

# Improving singing voice separation using Deep U-Net and Wave-U-Net with data augmentation

Alice Cohen-Hadria  
UMR STMS 9912  
Sorbonne Université, IRCAM, CNRS,  
Paris, France  
cohenhadria@ircam.fr

Axel Roebel  
UMR STMS 9912  
IRCAM, Sorbonne Université, CNRS,  
Paris, France  
axel.roebel@ircam.fr

Geoffroy Peeters  
LTCI - Telecom Paris  
Institut Polytechnique de France  
France  
geoffroy.peeters@telecom-paris.fr

**Abstract**—State-of-the-art singing voice separation is based on deep learning making use of CNN structures with skip connections (like U-Net model, Wave-U-Net model, or MS-DENSELSTM). A key to the success of these models is the availability of a large amount of training data. In the following study, we are interested in singing voice separation for mono signals and will investigate into comparing the U-Net and the Wave-U-Net that are structurally similar, but work on different input representations. First, we report a few results on variations of the U-Net model. Second, we will discuss the potential of state of the art speech and music transformation algorithms for augmentation of existing data sets and demonstrate that the effect of these augmentations depends on the signal representations used by the model. The results demonstrate a considerable improvement due to the augmentation for both models. But pitch transposition is the most effective augmentation strategy for the U-Net model, while transposition, time stretching, and formant shifting have a much more balanced effect on the Wave-U-Net model. Finally, we compare the two models on the same dataset.

**Index Terms**—Singing voice separation, data augmentation, convolutional neural network

## I. INTRODUCTION

In the case of music, source separation aims at separating the various instruments (such as the singing voice, guitar, piano or drums) present in the mixture (the mix). When the source of interest is the singing voice, various assumptions can be made to help the separation, such as assuming a source/filter production mechanism [1], using the sparsity in frequency of the vocals in Robust Principal Component Analysis (rPCA) [2], assuming the non-repetition of the vocal parts over time [3] or using Non Negative Matrix factorization [4]. Those assumptions lead to a first set of approaches for singing voice separation which are unsupervised, called Blind Audio Source Separation.

Recently, because of the availability of new annotated training datasets, supervised approaches have taken the lead, especially using neural networks methods. The current state-of-the-art, and winner of SiSEC 2018 [5], is a combination Long-Short Time Memory networks and Dense Convolutional Neural Networks, presented in [6] and use stereo signals.

In the following we will discuss the problem of singing voice separation using mono signals. In this context, [7] relies on Convolutional Neural Networks (ConvNet). A more

sophisticated version of these, the deep U-Net (also called U-Net) architecture has been proposed in [8]. Both process the spectrogram. They therefore necessitate the frequency-to-time reconstruction of the audio signal, which potentially leads to artifacts. For this reason, Stoller proposed in [9] the Wave-U-Net model which directly processes and separates the audio signal. The comparison of these two models is of interest, since they share most of architecture properties, while processing very different inputs (temporal audio signals and spectrogram). The goal of this paper is to compare these two models and the implication of using either audio signals or spectrograms as input for source separation. The Wave-U-Net has been trained on a rather limited dataset, consisting of the train part of the musdb18 dataset ( $\approx 6$ h) and the CCMixer dataset [9] ( $\approx 3$ h). For U-Net [8], the authors used a private dataset containing approximately 20,000 tracks ( $\approx 2$  months). In the following we discuss a strategy to produce a dataset of comparable size from the publicly available musdb18 dataset, by means of using state-of-the-art signal transformation algorithms to produce various signal. Using this augmented dataset, we will compare trained under the same conditions and will compare the effect of the different data augmentation strategies for the U-Net and Wave-U-Net models. We also conducted an in depth analysis of variations of the U-Net architecture (no skip-connections, comparing ratio masking with direct estimation of the separated source).

In section II, we review in detail the previous works on U-Net and Wave-U-Net. Section III presents data augmentation for the singing voice separation problem and how we created a large dataset to train and evaluate our models in a conjoint framework. Section IV presents the different experiments we conducted to study and compare Wave-U-Net and U-Net models and the results of these.

