NEURAL NETWORKS FOR INTERFERENCE REDUCTION IN MULTI-TRACK RECORDINGS

Rajesh R [1], *Padmanabhan Rajan*[2]

Indian Institute of Technology, Mandi
[1] s21005@students.iitmandi.ac.in, [2] padman@iitmandi.ac.in

ABSTRACT

Multi-track recordings are sometimes created by simultaneously capturing several sources with several microphones. This scenario can result in the interference of undesired source(s) in the various tracks. Interference reduction aims to recover the source(s) associated with a particular track. In this paper, we present two neural networks for interference reduction. The first network uses a convolutional autoencoder-based architecture and uses time-frequency representation as input. The second network uses a truncated U-net architecture and directly estimates the interference from the time-domain multi-track representation. Our experiments indicate the effectiveness of the proposed methods, with the truncated U-net showing superior performance. Also, the audio outputs produced by the proposed methods have improved quality, resulting in better music source separation performance. Code is available at https://github.com/its-rajesh/IRMR/

***Index Terms*—** interference reduction, music source separation, multi-track recordings

1. INTRODUCTION

The availability of multi-track music recordings is useful in a variety of music information retrieval tasks such as source separation, genre classification, and instrument recognition, to name a few. But creating multi-track recordings consisting of several sources (vocals, instruments etc.) is time-consuming and expensive. Studios can record each source in isolation, which can then be mixed later. Alternatively, multi-track recordings can be made during live concerts. In this case, although the microphones corresponding to various sources can be acoustically isolated to some extent, such recordings frequently suffer from interference from the various sources in each track. This is also termed bleeding, leakage or cross-talk. Such bleeding artefacts reduce the utility of each individual track. For instance, building source separation models require the use of individual, isolated sources in each track. Thus, multi-track recordings having bleeding artefacts cannot be used effectively in creating source separation models. Hence, bleeding reduction, or interference reduction, in multi-track recordings is a useful application.

The principles of generalised Wiener filtering in the time-frequency domain underlie the majority of contemporary strategies for interference reduction. In this paper we propose two neural network alternatives for this task: (a) a convolutional autoencoder (CAE) model that works in the time-frequency domain and (b) a truncated U-net (t-UNet) that works in the time domain. The CAE model learns to remove the interference under the assumption that it is noise. The t-UNet assumes a source separation model and enables the learning of nonlinear relationships between the various time-aligned tracks at different levels of abstraction. The resulting networks are significantly faster than generalised Wiener filtering based methods.

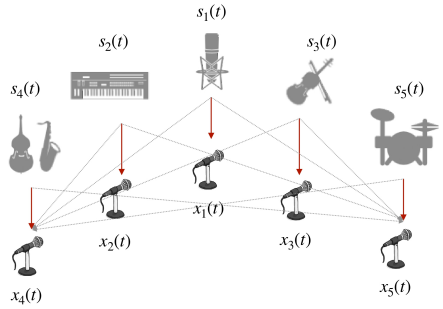

Figure 1: Illustration of interference effects in live recordings.

2. RELATED WORK

Of late, most state-of-the-art music source separation systems [1, 2, 3] are deep neural network based. The interference reduction problem can be thought of as a special case of the source separation problem. Alternately it can also be thought of as a signal denoising problem. Moreover, most works for interference reduction make use of the assumption that the microphone(s) physically nearest to a source maximally captures the source, and captures other sources to a lesser extent.

The time-frequency domain techniques [4, 5, 6] have been the main focus of interference reduction algorithms, which have produced good results. The state-of-the-art KAMIR (Kernel Additive Modeling for Interference Reduction) [7] estimates the clean sources through generalized Wiener filtering by iteratively estimating the power spectral density and its corresponding strength by non-negative matrix factorisation with the β divergence [8].

MIRA (Multi-track Interference Reduction Algorithm), an advancement of KAMIR, replaces the power spectral density with the fractional power density inspired from [9]. Also, it simplifies KAMIR by dropping the kernel filtering step and the frequency dependence on strength of the sources. Both algorithms are time-consuming and unsuitable for practical full-length recordings. Recently, FastMIRA [10] was introduced to overcome the time complexity issues in MIRA by using random projections, producing satisfactory results comparable to the parent algorithm MIRA.

