HTMD-Net: A Hybrid Masking-Denoising Approach to Time-Domain Monaural Singing Voice Separation

Christos Garoufis[1,2], Athanasia Zlatintsi[1,2], and Petros Maragos[1,2]
[1]*School of ECE, National Technical University of Athens, 15773 Athens, Greece*
[2]*Robot Perception and Interaction Unit, Athena Research Center, 15125 Maroussi, Greece*
cgaroufis@mail.ntua.gr, {nzlat, maragos}@cs.ntua.gr

***Abstract*—The advent of deep learning has led to the prevalence of deep neural network architectures for monaural music source separation, with end-to-end approaches that operate directly on the waveform level increasingly receiving research attention. Among these approaches, transformation of the input mixture to a learned latent space, and multiplicative application of a soft mask to the latent mixture, achieves the best performance, but is prone to the introduction of artifacts to the source estimate. To alleviate this problem, in this paper we propose a hybrid time-domain approach, termed the HTMD-Net, combining a lightweight masking component and a denoising module, based on skip connections, in order to refine the source estimated by the masking procedure. Evaluation of our approach in the task of monaural singing voice separation in the musdb18 dataset indicates that our proposed method achieves competitive performance compared to methods based purely on masking when trained under the same conditions, especially regarding the behavior during silent segments, while achieving higher computational efficiency.**

***Index Terms*—source separation, music signal processing, singing voice separation, deep learning, time-domain audio processing**

I. Introduction

Source separation is defined as the problem of decomposing an observed input signal into the components that constitute it. In the context of music processing, music source separation regards the isolation of vocal or instrumental tracks from a musical mixture. Historically, the problem of music source separation was tackled by signal processing-based methods [1], [2]. However, since the advent of deep learning, these methods have been gradually replaced by deep-learning based ones [3]–[5]. These methods can be divided in two categories: Methods that operate in a time-frequency representation of the signal, usually its Short-Time Fourier Transform (STFT), in order to perform the separation procedure [3], [6], [7], and those that directly leverage the signal waveform to separate the desired sources in an end-to-end fashion [4], [8], [9].

The majority of deep learning approaches operating in the STFT domain of a signal, inspired by traditional signal processing approaches, predict a mask which, when applied via element-wise multiplication to the input magnitude spectrogram, yields the magnitude spectrogram of the desired source [6]. On the contrary, time-domain approaches can be split in two main categories: Autoencoder architectures with skip connections, operating in multiple resolutions of the input waveform [4], [8], [10], and architectures that follow the Encoder-Separator-Decoder paradigm [9]. In the second case, the encoder and the decoder learn an overcomplete latent mixture representation, upon which the separator calculates a mask to be applied, similar to the STFT-based approaches.

Among time-domain approaches to audio source separation, neural network architectures based on the above-described Encoder-Separator-Decoder paradigm have achieved state-of-the-art performance in both speech separation [11] and music source separation [10]. However, while music source separation approaches based on masking generally outperform those primarily utilizing skip connections, a drawback of these approaches regards the introduction of noise artifacts in the predicted source [10], [12], [13]. Previous works in the field [13], [14] that use an STFT-representation of the signal attempt to overcome this problem via refining the initial mask estimate by serially stacking either similar [14] or suitably designed [13] modules upon the initial masking network, and training the whole network in an end-to-end fashion.

In this work, we propose the HTMD-Net (short for Hybrid Temporal Masking-Denoising Network), a hybrid architecture for end-to-end monaural music source separation, consisting of two serially connected modules: one that provides an initial source estimate via applying a mask to an overcomplete learned latent representation of the mixture, and a second, based on multi-resolution analysis, that refines the initial estimate. While our approach shares the concept of serial module connection with [13], [14], it deviates from those in that it operates directly in the time domain, as well as in the design of these modules. Furthermore, since the proposed framework allows the application of deep supervision, we conducted a number of experiments to gauge the behavior of HTMD-Net with respect to the training protocol used. Our proposed approach achieves competitive performance in the task of monaural singing voice separation in the widely used musdb18 [15] dataset, compared to time-domain approaches

This work is supported by the *iMuSciCA* project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 731861.