## II. MODELS

### A. U-Net model

The U-Net model has been originally proposed for biological cells image segmentation in [10]. Recently, Jansson has proposed in [8] to use it for singing voice separation. This model follows an encoder-decoder scheme. The **encoder** part of the network is made of a set of convolutional layers. The goal of the encoder is to reduce the inputs dimensionality

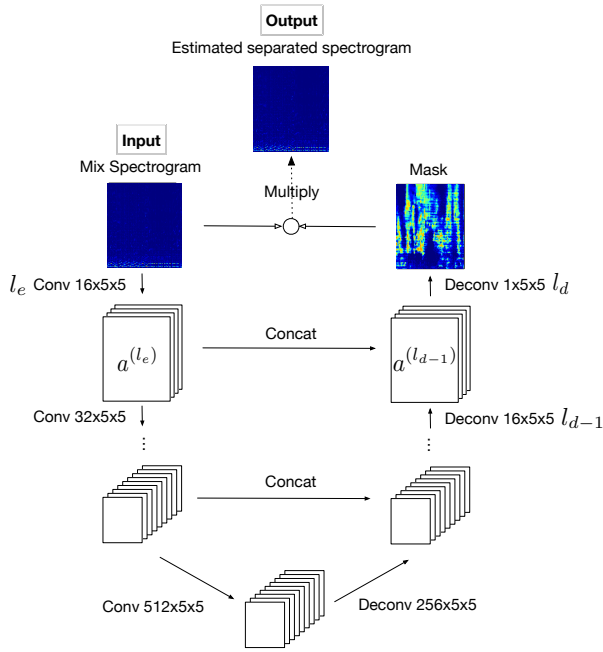


Fig. 1: U-Net architecture

while preserving relevant information for the task of interest. The **decoder** part is made of transposed convolution layers. Usually, the decoder attempts to recreate the input from the compressed representation provided by the encoder. In this case, the model is called an Auto-Encoder (AE). To apply this to singing voice separation, the input is a spectrogram excerpt (of 128 frames) of a mix track<sup>1</sup> and the output is a spectrogram of the isolated singing voice. Compared to a classical AE, the U-Net model adds two novel ideas that will be tested in part IV.

**Skip connections.** Since the U-Net has a symmetric architecture (i.e. each layer couple  $l_e$  in the encoder and  $l_d$  in the decoder have the same number of filters, sizes, strides and output dimensions), they can be connected through skip-connections (see figure 1). The motivating idea for these connections is to help the reconstruction by providing finer details in the decoding directly from the encoder (which are otherwise progressively lost during encoding).

**Output as mask.** In an usual AE, the decoder aims at reconstructing the given input. In this specific case of source separation, the decoder aims at reconstructing the spectrogram of the isolated source. In the U-Net model, instead of defining the output as the spectrogram of the isolated source  $Y$ , the output is defined as a continuous (which values range from 0 to 1) mask  $f_\theta(X)$  to be applied to the input spectrogram  $X$  to obtain the spectrogram of the isolated source  $Y$ . The loss function to be minimized is therefore defined as:  $\mathcal{L}(\mathcal{X}, \mathcal{Y}; \theta) = \|f_\theta(X) \odot X - Y\|_1^1$ . To test and analyze the specific features of the U-Net, we conducted one experiment (see IV) consisting

<sup>1</sup>For shorter notation, we will use spectrogram and spectrogram excerpt as synonyms in the remaining of this paper, and refer to full spectrogram for the spectrogram of the whole track.

in outputting directly a separated spectrogram and reproduced the experiment done in [8] of removing the skip connections.

1) *Details of the architecture, training and testing:* Each layer of the U-Net model is made of  $5 \times 5$  filters with stride 2. The first layer has 16 filters and the number of filters is doubled at each layer. The activation are Leaky ReLUs ( $\alpha = 0.2$ ). A batch normalization layer is used between all layers. The decoder part is mapped to the encoder part. The activations are ReLUs, except for the last layer which uses sigmoid activations (to keep the values of the masks between 0 and 1). Dropout ( $p = 0.5$ ) is applied on the first three layer of the decoder part. Training is done using minibatch of size 128, and ADAM [11] optimizer.