In earlier techniques, time-domain-based echo cancellation, IIR filters [11], and adaptive filtering [12] were explored. [13] proposes an algorithm that estimates the interference spectra through gradi-

979-8-3503-2372-6/23/$31.00 ©2023 IEEE

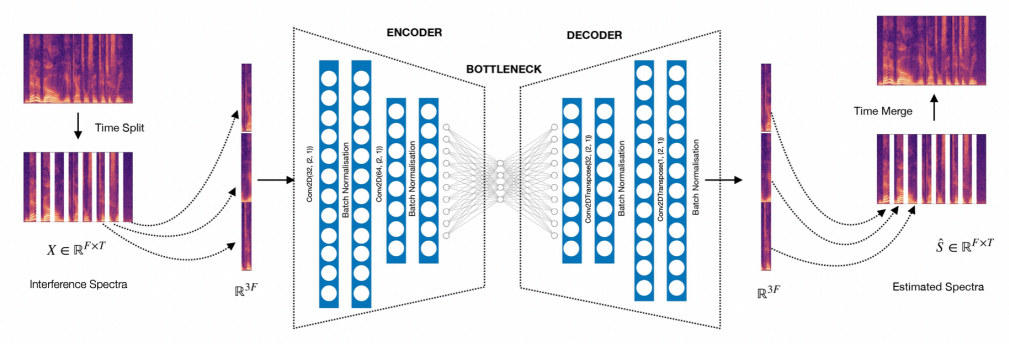

Figure 2: Convolutional autoencoder (CAE) for interference reduction.

ent descent by minimising the distance between the actual and estimated interference spectra, and then obtains the clean source by performing spectral subtraction. The disadvantage of this method is that it assumes that the sources present in the track are known beforehand.

When compared to time-domain filtering in the aforementioned methods, Wiener filtering-based time-frequency approaches produced good results. However, the sound quality, the blind source separation (BSS) evaluation metrics like source to distortion ratio (SDR), and the time complexity of these models are poor, making them unsuitable for use in many real-world scenarios.

3. CONVOLUTIONAL AUTOENCODER MODEL

Convolutional autoencoders (CAEs) have been used for processing audio inputs in both the time domain and the time-frequency domain. CAEs have been used to separate music of various genres from speech in [14]. They have also been used for single [15] and multichannel speech enhancement [16].

The proposed CAE model for interference reduction is depicted in Figure 2. In this approach, we treat the interference in each track as noise. If $x(t)$ represents the signal in a particular track, then we have

$$x(t) = s(t) + n(t), \tag{1}$$

where $s(t)$ represents the source(s) associated with the track, and $n(t)$ is the unwanted signals from all other sources. Let $X(f,t) \in \mathbb{R}^{F \times T}$ be the short-time magnitude spectrum of the input $x(t)$. Using paired training data, the CAE learns the relationship between the short-time magnitude spectra of $x(t)$ and $s(t)$. During evaluation, given the input $X(f,t)$ containing interference, the CAE estimates the spectrum of $s(t)$, denoted in the figure by $\hat{S}(f,t)$. To give temporal context, the input is provided by stacking three frames of the magnitude spectra, and is reshaped at the output.

3.1. Network architecture

The network consists of an encoder, decoder, and dense layers as shown in Figure 2. The encoder contains two sets of 2D convolution layers, each followed by batch normalisation layers. The convolution layer has 32 (2×1) and 64 (2×1) kernel filters. The encoder outputs latent features of size (e_1, e_2). We flatten the feature and pass it to a dense layer (100 neurons with RELU activation), then another dense layer ($e_1 e_2$ neurons with RELU activation) and reshape it to pass as input to the decoder. The decoder has a symmetric structure, the last layer of which outputs the estimated source spectrum $\hat{S}(f,t)$. The output signal is reconstructed by taking the inverse short-time Fourier transform, using the input phase.