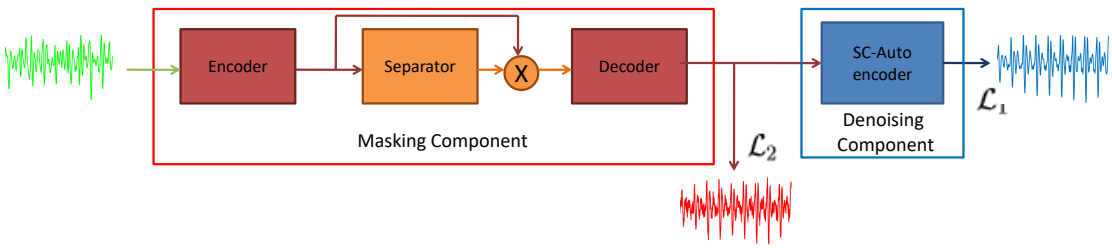

Fig. 1. An overview of the whole system architecture, including the masking and denoising components and the separate losses over the intermediate and final source estimates.

purely based on either masking or multi-resolution analysis, especially regarding the alleviation of inter-source interferences, as well as higher computational efficiency.

The rest of the paper is organized as follows: In Sec. II, we outline the proposed architecture in detail. The experimental setup we utilize is described in Sec. III, while in Sec. IV we report and discuss our results. Finally, in Sec. V we summarize our conclusions and propose some future research directions.

II. Methodology

The principal concept presented in this work regards decomposing the procedure of isolating a specific source from a time-domain mixture into two separate subprocesses: 1) Finding an optimal mask to be applied on a latent mixture representation, as in [9], and 2) refining the masked source estimate through a denoising module based on skip-connections. This two-step procedure is presented in Fig. 1.

A. Masking Network

To perform the initial masking operation, we use a masking neural network, similar to [9], that consists of a) a linear convolutional strided encoder that transforms the input mixture to a latent space, b) a mask estimation module, and c) a linear decoder that consists of transposed convolutions, which reverts the estimated source to the time domain. In the original Conv-TasNet architecture [9], the mask estimator is realised as a dilated Temporal Convolutional Network (dilated-TCN) and consists of R modules, placed in succession. In turn, each module comprises of consecutive blocks that apply successively 1x1 Convolutions and Depthwise Separable Convolutions with increasingly dilated kernels. Each convolutional block has two outputs, a mask estimate and a feature map to be used as input from the next block. The mask estimates from all blocks are summed together, and then scaled in a $[0, 1]$ range via a sigmoid activation, in order to be multiplied with the encoder's output, thus yielding a representation of the desired source in the encoder's latent space.

In [9], a total of 3 modules, each consisting of 8 convolutional blocks, were used, hence setting the maximum dilation rate to 2^8 [9]. In this work, we use only one module with 9 convolutional blocks, in order to increase its receptive field. Furthermore, after preliminary experiments, we replaced the PReLU activations and Layer Normalization operations with LeakyReLUs and Batch Normalization, respectively. The rest of the network's hyperparameters were left unchanged.

B. Denoising Network

In order to refine the source estimate produced by the masking network, we serially attach a second trainable module to the masking component, to perform a denoising operation on its output. We opted for an encoder-decoder network utilizing skip connections, since these architectures have proved efficient in the task of speech enhancement [16], [17]. Thus, an architecture similar to the Wave-U-Net [4] was used, replacing the convolutional bottleneck of the network with a recurrent module. The recurrent path consists of two bidirectional LSTM layers, as also proposed in [10], [18], of 168 units each, and a LeakyReLU activation after the second layer. Also, in order to keep the computational costs low, the number of filters in both the encoder and the decoder were halved compared to the original implementation [4]. Otherwise, the structure of the network follows [4]: the encoder block consists of 1D-convolutional filters, followed by a LeakyReLU activation and a downsampling layer. Similarly, the decoder alternates between upsampling layers and 1D-convolutions, again followed by a LeakyReLU activation - with the exception of the output layer, which uses a tanh activation. The outputs of each encoder convolutional block pre-downsampling are concatenated with the feature maps of the respective decoder block after being upsampled via skip connections.

C. Deep Supervision

Similar to [13], [14] we experiment with the application of deep supervision during training the network. Namely, we optimize the loss function:

$$\mathcal{L} = \alpha\mathcal{L}_1 + \beta\mathcal{L}_2, \tag{1}$$

where the losses $\mathcal{L}_1$, $\mathcal{L}_2$ correspond to the final and the intermediate source estimates, respectively, and α, β correspond to the loss weights.