At test time, a track is processed by passing non-overlapping patches  $X$  of 128 frames of the full spectrogram of the mix through the U-Net. The full spectrogram of the isolated singing voice for the track is simply obtained by concatenating temporally the separated spectrogram outputted by the U-Net, called  $\hat{Y}$ . The audio signal is reconstructed using  $\hat{Y}$  and the phase of  $X$ .

## B. Wave-U-Net model

A recent evolution of the U-Net is the Wave-U-Net [9], an end-to-end network using a similar topology as the U-Net, but working directly on the audio signal (therefore avoiding the problems related to reconstruction of the audio signal). The evaluation proposed in [9] seems to indicate that the Wave-U-Net can achieve similar performance as the U-Net, but, due to the fact that for the evaluation in [9] only a much smaller training database was available, the conclusions would benefit from an evaluation with the augmented dataset described in the following.

**Adaptation of Wave-U-Net.** Due to the enormous size of the augmented dataset we are proposing, we adopt the strategy presented in [8] and work with audio at 8192Hz sample rate. The change of the sample rate requires adaptation of the Wave-U-Net topology. From the many different possible choices, we chose to keep the time duration of the first layer 1-d filters approximately constant, reducing the filter size from 15 to 5 bins. This filter length is the most similar time span under the constraints of the Wave-U-Net. With respect to the receptive field of the model, the most similar setup compared to the evaluation in [9] is given by a receptive field covering  $\sim 7s$  (57431 samples). This provides an output vector of about 1s (8197 samples). The corresponding values in [9] are 6.7s (147443 samples, 22,05kHz) for prediction and  $\sim 0.74s$  (16389 samples).

A particularity of [9] is the fact that the training set is split into training and validation data, where the validation data is used to stop the training when no progress is made. For comparison with the original paper, we initially followed the same strategy, but later found that, with the augmented dataset, the problem of over-fitting is rather small. Therefore, for the final comparison with U-Net, we used the full 100 samples of training data (with augmentation) in musdb18 for training.

This example will be marked as DA-F in the results table. As initial experiment, and to confirm that our implementation for reduced sample rate performs correctly, we did use the optimal model for mono input (M3 in [9]) and trained it on a random selection of 75 audio tracks without augmentation, using the other 25 tracks for early stopping after 20 epochs without improvement and evaluated this baseline model using the median SDR proposed as evaluation measure in [9]. Training is done using minibatch of size 64, and ADAM [11] optimizer. In our implementation with the slightly reduced filter size and slightly increased length of predicted output, we obtain a median SDR of 4.09dB while [9] reports 3.96dB. Given the difference in network structure, the different split in training and validation data, the reduced sample rate, this value supports our idea that the network performs similar to the original in [9].

We present in the next section the methodology we implemented to create a very large dataset in order to train and compare U-Net and Wave-U-Net.

### III. DATA AUGMENTATION FOR SINGING-VOICE

Data augmentation can be used to increase the number of training examples leading to an improved coverage of the real world signal space. To be able to augment training data without requirement for extensive re-annotation of the ground truth annotation (labels or separated signals), one needs to find means that modify the available training data such that the ground truth either does not change or changes in a predictable way so that it can be adapted automatically as part of the data augmentation procedure. In the following we focus the discussion on sound specific transformations (leaving aside transformations such as dropout or added noise).

Data augmentation of singing voice has been performed for singing voice detection in [12]. In that case, the proposed transformations are applied directly on the mel band spectrogram, treating it as an image. Accordingly, time stretching and pitch shifting the spectrogram is performed by means of dilated or compressed along the time - or frequency axis.

While these image transformations did improve results for the voice detection task, they seem less pertinent for source separation, where a precise link between waveform and spectrogram is of central importance. The operations used in [12] will change the form and width of the sinusoidal peaks and deform the attacks. At the end, the spectrogram does no longer represent any realizable signal. Moreover, we note that these spectrogram modifications can't be used for Wave-U-Net, since it is not possible to retrieve the temporal signals once the magnitude spectrogram has been modified.

