representation," *Science*, vol. 246, no. 4929, pp. 486–488, 1989.

[56] C. J. Darwin, D. S. Brungart, and B. D. Simpson, "Effects of fundamental frequency and vocal-tract length changes on attention to one of two simultaneous talkers," *J. Acoustical Soc. Amer.*, vol. 114, no. 5, pp. 2913–2922, 2003.

[57] J. R. Hershey, S. J. Rennie, P. A. Olsen, and T. T. Kristjansson, "Super-human multi-talker speech recognition: A graphical modeling approach," *Comput. Speech Lang.*, vol. 24, no. 1, pp. 45–66, 2010.

[58] C. Weng, D. Yu, M. L. Seltzer, and J. Droppo, "Deep neural networks for single-channel multi-talker speech recognition," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 10, pp. 1670–1679, Oct. 2015.

[59] Y. Qian, X. Chang, and D. Yu, "Single-channel multi-talker speech recognition with permutation invariant training," *Speech Commun.*, vol. 104, pp. 1–11, 2018.

[60] K. Ochi, N. Ono, S. Miyabe, and S. Makino, "Multi-talker speech recognition based on blind source separation with ad hoc microphone array using smartphones and cloud storage," in *Proc. INTERSPEECH*, 2016, pp. 3369–3373.

[61] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 1695–1699.

[62] M. McLaren, Y. Lei, and L. Ferrer, "Advances in deep neural network approaches to speaker recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 4814–4818.

[63] S. Gannot *et al.*, "A consolidated perspective on multimicrophone speech enhancement and source separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 4, pp. 692–730, Apr. 2017.

[64] Z. Chen *et al.*, "Cracking the cocktail party problem by multi-beam deep attractor network," in *Proc. IEEE Autom. Speech Recognit. Understanding Workshop*, 2017, pp. 437–444.

[65] 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 Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 1–5.

Authors' photographs and biographies not available at the time of publication.