

Audio Source Separation

Shadi sartipi

Narges Mohammadi

Nicholas Boldt



Outline

- Audio Source Separation problem
- Practical Application
- Challenges
- Source Separation using Position

Models

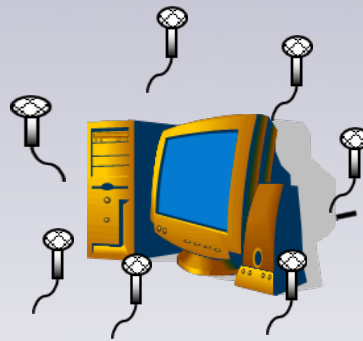
- DNN_based Approaches:
 - MULTI-SCALE MULTI-BAND
DENSENETS FOR AUDIO SOURCE
SEPARATION
 - End-to-end music source separation



Audio Source Separation

Why it is needed?

- Remix the balance
- Make the vocals louder
- Suppress unwanted sound
- Change the spatial location



Audio Source Separation

- “Cocktail party effect”
 - E. C. Cherry, 1953.
 - Ability to concentrate attention on a specific sound source from within a mixture.
 - Even when interfering energy is close to energy of desired source.
- “Prince Shotoku Challenge”
 - Legendary Japanese prince Shotoku (6th Century AD) could listen and understand simultaneously the petitions by ten people.
 - Concentrate attention on several sources at the same time!

Both allegories imply an extra step of semantic understanding of the sources, beyond mere acoustical isolation.

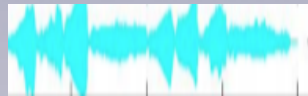


Audio Source Separation

Separating out the individual sounds in an audio mixture

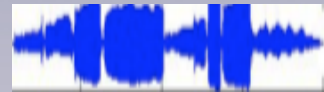


Source 1

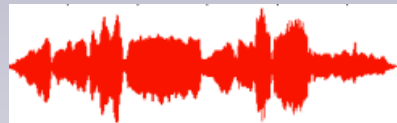


+

Source 2



Mixture



Source Separation

Estimate 1



Estimate 2

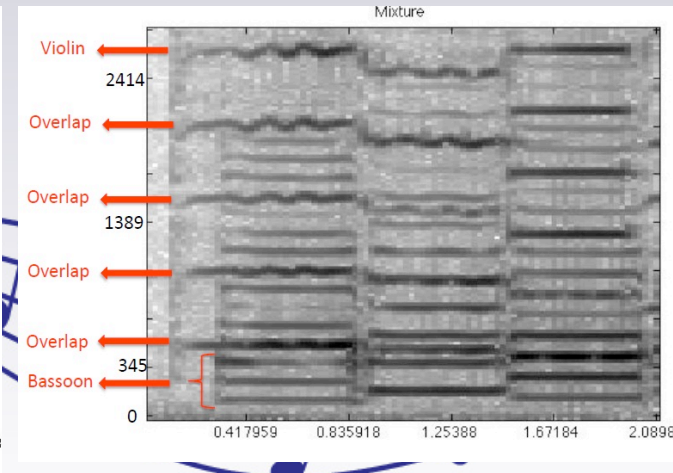
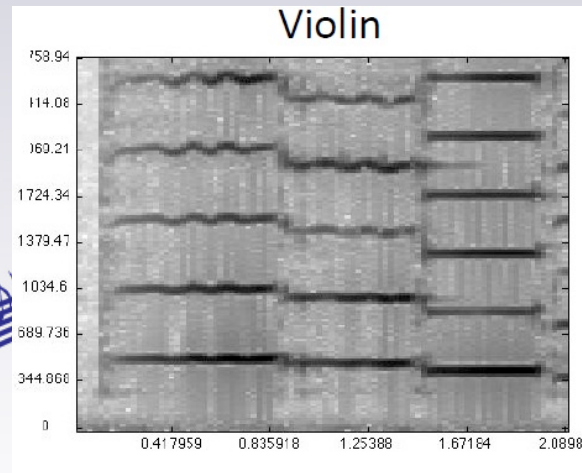
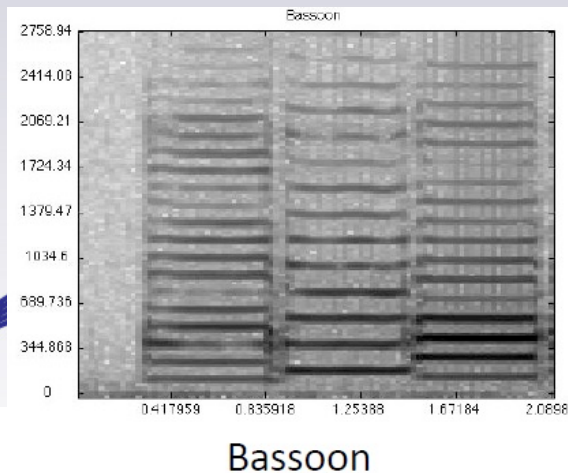
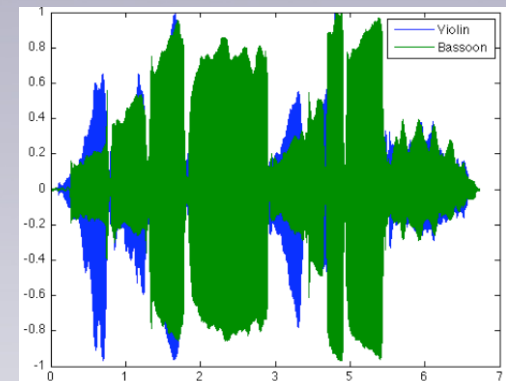
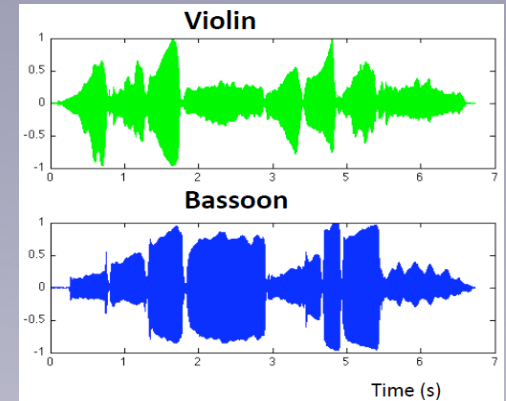