[13] proposes rather basic strategies for data augmentation for singing voice separation: random swapping left/right channels for each instrument, random scaling with uniform amplitudes, random chunking into sequences for each instrument, and random mixing of instruments from different songs. The effect of the data augmentation evaluated on the DSD100 dataset remains rather limited, improving the results on average for 0.2dB for SDR metric and the vocal target on the test set

of the DSD100 dataset [tab. 2] [13]. We note that a few of the augmentation strategies rely on stereo data, a situation not covered by the present article.

The software framework `muda` has been proposed in [14] as a flexible tool for augmenting musical datasets. The framework includes transformations comprising dynamic range compression, mixing with noise, as well as time stretching and pitch shifting operations. The last two operations are implemented using the open source library `rubberband`<sup>2</sup> which according to its documentation is based on a phase vocoder algorithm that loosely implements the key points of state of the art phase vocoding: dedicated handling of transients (e.g. [15]) and intra partial vertical phase coherence (e.g. [16]). Shape-invariant processing, an essential feature for high quality speech-processing [17], is not addressed, but might not be of major importance for approaches based on masking STFT magnitudes. More importantly, `muda` does not allow modifying the spectral envelope (formants) independently of the pitch, one of the key elements for voice transformation, avoiding for example the “mickey mouse” effect when transposing the pitch up.

#### A. Proposed data augmentation strategy

The data augmentation strategy used in the following experiments benefits from the fact that the `musdb18` dataset is provided in form of 4 separate signals containing: voice, drum, bass and accompaniment. Each of the four signals is transformed separately, selecting the musically and technically most appropriate signal processing parameters, as for example, excluding the drum signal from pitch shifting transformations. The full set of transformations applied to the `musdb18` tracks contains the following operations (transposition in cents):

- pitch-shifting but preserving the spectral envelope  $\in [-300, -200, -100, 0, 100, 200, 300]$
- time-stretching  $\in [0.5, 0.93, 1, 1.07, 1.15]$
- transformation of the spectral envelope only of the singing voice  $\in [-150, -100, 0, 100, 150]$

Combining all these modifications leads to 175 possible variants (including the original) of each track. Given that the `musdb18` training data contains 100 tracks with a bit more than 6 hours, the augmented dataset (called DA) contains 15.000 tracks of music with a total duration of about 1.5 months of continuous music.

Specific considerations for the individual sources are as follows. The singing voice is transformed by means of pitch shifting, formant shifting and time stretching using a state of the art shape invariant phase vocoder [17]. The formant shifting is performed using the algorithm presented in [18]. The parameterization of the voice transformation algorithm is performed dynamically over time using as main control the  $F_0$  calculated using the swipe  $F_0$  estimation algorithm [19]. To achieve high quality formant shifting (or preservation), the order of the spectral envelope is adapted to the  $F_0$  following [20], such that the formant modification/preservation

<sup>2</sup><https://www.breakfastquay.com/rubberband/>

TABLE I: Datasets used

Acronym	training set	#tracks for training	validation set	test set
no-DA	75% train set of musdb18	75 tracks	25% train set of musdb18	test set musdb18
DA	75% train set of augmented musdb18	11 250 tracks	25% train set of augmented musdb18	test set musdb18
DA-F	100 %train set of augmented musdb18	15 000 tracks	×	test set musdb18

affects as good as possible the distinguishing characteristics of the singing voice. The window size is adapted to be four times the local period. For the drum signal, only time stretching transformations are applied using the transient preservation algorithm described in [15] and a fixed window size of 50ms. Finally, for the bass signal and the remaining accompaniments we apply a phase vocoder algorithm [16] again using transient preservation as in [15]. The transformations described above have been performed with the signal transformation kernel available in version 3 of the AudioSculpt program [21] that can be scripted and controlled via the Unix command line.

#### IV. EXPERIMENTS AND RESULTS

In the following experiments, we evaluate the separation using the `mir_eval-toolbox`<sup>3</sup>. We compute the following three metrics [22]: Source-to-Interference Ratios (SIR), Source-to-Artifact Ratios (SAR) and Source-to-Distortion Ratios (SDR) and report the median over the test database. As measure of variability, we use the median of the absolute deviation from the median (MAD) [9].