4. INTERFERENCE LEARNING-BASED REDUCTION

Despite producing good results, the CAE model may have difficulty generalizing to new types of sources. For instance, the CAE for reducing interference in vocal track may not work effectively for reducing interference in drum track. Thus, a separate CAE has to be created for handling each track. Also, the CAE works on the magnitude spectrogram, which discards phase information. Recent source separation models, such as those proposed in [3, 17], reveal that time-domain approaches outperform spectrogram-based techniques. To better generalise the interference reduction for various sources and to avoid artefacts due to short-term processing, we propose our second learning framework called t-UNet.

Consider the scenario where we have K microphones capturing N sources, with $K \geq N$, and the assumption that each source has at least one dedicated microphone. The dedicated microphone(s) for a source can be thought of as predominantly capturing the signal from that source, and to a lesser extent, signals from the other sources. Thus the signal received at the kth microphone can be represented as

$$x_k(t) = \lambda_{k1} s_1(t) + \lambda_{k2} s_2(t) + \ldots + \lambda_{kN} s_N(t), \tag{2}$$

where λ_{kn} represents the gain of the acoustic path from the nth source to the kth mic, and $s_n(t)$ represents the nth true source.

In general, let $X \in \mathbb{R}^{K \times L}$ represent the time-aligned signal received by K microphones corresponding to an audio signal of L samples, and let $S \in \mathbb{R}^{N \times L}$ represent the true sources. Then the relationship between X and S is captured by the $\mathbb{R}^{K \times N}$ interference matrix Λ such that,

$$X = \Lambda S, \tag{3}$$

where,

$$\Lambda = \begin{pmatrix} \lambda_{11} & \lambda_{12} & \ldots & \lambda_{1N} \\ \lambda_{21} & \lambda_{22} & \ldots & \lambda_{2N} \\ \vdots & & & \vdots \\ \lambda_{K1} & \lambda_{K2} & \ldots & \lambda_{KN} \end{pmatrix}$$

$$X = \begin{bmatrix} x_1(t) & x_2(t) & \ldots & x_K(t) \end{bmatrix}^T$$

$$S = \begin{bmatrix} s_1(t) & s_2(t) & \ldots & s_N(t) \end{bmatrix}^T$$

By making use of the observation that all the rows in X are interrelated, in t-UNet, we learn the interference matrix Λ in the time

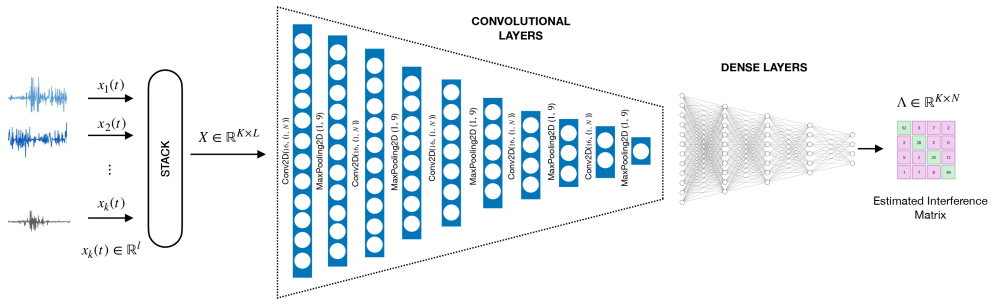

Figure 3: t-UNet for interference reduction.

domain. By utilising relevant training data consisting of pairs of X and the corresponding Λ (equation 3), the relationship between the interfered signal and the individual sources is inferred by the network. Once Λ is inferred, the interference reduction is achieved by approximating the true sources S as

$$\hat{S} = \Lambda^{\dagger} X, \tag{4}$$

where $\dagger$ represents the pseudoinverse.

4.1. Network architecture

The network consists of an encoder and dense layers as shown in Figure 3. The encoder has five levels of convolution layers with size $(1 \times K)$ and max-pooling with kernel size (1×9). This ensures K-dimensional input throughout the network. The encoder is adapted from Wave-U-Net [3] without concat connections. The encoder encodes the relation between the sources and gives a meaningful feature representation. Five fully connected layers of size 512, 128, 64, 32 and KN are used. The output of the last layer is reshaped to give the interference matrix Λ.