A potential advantage of the proposed deeply supervised framework regards the ability to incorporate different loss functions on the network's bottleneck and the final output. Since both $\mathcal{L}_1$, $\mathcal{L}_2$ are applied in the time domain, and not in a latent space, we experiment with using combinations of the mean square error (MSE) and mean absolute error (MAE) between the true and estimated sources as loss functions.

III. Experimental Setup

A. Dataset

For our experiments in singing voice separation, we utilize the musdb18 [15] dataset. This dataset consists of a total of

TABLE I

COMPARISON OF THE HTMD-NET TO A REIMPLEMENTATION OF CONV-TASNET [9] AND A WAVE-U-NET [4]. BOLD DENOTES THE BEST RESULTS AT A STATISTICAL SIGNIFICANCE LEVEL OF $p < 0.01$. HIGHER VALUES ARE BETTER FOR ALL METRICS, EXCEPT PES (DB).

Method	Loss Function	Song-Wise Metrics			Segment-Wise Metrics				
		SDR (dB)	SIR (dB)	SAR (dB)	SDR (dB)	SIR (dB)	SAR (dB)	PES (dB)	VAD (%)
HTMD-Net	(MSE, MSE)	5.16	**10.24**	8.53	**4.69/0.60**	**9.80/6.98**	7.92/6.53	**-62.2**	**84.7**
Conv-TasNet*	MSE	**5.25**	**9.74**	**8.85**	4.83/-0.07	9.59/7.00	**8.18/7.09**	-57.8	82.5
Wave-U-Net	MSE	4.37	9.46	7.61	4.04/-0.14	9.00/6.49	7.17/6.24	-61.4	82.1
HTMD-Net	(MAE, MAE)	5.18	**11.30**	8.43	4.62/2.26	**11.44/9.95**	8.14/6.24	**-80.1**	**85.3**
Conv-TasNet*	MAE	**5.20**	10.73	**8.82**	**4.84/1.63**	10.81/8.80	**8.44/6.83**	-73.1	85.2
Wave-U-Net	MAE	4.07	9.67	8.17	3.61/0.90	9.62/8.00	7.48/5.90	-70.0	82.8

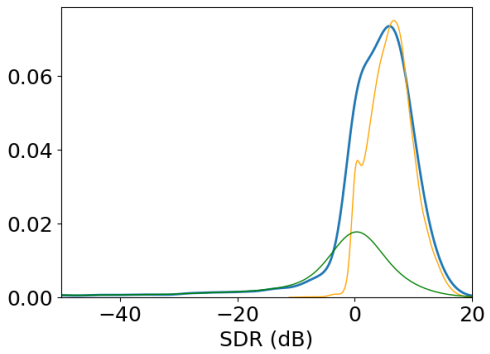

Fig. 2. Kernel density estimate (KDE) for the segment-wise SDR for HTMD-Net, superimposed with the normalized KDEs of the segment-wise SDR corresponding to near-silent (green) or non-silent (orange) segments.

150 songs at stereo format and sampled at 44.1 kHz, as well as separate tracks for the vocals, bass, drums and the rest of the instrumental components (accompaniment) of each song, divided into a training set of 100 songs and a testing set of 50 songs. As preprocessing, we downsampled the audio excerpts corresponding to the song mixtures and the vocals to 22.05 kHz, after conversion from stereo to mono, as in [4].

B. Training Setup and Baselines

As baselines, we employ the following architectures:

- A Conv-TasNet, consisting of 3 repetitions of the dilated-TCN separation module, as proposed originally in [9]. The network's hyperparameters were set according to the optimal setup in [9], with the exception of Batch Normalization and LeakyReLU activations, since those were used in the masking component of HTMD-Net, and using an input length of 16384 samples. We note that we have used 9 dilated blocks in each repetition, in order to achieve a receptive field at least equal to the input length.
- A Wave-U-Net architecture [4], with the network's hyperparameters following the original implementation.