For all experiments, the starting point is the `musdb18` dataset [24]. This dataset contains 150 tracks (~10h duration) of different styles. The 150 tracks are split into 100 tracks for training, and 50 for testing. This dataset is called no-DA in Table II. Note that [8] is evaluating the U-Net model on `MedleyDB` not using early stopping<sup>4</sup>. For the experiments with Wave-U-Net [9] used only 75% of the training data of `musdb18` to perform training and keeps 25% data as a validation set used for early stopping. For comparison with [9] we use early stopping with Wave-U-Net and for comparison with U-Net we train Wave-U-Net on the full training data. To distinguish these setups we denote experiments without early stopping with an F appended to the dataset, e.g. DA-F for the dataset with full augmentation and no hold out validation data. See table I for a summary. Like in [8], we use mono signals down-sampled at 8192 Hz to reduce storage space and training time. For the U-Net experiments, we use STFTs with 1024 window length and overlap 256, normalized between 0 and 1.

**Wave-U-Net model.** Compared to original results in [9], our adapted Wave-U-Net, trained on no-DA (`musdb18` dataset without augmentation) performs slightly better: 4.09dB SDR where [9, M3 in table 2] using an extended `musdb18` dataset reports 3.96dB. An explanation might be that the evaluation in [9] uses 22.05kHz sample rate while we use only 8192Hz. Regarding the data augmentations, and using the early-stopping strategy used in [9], we can see that time stretching

and pitch shifting alone have only minor impact, for the SDR (+0.1dB). The transformation of the spectral envelope even has a negative impact: from 4.09 dB (resp. 5.52) to 3.77 (resp. 5.23) for SDR (resp. SAR). Shifting only the envelope does not seem to provide useful augmentation. Still, using all augmentation strategies leads to a +0.6 db on SDR and a +1.15 db on SIR, which might indicate that pitch and formant transformation is necessary to provide useful augmentation. The additional experiment without early stopping (DA-F) yields another +0.7dB on SDR, and gives overall our best Wave-U-Net-model. For such a large training database, early stopping doesn't seem to be beneficial.

**Unet model** Like the Wave-U-Net model, time stretching and envelope transposition do not have a strong impact. The most effective transformation is pitch shifting, giving +1db in SDR (from 4.52 to 5.20). Our hypothesis is that the other two augmentation have a minor effect on the variation in the spectral mask. Overall, using all the transformations proposed, the U-Net model gave the best performance on the test set of `musdb18`, on both SAR (from 5.76dB to 6.40 dB) and SDR (from 4.52 to 5.20). The results indicated in the first lines of Table IIb have been obtained using the original U-Net model, with skip connections and estimating masks  $f_{\theta}(X)$ . In order to further investigate those properties, we propose results for U-Net trained without skip connections and outputting directly a spectrogram instead of a mask. In line “no-skip”, we indicate the results obtained by only removing the skip-connections. We see that it damages the results, as in [8], as they drop from 6.40 to 5.60 (for SAR), from 11.98 to 9.92 (SIR) and 5.20 to 3.44 (SDR). This can be explained by the fact that, as expected, the skip connections bring a lot of details in the reconstruction, making the masks way sharper. In line “no-mask”, we change the definition of the output: instead of estimating the masks  $f_{\theta}(X)$  we directly estimate the spectrogram of the separated source  $\hat{Y}$ . We see that it also damages the results as they drop from 6.40 to 4.87 (SAR) and 5.20 to 4.18 (SDR).

**Comparison U-Net and Wave-U-Net.** The two models are very close: they both follow the encoder-decoder paradigm and both use skip connections. The difference between both is the input/output representation. The U-Net processes spectrograms and hence necessitates an extra step to reconstruct the audio signal (necessary to evaluate the model and listen to the results) which is potentially prone to artifacts. The Wave-U-Net processes directly the temporal signals and hence does not necessitates any reconstruction. However, the temporal signal is a lot more difficult to analyze. Comparing these two architectures is therefore quite interesting. Here we refer to results of dataset DA-F in Table II. We can see that the Wave-U-Net model gives the best results for all metrics:

<sup>3</sup>[http://craffel.github.io/mir\\_eval/#mir\\_eval.separation.bss\\_eval\\_sources](http://craffel.github.io/mir_eval/#mir_eval.separation.bss_eval_sources)

<sup>4</sup>`MedleyDB` is a dataset presented in [23]. 46 (out of 122) tracks of `MedleyDB` is actually included in `musdb18`.