5. EXPERIMENTAL EVALUATION

5.1. Datasets & Experiments

We mainly utilise the standard MUSDB dataset [18] to evaluate both the proposed networks. MUSDB has four stems[1], consisting of vocals, bass, drums, and other. Since these stems have isolated sources (with no interference), they are artificially mixed to simulate interference, giving X. Having access to the isolated sources allows us to use BSS evaluation metrics to report the effectiveness of various techniques. The performance is evaluated separately for both high interference and low interference conditions. We also evaluate the networks on more realistic recordings by introducing time delays and room responses to MUSDB dataset by utilising `pyroomacoustics` [19]. The resulting dataset is termed as MUSDBR. The same train/test setup followed for both.

CAE training: For training the CAE, data was generated by interfering a given stem with the other three stems, each reduced by 20 dB. The stems were chosen from the same track, and this was repeated for all the tracks in the train subset of MUSDB, resulting in interfered versions of each stem. Spectrograms were computed with a window size of 93 msec, with a hop of 25% and a 2048 point FFT. A temporal context of three frames is used as shown in Figure 2. The CAE was trained using pairs of clean and interfered spectrograms, using mean-square error (MSE) loss function, Adam optimizer with a learning rate of 0.001, and a batch size of 16. Separate CAEs are trained for vocals, bass, drums and other stems.

t-UNet training: For t-UNet, we generate Λ such that the diagonal is dominant and off diagonals are chosen uniformly randomly in the range $[0.01, 0.5]$. This is because, for MUSDB we have number of sources is equal to the number of microphones ($K = N$). 10-second segments of stems from the same track are interfered according to the generated Λ. Audio segments having only zeros were not included, resulting in 2450 10-second segments for each stem. Mean square error loss function, Adam optimiser with a learning rate of 0.01, batch size of 64 is used.

Evaluation data: 10-second segments of stems of the same track from MUSDB test data are interfered with using a random Λ. This is repeated so as to create evaluation data with high interference and with low interference. There are total 100 test tracks, each track having two 10-seconds chunks with low and high interference. Similarly, MUSDBR is obtained by adding time delays and room impulse responses to those MUSDB test data.

5.2. Results

MUSDB: We compare the performance of the proposed models with the state-of-the-art KAMIR algorithm [7]. Additionally, the reference SDR of the true source $s(t)$ and the interfered input $x(t)$ is also estimated. These results are summarized in Figure 4. It is apparent that any form of interference reduction improves the SDR. All algorithms are more successful in removing low interference than high interference. The CAE model performs at par or better than the KAMIR algorithm, except for vocals under low interference conditions. The t-UNet consistently performs better than both KAMIR and CAE in all evaluation conditions. Moreover, both CAE and t-UNet are much faster than the KAMIR and fastMIRA, which are both iterative algorithms. It took on average 660.4s for KAMIR, 2.4s for CAE and 2.19s for t-UNet, for evaluating 100 test tracks, each of 10 seconds, on a 12GB GPU under Keras environment.

An example of the spectrograms of the resulting vocal outputs are shown in Figure 5. It can be seen that interference components are present in KAMIR, but are completely removed by both CAE and t-UNet. Figure 6 shows boxplots corresponding to the difference of Frobenius norms of the actual Λ and the Λ predicted by the methods compared. It can be seen that the t-UNet model estimates

[1]To have consistent terminology with the MUSDB dataset, we henceforth term each song as *track*, and sources in the song as *stems*.

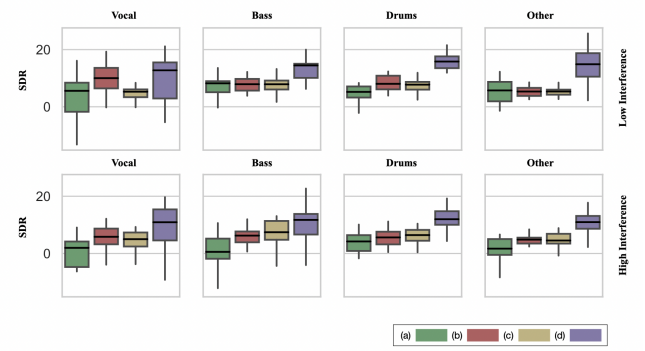

Figure 4: MUSDB: SDR comparison of two settings, low interference and high interference for different algorithms. (a) Reference SDR, (b) KAMIR, (c) CAE, and (d) t-UNet.