All of the tested architectures were implemented in Keras and trained with the Adam optimizer [19] with a learning rate of 0.0001, using a batch size of 16, with the exception of the Conv-TasNet, where we used a batch size of 8 due to memory limitations. All networks were trained using either the MSE or the MAE between the true and estimated sources as the loss function $\mathcal{L}_1$. The musdb18 training set was split in training and validation data, by using 75 out of the 100 songs for network training, and the rest as a validation set. No data augmentation was performed, and early stopping was applied after 20 epochs of no improvement in the validation set.

C. Evaluation Protocol

As our primary metric, we utilize the Signal-to-Distortion Ratio (SDR) between the true and predicted sources in the musdb18 test set, estimated over 1-sec segments. We further report, in accordance with [20], on the Signal-to-Artifact Ratio (SAR) and the Signal-to-Interference Ratio (SIR) - the former is used to gauge the existence of auditory artifacts, while the latter measures the contamination of the extracted sources. We report on both the song-wise median, as in [21], and the segment-wise median and mean, similar to [4].

However, the above metrics are insufficient in assessing the performance of source separation algorithms in the time domain when used standalone, since they are not defined over silent segments of audio. Thus, we also employ as metrics the mean predicted energy at silence (PES), as in [22], [23], measured in 4096-sample frames, with a negative threshold of -100dB, and the correct vocal activity detection (VAD) percentage as measured in 20-ms frames of the network's output. To acquire the ground-truth VAD labels, we applied pyvad, a wrapper for the WebRTC Voice Activity Detection system, on the original vocal tracks.

IV. RESULTS AND DISCUSSION

Comparison to Baselines: The quantitative results of how HTMD-Net performs compared to the purely masking-based and skip-connection based baselines are presented in Table I. We observe that our proposed architecture clearly outperforms the base Wave-U-Net [4], as well as performs comparably to our re-implementation of Conv-TasNet [9]. We note that the median SDR value corresponding to the Conv-TasNet is lower compared to that reported in [10]. This is likely due to a variety of factors, including the increased model size used in [10] or the stereo-mono conversion performed in our case.

In order to assess whether the reported metric deviations between HTMD-Net and the two baselines could be attributed to random chance, pairwise statistical significance tests were performed for all metrics, between networks trained with the same training protocol. In specific, the paired Wilcoxon signed-rank test was performed over the distributions of all continuous metrics, and the paired McNemar's test over the binary variable denoting vocal activity estimation, using a p-value of 0.01 in both cases. The results indicate that in comparison to the Conv-TasNet, HTMD-Net performs comparably considering the SDR, recording a lower median but a higher mean value. Additionally, it performs better in the absence of

TABLE II

COMPARISON BETWEEN THE TRAINING PROTOCOLS USED FOR HTMD-NET. HIGHER VALUES ARE BETTER FOR ALL METRICS, EXCEPT PES (DB).

Loss Functions	Loss Weights	Song-Wise Metrics			Segment-Wise Metrics				
$(\mathcal{L}_2, \mathcal{L}_1)$	(β, α)	SDR (dB)	SIR (dB)	SAR (dB)	SDR (dB)	SIR (dB)	SAR (dB)	PES (dB)	VAD (%)
(MSE, MSE)	(0.5, 1)	5.16	10.24	8.53	4.69/0.60	9.80/6.98	7.92/6.53	-62.2	84.7
(MAE, MAE)	(0.5, 1)	5.18	**11.30**	8.43	4.62/**2.26**	**11.44/9.95**	8.14/6.24	-80.1	**85.3**
(MAE, MSE)	(0.05, 1)	5.16	10.33	8.36	4.68/0.34	9.97/7.87	8.06/6.65	-59.9	84.2
(MSE, MAE)	(1, 0.1)	5.21	11.29	8.34	4.74/2.21	10.90/9.03	7.95/6.04	**-82.5**	85.0
(-, MSE)	-	**5.30**	10.05	8.62	**4.76**/0.10	9.76/7.79	**8.21/6.85**	-57.1	82.4
(-, MAE)	-	4.77	9.88	**8.63**	4.37/1.88	9.58/8.02	7.94/6.43	-74.5	84.8