Practical Applications

- Hearing Aids
- Automated transcription of speech and music
- Automated sound source identification
- Speech recognition system

Challenges

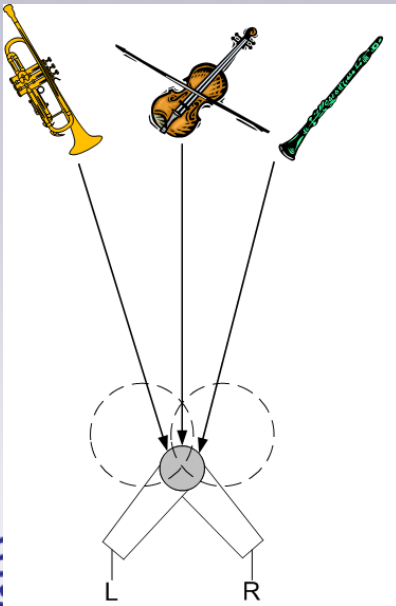
- Might present many musical instrument and voice in the mixture
- Processed with addition of filters and reverberation
- Time-varying mixture



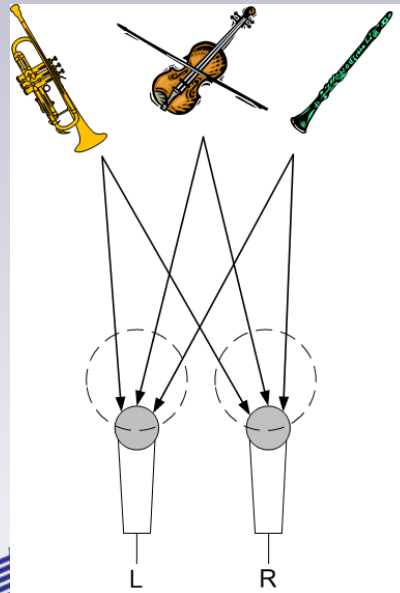
Interesting Questions

- How do humans separate sounds?
- What cues in the sound are important to separate one sound from background noise?

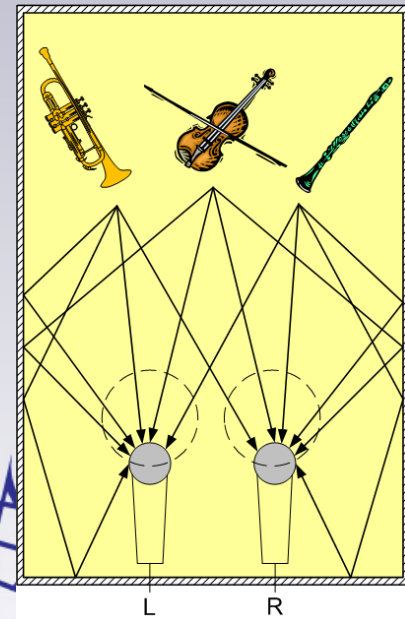
XY Stereo



AB Stereo



Reverberant environment

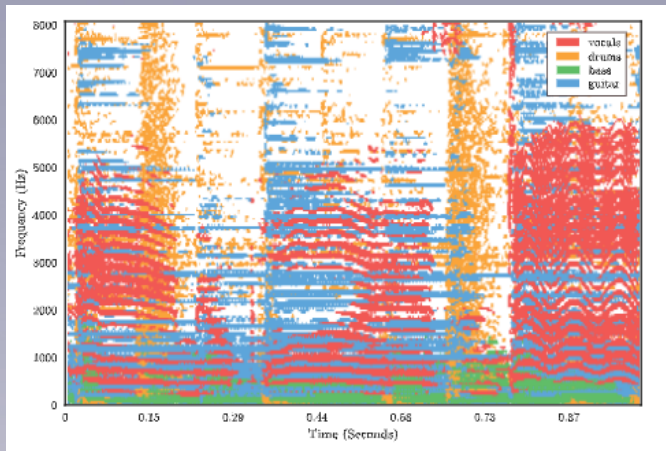


Musical Source Separation

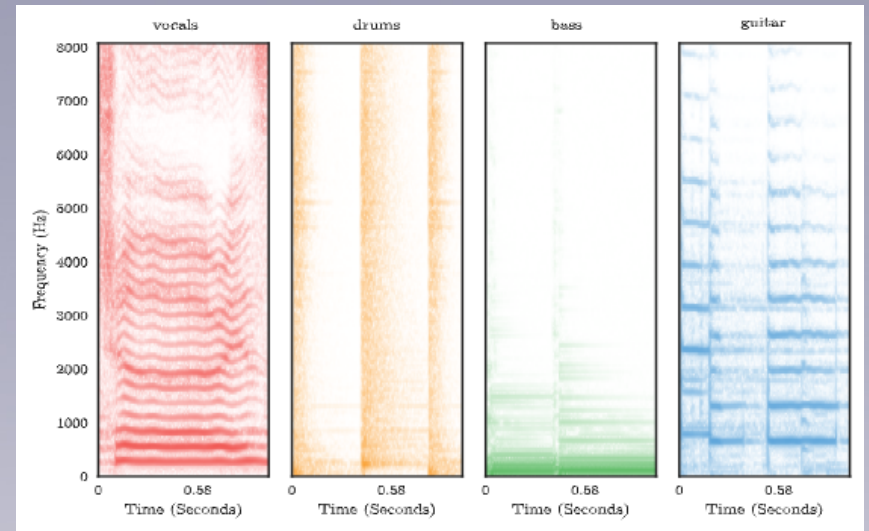
Characteristics of Music signals

- Music signals have distinct characteristics that clearly differentiate them from other types of audio signals
- All music separation problems start with the definition of the desired musical source to be separated, often referred to as the target source
- musical sources are often categorized as either predominantly harmonic, Predominantly percussive, or as singing voice
- notable property of musical sources is that they are typically sparse in the sense that for the majority of points in time and frequency, the sources have very little energy present.