TABLE II: Results of experiments 1), 2) and 3)

(a) Results for the Wave-U-Net models. Median and median absolute deviation (MAD) of SAR, SIR and SDR.

Model	Dataset used	SAR		SIR		SDR	
		med	MAD	med	MAD	med	MAD
W <sup>8,2</sup>	no-DA	5.52	1.96	10.87	2.02	4.09	2.07
	Stretch <sup>5</sup>	5.60	1.86	<b>12.11</b>	2.99	4.20	1.58
	Env. <sup>6</sup>	5.23	1.86	11.22	2.32	3.77	1.61
	Pitch <sup>4</sup>	<b>6.09</b>	1.65	10.68	2.40	4.18	1.99
	DA <sup>1</sup>	5.86	1.63	12.02	2.19	<b>4.67</b>	1.71
	DA-F <sup>1</sup>	<b>6.62</b>	1.80	<b>13.90</b>	2.74	<b>5.42</b>	1.72
W [9]	No-DA+CCMix <sup>11</sup>	×	×	×	×	3.96	3.0

(b) Results of U-Net models. Median and MAD of SAR, SIR and SDR.

Model	Dataset used	SAR		SIR		SDR	
		med	MAD	med	MAD	med	MAD
U <sup>7</sup>	no-DA-F <sup>9</sup>	5.76	4.21	11.75	2.05	4.52	2.48
	Stretch-F	5.73	2.28	12.38	2.48	4.85	2.06
	Env.-F	6.06	2.28	11.06	2.66	4.55	2.24
	Pitch-F	6.35	2.21	<b>12.69</b>	2.69	<b>5.20</b>	2.09
	DA-F	<b>6.40</b>	2.20	11.98	2.37	<b>5.20</b>	2.22
U [8]	DS-priv <sup>10</sup>	<b>11.30</b>	×	<b>15.31</b>	×	×	×
	no-skip DA-F	5.60	2.39	9.92	2.08	3.44	2.13
	no-mask DA-F	4.87	3.25	<b>14.71</b>	3.50	4.18	3.27

<sup>1</sup> DA: all Data Augmentation, <sup>2</sup> 25% of musdb18 kept for early stopping according to [9], <sup>3</sup> MDB: MedleyDB, [23], for the testing phase, <sup>4</sup> Pitch: pitch shifting, <sup>5</sup> Stretch: time stretching, <sup>6</sup> Env.: transformation of the singing voice spectral envelope, <sup>7</sup> U: U-Net model, <sup>8</sup> W: Wave-U-Net model, <sup>9</sup> F: Full musdb18, training on full training data - no early stopping, <sup>10</sup>: private dataset see [8], <sup>11</sup> Musdb + CCMixer datasets [9] M3 in table 2.

6.62db versus 6.40 db for SAR, 5.42 dB versus 5.20 db for SDR. While this result seem to indicate an advantage for Wave-U-Net, we consider this for the moment as only a first element. The computational complexity of both networks needs to be taken into account and, given the large dataset, an increased complexity leading to improvements for both seems possible. Ongoing work will be reported in the future, including listening tests revealing the perceptual relevance of these quantitative measures.

## V. CONCLUSION

In this paper, we proposed a new set of data augmentations designed for singing voice detection. We reviewed two singing voice separation state-of-the-art models: the U-Net model and the Wave-U-Net model. With our data augmentation strategy, we produced a very large dataset, giving us a robust conjoint framework to compare these models. We showed that the use of these augmentations improved the results over the musdb18 dataset, the largest publicly available dataset for singing voice separation, for both the U-Net and the Wave-U-Net model. However for both models, the results are rather close, which is very interesting given the different representations taken as input by the two models.

We also studied the U-Net architecture. We proved that the skip connections of the model are crucial to reconstruct the singing voice separated spectrogram. We also showed that outputting masks rather than spectrograms yields better results.