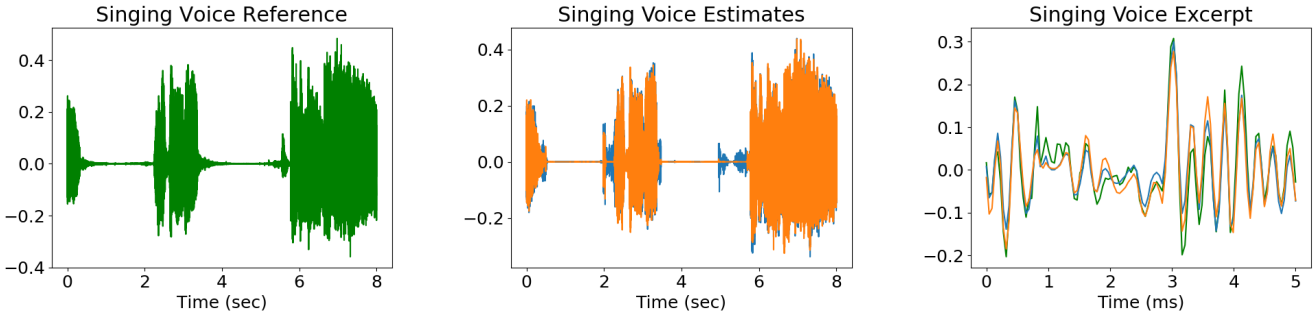

Fig. 3. An 8-sec vocal track segment from the musdb18 test set, in green (left), the singing voice estimates for this segment provided by Conv-TasNet and HTMD-Net, in blue and orange respectively (center), and an utterance-level plot for both the reference and vocal estimates (right), using the same color code.

a vocal source, as it can be inferred from the lower PES and higher correct VAD percentage. We also note that in general, HTMD-Net records higher SIR scores, but lower SAR scores, in comparison to Conv-TasNet. This trend could be attributed to deep supervision, since multiple applications of the loss function should make the extracted source less contaminated by inter-source interferences. Finally, in comparison to the Wave-U-Net, the reported improvements of HTMD-Net are deemed statistically significant over all metrics.

By definition, the SDR is sensitive to outliers corresponding to near-silent segments [4], which explains the big difference between the segment-wise median and mean SDR, and also the higher mean SDR reported for the HTMD-Net, since it performs better in near-silence. This effect is visualized in Fig. 2, where the kernel density estimate (KDE) of the segment-wise SDR values is displayed for the HTMD-Net (blue), superimposed with the normalized KDEs of the subsets of the SDR values that correspond to near-silent (green) or non-silent (orange) 1-sec segments, as classified by pyvad. We observe that the vast majority of the outlier SDR values correspond to near-silent segments, since their SDR distribution almost overlaps with the overall one in negative SDR values. A similar trend was observed regarding SIR and SAR as well.

Training Loss Schemes: Upon inspection of Table I, it is noted that using MAE as a loss function instead of the MSE does not necessarily imply improved network performance, but almost certainly provides more stable behavior regarding energy suppression in silent segments. Motivated by this, we also train variants of HTMD-Net using different loss functions in the bottleneck of the network and the final source estimate.

The results are reported in Table II, along with HTMD-Net variants trained without any deep supervision. We observe that among those variants, the model that was trained using the MSE as $\mathcal{L}_2$, and the MAE as $\mathcal{L}_1$, respectively, achieves competitive performance in most of the reported metrics.We assume that with this loss function combination, the mask estimation module focuses more on following the vocal contour accurately, while the skip-connection module refines the initial estimation by enforcing silence in non-vocal segments.

We further note that deep supervision has a significant effect in the quality of the network's output, especially regarding the Source-to-Interference Ratio (SIR), the mean segment-wise SDR values, and the silent segment performance as measured by PES. However, non-deeply supervised HTMD-Net variants achieve consistently good SAR values, and in the case of using the MSE loss, the song-wise median SDR is actually improved over our reimplementation of Conv-TasNet.

TABLE III

COMPARISON BETWEEN THE INTERMEDIATE OUTPUTS IN THE BOTTLENECK OF THE HTMD-NET, DEPENDING ON THE $\mathcal{L}_2$ USED, WHEN USING THE MAE AS $\mathcal{L}_1$.