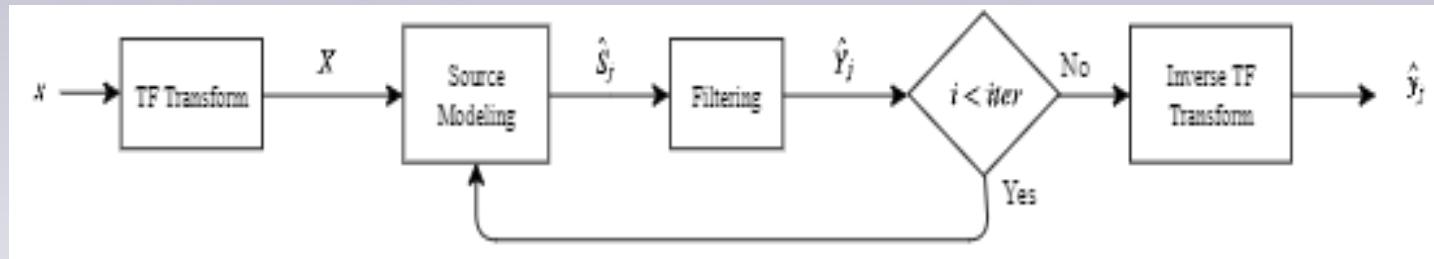




Representation of a music mixture in the time-frequency domain



Magnitude spectrogram of four music signals separately



One of the Common MSS work flow: source models are obtained from the spectrogram of the audio mix

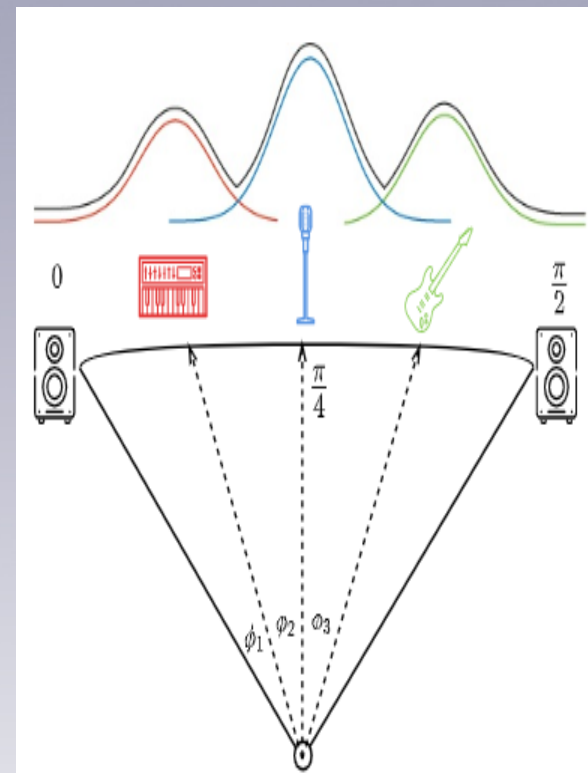


Musical Source Position Models

In the case of multichannel music signals, the spatial position of the sources has often been exploited to perform music source separation.

Several techniques using spatial position for separation:

- Independent Component Analysis (ICA) [1]
- DUET [2]
- ADress [3]
- PROJET [4]



[1] A. Hyvärinen, J. Karhunen, and E. Oja, Eds., Independent Component Analysis, Wiley and Sons, 2001.

[2] S. Rickard, The DUET blind source separation algorithm, pp. 217–241, Springer Netherlands, Dordrecht, 2007.

[3] D. Barry, B. Lawlor, and E. Coyle, “Real-time sound source separation using azimuth discrimination and resynthesis,” in 117th Audio Engineering Society (AES) Convention, San Francisco, CA, USA, 2004.

[4] D. FitzGerald, A. Liutkus, and R. Badeau, “Projection-based demixing of spatial audio,” IEEE/ACM Trans. on Audio, Speech, and Language Processing, vol. 24, no. 9, pp. 1560–1572, 2016

Overview of PROJET model for audio separation through projections

Signal Model:

$$x(f, t) = \sum_j y_j(f, t) \quad y_j(f, t) = h(\phi, \tau \mid f) s_j(f, t)$$

Inputs:

- Location set $L = (\phi_1, \tau_1), \dots, (\phi_L, \tau_L) \in \mathbb{L}^L$
- Projection set $P = (\phi_1, \tau_1), \dots, (\phi_M, \tau_M) \in \mathbb{L}^M$
- Mixture x
- No. Of iterations, parameter α and divergence to use

Initialization:

- Initialize parameters randomly:
- Parameter fitting: for each object j :
 - Update α -PSD
 - Update pan-delay coefficient

Separation; for each object j :

- Estimate $M+1$ projected images $y_j^c(f, t)$ through:

$$\hat{y}_{m,j}^c(f, t) = \frac{P_j^\alpha k_m(f)^T Q_j}{\sum_{j'} P_{j'}^\alpha k_m(f)^T Q_{j'}} c_m(f, t)$$

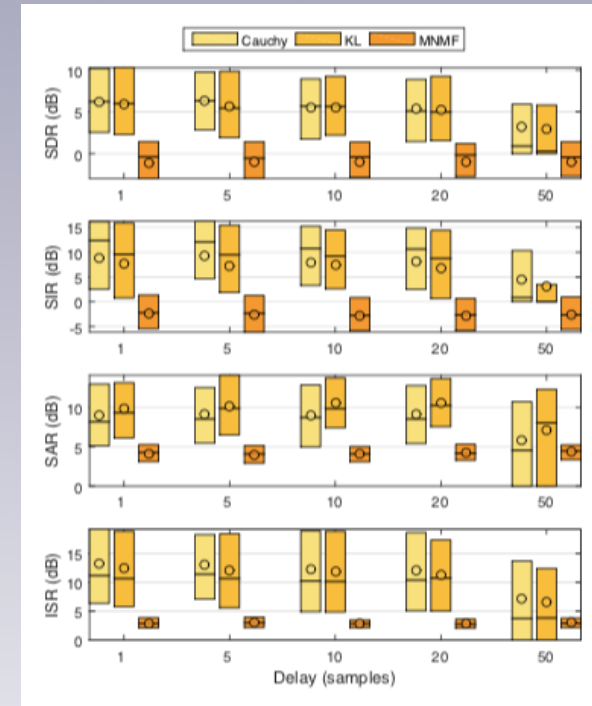
- Estimate object image \hat{y}_j through using pseudo inverse of N_f : $\hat{y}_j(f, t) = N_f^\dagger y_j^c(f, t)$
- Apply ISTFT to \hat{y}_j to recover waveforms