## REFERENCES

- [1] J. L. Durrieu, G. Richard, B. David, and C. Févotte, "Source/filter model for unsupervised main melody extraction from polyphonic audio signals," *IEEE Transactions on Audio, Speech and Lang. Proc.*, 2010.
- [2] Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright, "Robust principal component analysis?," *J. ACM*, 2011.
- [3] Z. Rafii and B. Pardo, "A simple music/voice separation method based on the extraction of the repeating musical structure," *Proc of ICASSP*, 2011.
- [4] S. Vembu and S. Baumann, "Separation of vocals from polyphonic audio recordings," in *Proc. of ISMIR*, 2005.
- [5] F.-R. Stöter, A. Liutkus, and N. Ito, "The 2018 signal separation evaluation campaign," in *Int. Conf. LVASS*, 2018.
- [6] Naoya Takahashi, Nabarun Goswami, and Yuki Mitsufuji, "MM-DENSELSTM: An efficient combination of convolutional and recurrent neural networks for audio source separation," *2018 16th Int. Workshop on Acoustic Signal Enhancement (IWAENC)*, 2018.
- [7] P. Chandna, M. Miron, J. Janer, and E. Gómez, "Monoaural audio source separation using deep convolutional neural networks," in *13th Int. Conf. LVA*, 2017.
- [8] A. Jansson, E. J. Humphrey, N. Montecchio, R. M. Bittner, A. Kumar, and T. Weyde, "Singing voice separation with deep u-net convolutional networks," in *Proc. of ISMIR*, 2017.
- [9] D. Stoller, S. Ewert, and S. Dixon, "Wave-u-net: A multi-scale neural network for end-to-end audio source separation," *CoRR*, 2018.
- [10] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*. 2015, LNCS, Springer.
- [11] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *CoRR*, 2014.
- [12] Jan Schlüter and Thomas Grill, "Exploring Data Augmentation for Improved Singing Voice Detection with Neural Networks," in *Proc. of ISMIR*, 2015.
- [13] S. Uhlich, M. Porcu, F. Giron, M. Enekl, T. Kemp, N. Takahashi, and Y. Mitsufuji, "Improving music source separation based on deep neural networks through <https://www.overleaf.com/2333256817phxmxjfszkhzgh> data augmentation and network blending," in *Proc. of ICASSP*, 2017.
- [14] Brian McFee, Eric J. Humphrey, and Juan Pablo Bello, "A software framework for musical data augmentation," in *Proc. of ISMIR*, 2015.
- [15] A. Röbel, "A new approach to transient processing in the phase vocoder," in *Proc. of DAFx03*, 2003.
- [16] J. Laroche and M. Dolson, "Improved phase vocoder time-scale modification of audio," *IEEE Trans. on Speech and Audio Proc.*, 1999.
- [17] A. Röbel, "Shape-invariant speech transformation with the phase vocoder," in *Proc. of InterSpeech*, 2010.
- [18] A. Röbel and X. Rodet, "Efficient spectral envelope estimation and its application to pitch shifting and envelope preservation," in *Proc. of DAFx*, 2005.
- [19] Arturo Camacho, *SWIPE: A sawtooth waveform inspired pitch estimator for speech and music*, Ph.D. thesis, University of Florida, 2007.
- [20] A. Röbel, F. Villavicencio, and X. Rodet, "On cepstral and all-pole based spectral envelope modeling with unknown model order," *Pat. Reco. Letters, Special issue on Advances in Pattern Recognition for Speech and Audio Processing*, 2007.
- [21] N. Bogaards, A. Röbel, and X. Rodet, "Sound analysis and processing with AudioSculpt 2," in *Proc. Int. Computer Music Conf. (ICMC)*, 2004.
- [22] Emmanuel Vincent, Rémi Gribonval, and Cédric Févotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech and Language Processing*, 2006.
- [23] Rachel Bittner, Justin Salamon, Mike Tierney, Matthias Mauch, Chris Cannam, and Juan Bello, "Medleydb: A multitrack dataset for annotation-intensive mir research," in *Proc. of ISMIR*, 2014.
- [24] Zafar Rafii, Antoine Liutkus, Fabian-Robert Stöter, Stylianos Ioannis Mimilakis, and Rachel Bittner, "The musdb18 corpus for music separation," dec 2017.