$\mathcal{L}_2$	SDR (dB)	SIR (dB)	SAR (dB)
MSE	**4.36**/-1.00	**8.21/5.78**	7.23/5.62
MAE	4.09/**0.20**	**8.21**/5.29	7.79/7.06
-	-6.31/-13.1	2.81/1.22	**7.26/7.25**

Behavior of Intermediate Output: In Table III, we report on the median and mean segment-wise SDR, SIR and SAR values for all HTMD-Net variants trained using the MAE as $\mathcal{L}_1$, measured at the bottleneck of the network where $\mathcal{L}_2$ was applied. We observe that while the deeply supervised variants record higher SDR and SIR values at the bottleneck, in the case where no $\mathcal{L}_2$ was applied, the reported SAR values are competitive, despite the lack of any supervision at this point. Given the overall performance of the non-deeply supervised variants, these results merit further exploration.

Qualitiative Results[1] : In Fig. 3 (left), we present an 8-second segment of the track "Secretariat - Over the Top"

[1]Audio samples/code available at: https://github.com/cgaroufis/HTMD-Net

from the musdb18 test set. We can see that this segment contains two silent sections at 1 and 4 sec. From Fig. 3 (center), we can deduce that the performance of Conv-TasNet (blue) significantly deteriorates in the silent sections, whereas HTMD-Net (orange) is more successful in removing the other active instrumental sources in these areas. On the other hand, the vocal estimate provided by Conv-TasNet is closer to the reference vocals (green) regarding the utterance-level vocal peaks, as inferred from Fig. 3 (right). These observations agree with the quantitiative results presented earlier, since HTMD-Net achieves slightly lower median SDR compared to the Conv-TasNet, but better performance at silent sections as measured by VAD (%), PES, as well as the mean SDR.

Runtime Comparison: Finally, in Table IV, the total model sizes for Conv-TasNet, Wave-U-Net, and HTMD-Net are presented, along with the required time (in sec) to process 30 sec of audio, sampled at 22.05 kHz, in an AMD-A9 CPU and an NviDIA Ge-Force GTX 1080 GPU, respectively, averaged over 5 runs. We note that the HTMD-Net has a marginally smaller model size compared to our Conv-TasNet adaptation, and below half the size of a Wave-U-Net. The processing time is higher than the one recorded for the Wave-U-Net, but significantly lower compared the one of Conv-TasNet, and approaches real-time performance even on the AMD A9 CPU.

TABLE IV
COMPARISON OF THE HTMD-NET TO A REIMPLEMENTATION OF CONV-TASNET [9] AS WELL AS A WAVE-U-NET [4], REGARDING EXECUTION RUNTIME AND PARAMETER FOOTPRINT.

Method	CPU-time	GPU-time	# Params
Conv-TasNet*	140.7	0.65	5.5M
Wave-U-Net	13.6	0.07	10.3M
HTMD-Net	50.5	0.14	4.5M

V. CONCLUSIONS

In this paper, we presented a hybrid approach to monaural singing voice separation that employs both a masking component in order to find an optimal separating mask and a denoising module with skip connections to reduce the inter-source interference artifacts introduced by the masking procedure. The results of our method are promising, since HTMD-Net is able to perform competitively with the best-performing time-domain architectures when trained under similar settings, achieving a more stable behavior in silent sections, while maintaining a smaller parameter footprint and requiring less time for inference. In the future, we are interested in examining whether our findings can scale to larger input lengths and time-domain adaptations of architectures that are designed to handle multiple sources [24], [25], as well as testing HTMD-Net in a speech enhancement/separation framework. Finally, perceptual subjective evaluation tests could be performed, in order to support the objective results presented in this work.

REFERENCES

[1] T. Virtanen, "Monaural Sound Source Separation by Nonnegative Matrix Factorization with Temporal Continuity and Sparseness Criteria," *IEEE Transactions on Audio, speech, and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[2] P. Huang, S. Chen, P. Smaragdis, and M. Hasegawa-Johnson, "Singing Voice Separation from Monaural Recordings using Robust Principal Component Analysis," in *Proc. ICASSP 2012*, Kyoto, Japan, 2012.

[3] F.-R. Stöter, S. Uhlich, A. Liutkus, and Y. Mitsufuji, "Open-unmix-a Reference Implementation for Music Source Separation," in *Journal of Open Source Software*, 2019.

[4] D. Stoller, S. Ewert, and S. Dixon, "Wave-U-Net: A Multi-Scale Neural Network for End-to-End Audio Source Separation," in *Proc. ISMIR 2018*, Paris, France, 2018.

[5] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, D. FitzGerald, and B. Pardo, "An Overview of Lead and Accompaniment Separation in Music," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 8, pp. 1307–1335, 2018.