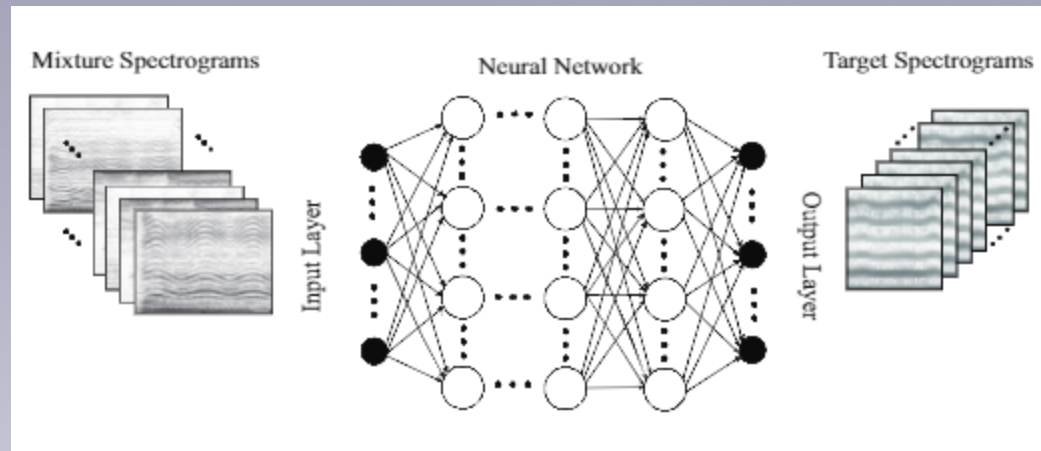
Simulation Setup of Project Method:

Dataset: open-source MSD100 dataset

- Dev_set: consist of 50 full length songs created from mixtures of 4 objects
- Test_set:
 - extract 30s of each song to create stereo image using pan-delay model
 - Create mixture signal x by summing stereo images
 - Two distinct set of mixtures to evaluate the effects of panning and delay separately
 - In the first set, objects were mixed with an equal angle between them, with no delay between the channels (angle between objects was varied from 10 degrees to 30 degrees in steps of 10)—>total of 150 test mixtures, with a total of 600 separated sources
 - In the second set of mixtures, the angle between the sources was fixed at a value of 20 degrees, and the delay between left and right channels for each of the 4 sources was varied
- Separation evaluation criteria: Signal to Distortion Ratio (SDR), Signal to Interference Ratio (SIR) and Image to Spatial Distortion Ratio (ISR)



Deep Neural Network (DNN) Model



- In contrast to the mentioned approaches which require explicit models of the source for processing, methods based on **DNNs** take advantage of optimization techniques to train source models in a supervised manner
- Regardless of the inputs and targets used, DNN methods work by training the parameters of non-linear functions to minimize the reconstruction error of the chosen outputs (spectrograms or masks) based on the inputs (audio mixes).



How DNNs have been used?

Earliest DNN_based approaches: consist of fully connected networks (FCN);

Drawback: large number of parameters, restricted the use of temporal context in such networks to less than one second

Recurrent neural networks (RNNs): they apply their weights recursively over an input sequence, and can process sequential data of arbitrary length

Limitations of DNNs:

- Need for large amount of training data
- The cost function results (like MSE) results in a well-behaved stochastic gradient optimization problem, it also poorly correlates with perceived audio quality.



Two frameworks of applying DNN in separation



MULTI-SCALE MULTI-BAND DENSENETS FOR AUDIO SOURCE SEPARATION

Takahashi, Naoya, and Yuki Mitsufuji. "Multi-scale multi-band densenets for audio source separation." *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2017.



Background:

- The idea of DenseNet is to use concatenation of output feature maps of preceding layers as the input to succeeding layers.
- This iterative connection enables the network to learn explicit cross-layer interactions
- Efficient use of parameters which suits the audio source separation problem very well.



Motivation:

- DenseNet is inherently memory demanding
- In audio source separation, both the input and output dimension would be far larger (e.g. 1024 frequency bins \times 128 frames) in order to utilize sufficiently long contexts with high frequency resolution

Solution:

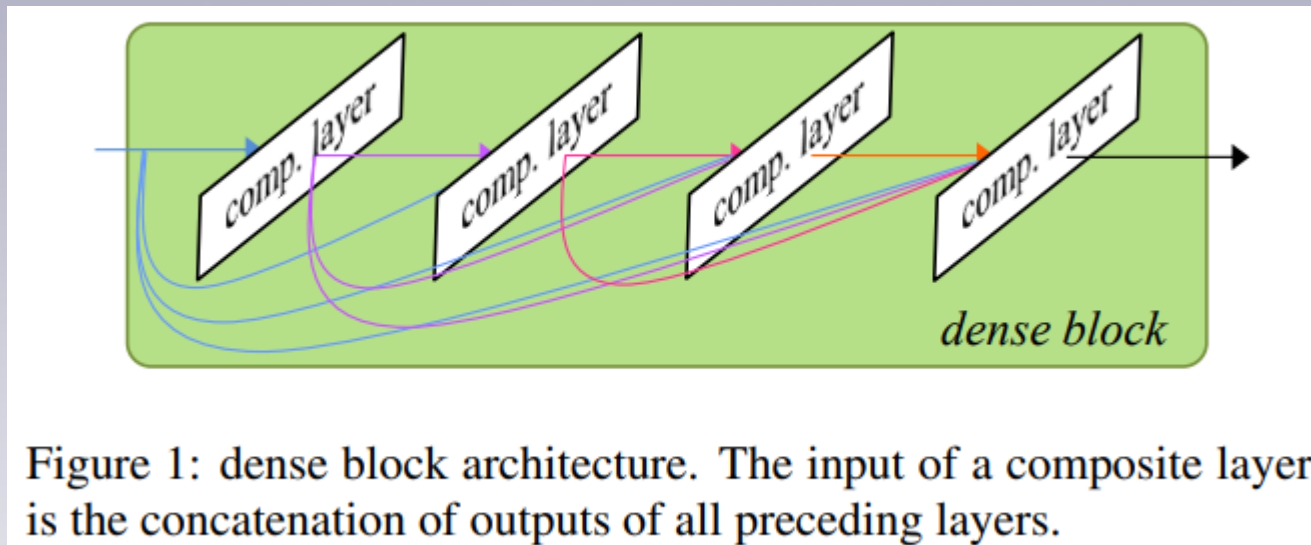
- A fully convolutional multi-scale DenseNet equipping dense blocks with multiple resolutions was proposed