[6] A. Jansson, E. Humphrey, N. Montecchio, R. Bittner, A. Kumar, and T. Weyde, "Singing Voice Separation with Deep U-Net Convolutional Networks," in *Proc. ISMIR 2017*, Suzhou, China, 2017.

[7] N. Takahashi, N. Goswami, and Y. Mitsufuji, "MMDenseLSTM: An Efficient Combination of Convolutional and Recurrent Neural Networks for Audio Source Separation," in *Proc. IWAENC 2018*, Tokyo, Japan, 2018.

[8] E. Grais, D. Ward, and M. Plumbley, "Raw Multi-Channel Audio Source Separation using Multi-Resolution Convolutional Auto-Encoders," in *Proc. EUSIPCO 2018*, Rome, Italy, 2018.

[9] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing Ideal Time–Frequency Magnitude Masking for Speech Separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 8, pp. 1256–1266, 2019.

[10] A. Défossez, N. Usunier, L. Bottou, and F. Bach, "Music Source Separation in the Waveform Domain," *arXiv preprint arXiv:1911.13254*, 2019.

[11] Y. Luo, Z. Chen, and T. Yoshioka, "Dual-path RNN: Efficient Long Sequence Modeling for Time-domain Single-channel Speech Separation," in *Proc. ICASSP 2020*, Barcelona, Spain, 2020.

[12] E. Cano, D. FitzGerald, A. Liutkus, M. Plumbley, and F.-R. Stöter, "Musical Source Separation: An Introduction," *IEEE Signal Processing Magazine*, vol. 36, no. 1, pp. 31–40, 2018.

[13] K. Drossos, S. I. Mimilakis, D. Serdyuk, G. Schuller, T. Virtanen, and Y. Bengio, "MaD TwinNet: Masker-Denoiser Architecture with Twin Networks for Monaural Sound Source Separation," in *Proc. IJCNN 2018*, Rio de Janeiro, Brazil, 2018.

[14] S. Park, T. Kim, K. Lee, and N. Kwak, "Music Source Separation using Stacked Hourglass Networks," in *Proc. ISMIR 2018*, Paris, France, 2018.

[15] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, and R. Bittner, "MUSDB18- A Corpus for Music Separation," 2017. [Online]. Available: https://doi.org/10.5281/zenodo.1117372

[16] S. Pascual, A. Bonafonte, and J. Serra, "SEGAN: Speech Enhancement Generative Adversarial Network," in *Proc. Interspeech 2017*, Stockholm, Sweden, 2017.

[17] R. Giri, U. Isik, and A. Krishnaswamy, "Attention Wave-U-Net for Speech Enhancement," in *Proc. WASPAA 2019*, New Paltz, NY, USA, 2019.

[18] E. T. Kaspersen, T. Kounalakis, and C. Erkut, "Hydranet: A Real-Time Waveform Separation Network," in *Proc. ICASSP 2020*, Barcelona, Spain, 2020.

[19] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc. ICLR 2015*, San Diego, CA, USA, 2015.

[20] E. Vincent, R. Gribonval, and C. Févotte, "Performance Measurement in Blind Audio Source Separation," *IEEE transactions on audio, speech, and language processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[21] F.-R. Stöter, A. Liutkus, and N. Ito, "The 2018 Signal Separation Evaluation Campaign," in *Proc. LVA/ICA 2018*, Guildford, UK, 2018.

[22] K. Schulze-Forster, C. Doire, G. Richard, and R. Badeau, "Weakly Informed Audio Source Separation," in *Proc. WASPAA 2019*, New Paltz, NY, USA, 2019.

[23] O. Slizovskaia, G. Haro, and E. Gómez, "Conditioned Source Separation for Music Instrument Performances," *arXiv preprint arXiv:2004.03873*, 2020.

[24] G. Meseguer-Brocal and G. Peeters, "Conditioned-U-Net: Introducing a Control Mechanism in the U-Net for Multiple Source Separations," in *Proc. ISMIR 2019*, Delft, the Netherlands, 2019.

[25] V. Kadandale, J. Montesinos, G. Haro, and E. Gómez, "Multi-task U-Net for Music Source Separation," in *Proc. MMSP 2020*, Tampere, Finland, 2020.