DENSENETS

- DenseNet further improves the information flow between layers by replacing the simple addition of the output of a single preceding layer with a concatenation of all preceding layers:

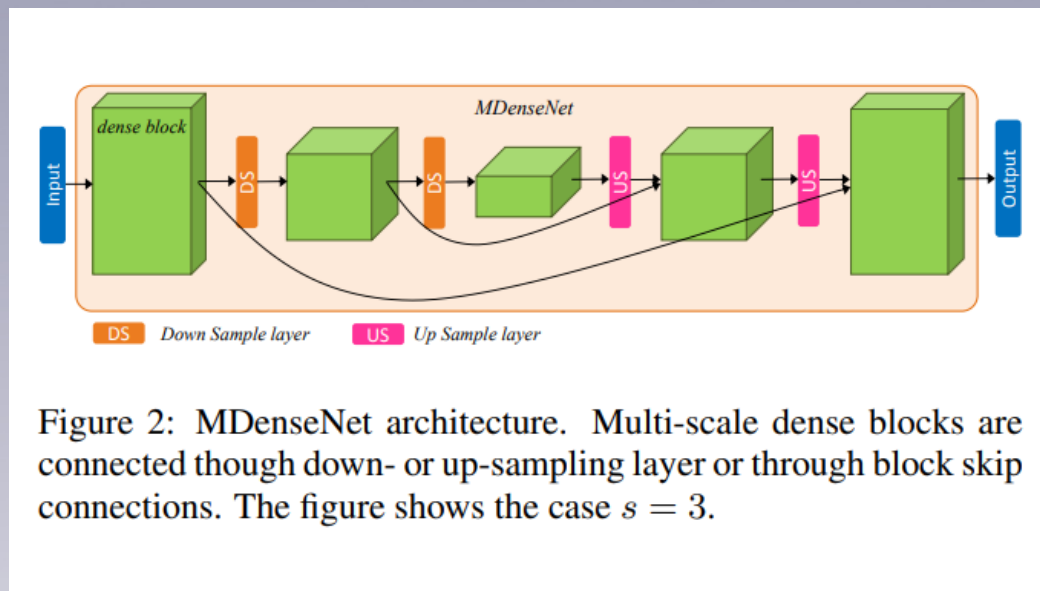
$$x_l = H_l([x_{l-1}, x_{l-2}, \dots, x_0]),$$



Limitation : Memory demanding



Multi-Scale DENSENET (MDENSENET)

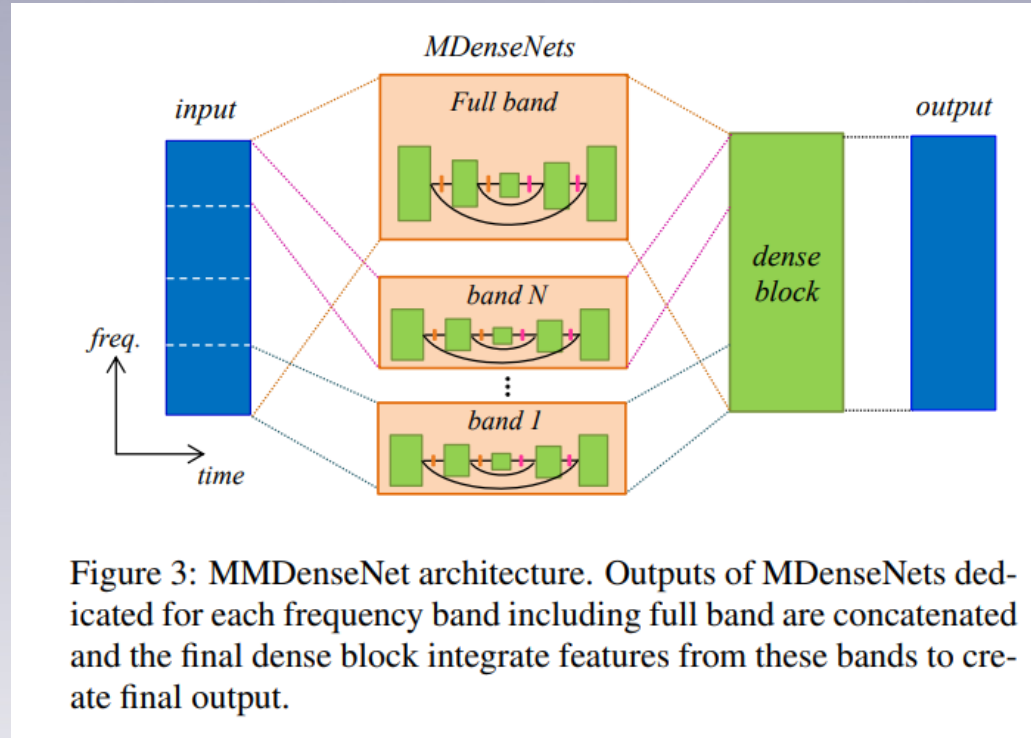


- Note that the proposed architecture is fully convolutional and thus can be applied to arbitrary input length.



Multi-band MDENSENET

- Splitting the input into multiple bands and apply multiscale DenseNet to each band



Architecture Details

One advantage of MMDenseNet is **designing suitable architectures for each band individually** and assign computational resources according to the importance of each band which may differ depending on the target source or application

Table 1: *The proposed architectures. All dense blocks are equipped with 3×3 kernels with L layers and k growth rate. The pooling size and transposed convolution kernel size are 2×2 .*

Layer	scale	MMDenseNet			MDenseNet
		low	high	full	
band split		first half	last half	-	-
conv ($t \times f, ch$)	1	$3 \times 4, 32$	$3 \times 3, 32$	$3 \times 4, 32$	$3 \times 4, 32$
dense 1 (k, L)		14, 4	10, 3	6, 2	12, 4
down sample	$\frac{1}{2}$	pool	pool	pool	pool
dense 2 (k, L)		16, 4	10, 3	6, 2	12, 4
down sample	$\frac{1}{4}$	pool	pool	pool	pool
dense 3 (k, L)		16, 4	10, 3	6, 2	12, 4
down sample	$\frac{1}{8}$	pool	pool	pool	pool
dense 4 (k, L)		16, 4	10, 3	6, 4	12, 4
up sample	$\frac{1}{4}$	t.conv	t.conv	t.conv	t.conv
concat.		low dense 3	high dense 3	full dense 3	dense 3
dense 5 (k, L)		16, 4	10, 3	6, 2	12, 4
up sample	$\frac{1}{2}$	t.conv	t.conv	t.conv	t.conv
concat.		low dense 2	high dense 2	full dense 2	dense 2
dense 6 (k, L)		16, 4	10, 3	6, 2	12, 4
up sample	1	t.conv	t.conv	t.conv	t.conv
concat.		low dense 1	high dense 1	full dense 1	dense 1
dense 7 (k, L)		16, 4	10, 3	6, 2	12, 4
concat. (axis)	1	freq			-
concat. (axis)		channel			-
dense 8 (k, L)		4, 2			4, 2
conv($t \times f, ch$)		$1 \times 2, 2$			$1 \times 2, 2$



Results on the SiSEC 2016 DSD100 dataset

Fig. 4 show the signal to distortion ratio (SDR) computed using the BSS Eval toolbox

Table 2: *Comparison of SDR.*

Method	SDR in dB				
	Bass	Drums	Other	Vocals	Acco.
DeepNMF [4]	1.88	2.11	2.64	2.75	8.90
NUG [10]	2.72	3.89	3.18	4.55	10.29
FNN [11]	2.54	3.75	2.92	4.47	11.12
BLSTM [12]	2.89	4.00	3.24	4.86	11.26
BLEND [12]	2.98	4.13	3.52	5.23	11.70
MDenseNet	2.74	4.37	3.33	4.91	11.21
MMDenseNet	3.91	5.37	3.81	6.00	12.10
MMDenseNet+	4.13	5.19	4.37	6.06	12.66



Conclusion

- ❑ multi-scale dense block enables the network to model the signal on different scales, i.e. the global context in the downscaled blocks and local fine-grained structure in the high resolution blocks.
- ❑ dense blocks at multiple scales connected through down-sampling and up-sampling layers
- ❑ a multi-band DenseNet to enable kernels in convolution layer to learn more effectively



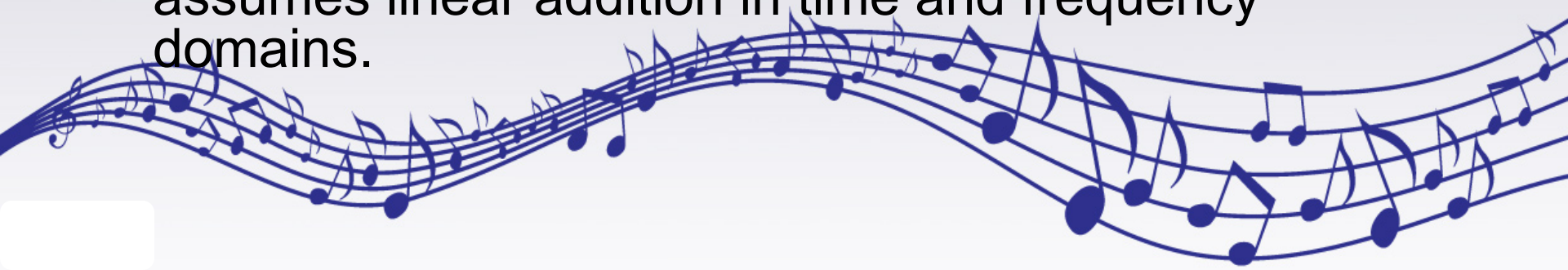
End-to-end music source separation

Lluís, F., Pons, J., & Serra, X. (2019). End-to-End Music Source Separation: Is it Possible in the Waveform Domain? *Interspeech 2019*. doi: 10.21437/interspeech.2019-1177



Background

- Most successful deep learning algorithms use the magnitude spectrogram as input, which omits information about the signal – phase.
- Waveform representations contain all the information in the signal, however:
 - Phase is unpredictable – unlikely to see identical waveforms produced by a source
 - Hence, one either needs a large number of bases or shift-invariant bases for accurate decompositions.
- However using magnitude spectrogram (as opposed to power spectrograms) without phase wrongly assumes linear addition in time and frequency domains.



Questions

- Does discarding phase result in loss of information?
- Are there artifacts from introducing phase into the mixture at synthesis time?
- Since magnitude spectrograms are different from power spectrograms, and are not additive, what is the effect of relying on an incorrect model?



Models

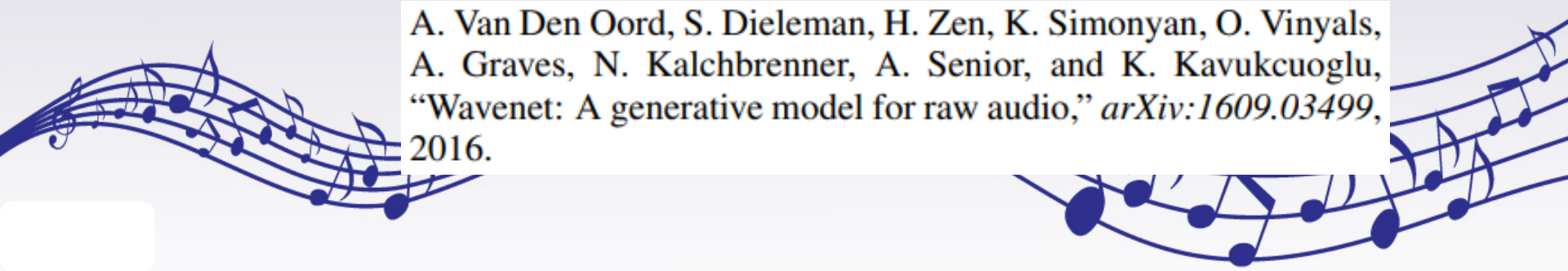
1. Wavenet-based model – new model for music source separation
2. DeepConvSep – spectrogram-based deep learning model for multi-instrument separation
3. Wave-U-Net – waveform based model for voice separation



Wavenet

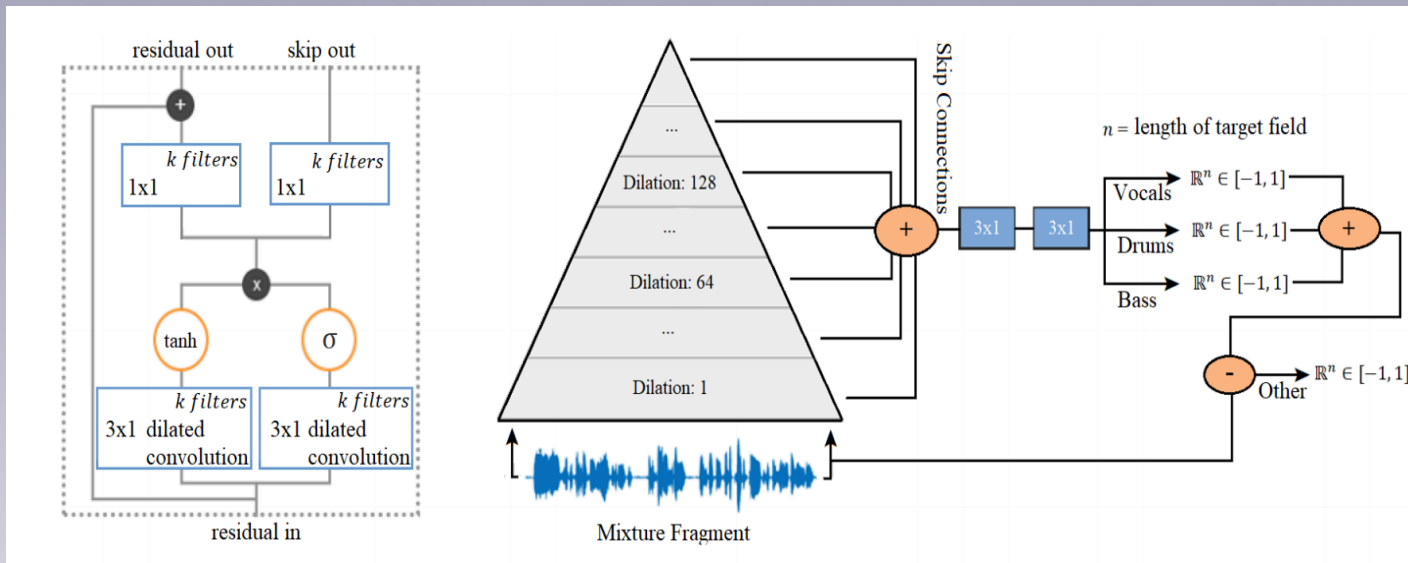
- Generative model operating directly on raw audio waveform
- Joint probability of waveform $\mathbf{x} = \{x_1, x_2, \dots, x_T\}$
 - Each audio sample conditional on samples at all previous timesteps
- Conditional probability distribution modelled by stack of causal convolutional layers – model cannot violate ordering in which data modeled
- Model outputs categorical distribution over next value x_t with softmax layer, and optimized to maximize log-likelihood.

$$p(\mathbf{x}) = \prod_{t=1}^T p(x_t | x_1, \dots, x_{t-1})$$

A decorative graphic of blue musical notes and staves is positioned at the bottom of the slide, spanning across the width of the content area.

A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, “Wavenet: A generative model for raw audio,” *arXiv:1609.03499*, 2016.

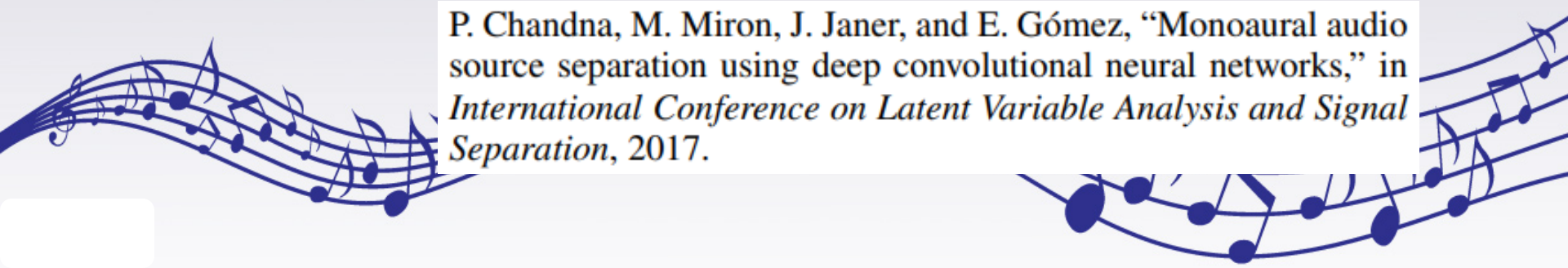
Wavenet-based Model



- Removes causality requirement of original model
- Directly regress waveform sources instead of sampling from softmax output
- Output:
 - 3 outputs for multi-instrument
 - 1 output for voice separation

DeepConvSep Model

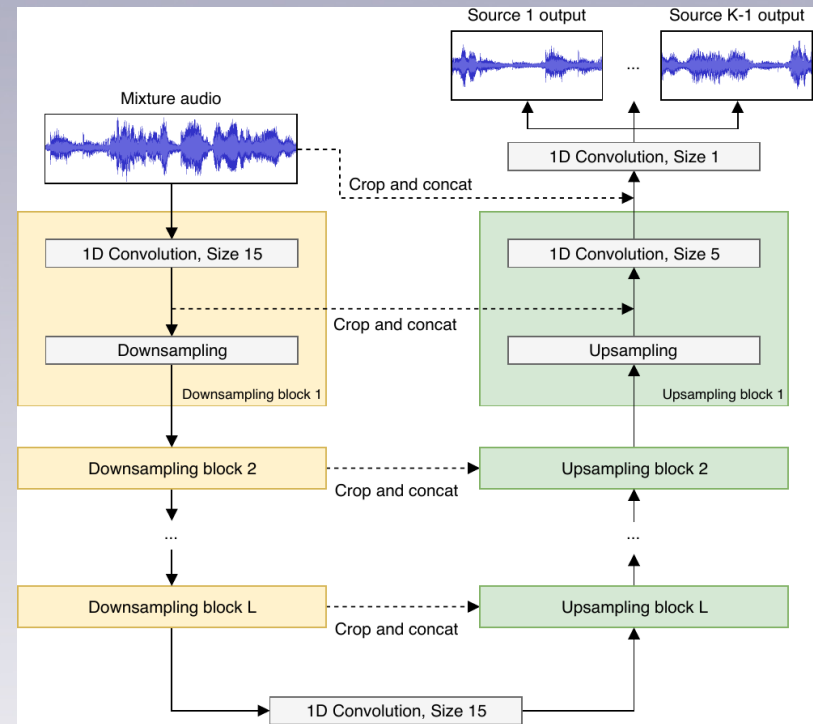
- Based on convolutional encoder-decoder
- Encoder:
 - CNN layer with 50 vertical filters for timbral representation
 - second CNN layer with 30 horizontal filters for modeling temporal cues
 - dense layer with 128 units
- Decoder:
 - two deconvolutional layers which up-sample bottleneck feature maps up to input size, corresponding to estimated masks

A decorative graphic at the bottom of the slide featuring blue musical notes and staff lines that curve across the width of the slide.

P. Chandna, M. Miron, J. Janer, and E. Gómez, “Monoaural audio source separation using deep convolutional neural networks,” in *International Conference on Latent Variable Analysis and Signal Separation*, 2017.

Wave-U-Net for Voice Separation

- Encoder-decoder architecture
- Encoder (12 layer) down-samples feature maps, decoder (12 layers) up-samples feature maps to have required output-length
- All decoder layers can access feature maps from the encoder at the same level



D. Stoller, S. Ewert, and S. Dixon, "Wave-u-net: A multi-scale neural network for end-to-end audio source separation," *ISMIR*, 2018.

Results: Multi-instrument Separation

- Wavenet: Wider vs. Deeper
- Model: Wavenet vs. DeepConvSep
- Participants did not show any preference for the vocals' separations
- Although Wavenet-based models seem to better remove the accompanying sources, they introduce noticeable artifacts

Table 2: Multi-instrument source separation median scores.

Wavenet-based	Vocals			Drums		
	SDR	SIR	SAR	SDR	SIR	SAR
1 stack	0.35	3.94	4.38	1.24	7.98	3.56
2 stacks	0.07	4.48	3.49	-0.09	6.87	2.88
3 stacks	3.46	11.26	5.18	4.39	13.37	5.08
4 stacks	3.35	11.25	5.24	4.13	13.23	5.00
5 stacks	2.84	9.56	5.20	4.60	12.66	6.08
4 stacks + \mathcal{L}_d	3.05	10.58	4.80	4.09	12.85	5.31
DeepConvSep 16kHz	2.38	4.45	8.39	3.19	6.69	6.58
DeepConvSep 44kHz	2.37	4.65	8.04	3.14	6.73	6.55

Wavenet-based	Bass			Other		
	SDR	SIR	SAR	SDR	SIR	SAR
1 stack	0.35	4.54	4.70	-2.70	-1.37	6.75
2 stacks	-0.55	0.87	7.80	-2.05	-0.97	8.96
3 stacks	2.24	6.36	5.94	0.54	4.07	4.41
4 stacks	2.49	6.53	5.77	0.41	3.83	4.47
5 stacks	2.48	6.70	6.27	0.18	3.26	4.75
4 stacks + \mathcal{L}_d	2.23	5.66	6.37	-0.19	4.37	3.24
DeepConvSep 16kHz	0.27	1.92	7.46	-2.02	1.74	2.50
DeepConvSep 44kHz	0.17	1.98	7.06	-2.13	1.84	2.33



Results: Voice Separation

- Deeper vs. Wider
- Data-sampling strategies (percent of training data containing singing voice) – compare to previous Table
- Wavenet-based model vs Wave-U-Net

Table 4: *Singing voice source separation median scores.*

<i>Wavenet-based</i>	<i>Vocals</i>			<i>Accompaniment</i>		
	SDR	SIR	SAR	SDR	SIR	SAR
1 stack	2.76	10.11	4.78	9.73	12.73	13.77
2 stacks	3.05	11.13	4.50	10.13	13.82	12.93
3 stacks	3.62	12.33	4.96	10.41	13.97	13.53
4 stacks	3.67	12.14	5.24	10.64	14.43	13.22
5 stacks	3.02	12.44	4.44	10.42	13.89	13.30
4 stacks+ \mathcal{L}_d	3.78	11.76	5.44	10.90	14.26	13.84
4 stacks+ $\mathcal{L}_d+25\%$	3.98	12.20	5.19	10.75	14.21	13.70
4 stacks+ $\mathcal{L}_d+50\%$	4.49	13.52	6.17	11.39	16.37	13.49
4 stacks+ $\mathcal{L}_d+75\%$	3.93	12.93	5.40	11.14	16.18	13.37
4 stacks+ $\mathcal{L}_d+100\%$	2.36	6.25	5.88	10.44	16.73	12.15
<i>Wave-U-Net</i>	4.60	14.30	5.54	11.87	16.08	14.20

Table 5: *Singing voice source separation perceptual scores.*

MOS	<i>Wavenet-based</i>	<i>Wave-U-Net</i>
<i>Vocals</i>	3.0 ± 1.0	3.3 ± 0.85



Thank You!