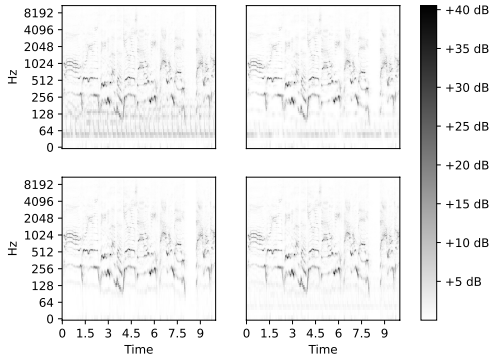

Figure 5: MUSDB: Spectrogram of a specific vocal example is shown. From top left clockwise: vocal with interference from bass, drums and others; KAMIR prediction; CAE prediction; and t-UNet prediction.

the interference matrix with high accuracy.

MUSDBR: In terms of SDR, all models perform poorer on MUSDBR data. The proposed t-UNet model handles mixtures with room responses and time delays reasonably well. Models trained with MUSDB were fine-tuned with MUSDBR and showed improved performance. On the other hand, the proposed CAE model and KAMIR does not perform as well as the t-UNet. Fig 7 gives the average SDR across the four stems.

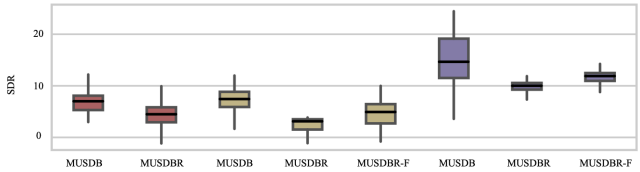

Figure 7: MUSDBR: SDR for different experiments for KAMIR, CAE, and t-UNet represented in Red, Yellow, and Magenta respectively. Suffix F represents models fine-tuned with MUSDBR.

Evaluation of music source separation performance: Interference reduction systems can be used as a pre-processing step before building music source separation (MSS) systems. Effective removal of interfering sources creates tracks of higher quality, resulting in better MSS models. We evaluate the MSS performance of the recently proposed Wave-U-Net [3] using interference reduction of the two methods proposed in this paper. Evaluation is performed on MUSDB, with MSS models trained separately on clean data, data with interference, and data pre-processed with the proposed methods.

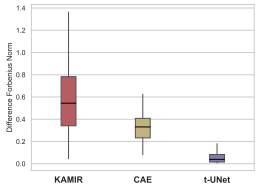

Figure 6: Difference of Frobenius norm of the true Λ with the predicted $\hat{\Lambda}$.

Table 1 summarizes the results. It can be seen that using training data having interference brings down the MSS performance. Pre-processing with CAE or t-UNet before building MSS models improves performance.

	Clean	Interference	CAE Cleaned	t-UNet cleaned
SDR	2.32	0.96	1.72	2.03

Table 1: Music source separation performance.

5.3. Discussion

The results presented in the previous section are on data on which interference, time delays, and the room responses has been created artificially. In real-world live recordings the interference could be more complex. We also evaluate CAE and t-UNet on a few recordings from a live classical music concert corpus. Since these are live recordings, the clean sources are not available, and hence BSS metrics like SDR cannot be estimated. Preliminary listening tests after interference reduction seem to indicate that the interference is not completely removed. Factors such as domain mismatch (trained on MUSDB, evaluated on classical music), and the limitation of the linear relationship in equation 3 are possible shortcomings.

6. CONCLUSION

In this paper, we examined two neural networks for reducing the interference in multi-track audio recordings. The convolutional autoencoder working in the time-frequency domain, and the truncated U-Net working in time domain showed promise as learned models for this task. Compared to the KAMIR algorithm, the proposed models demonstrate better computational complexity and improved source to distortion ratio, and can be used for effective pre-processing in music source separation. Future work will address the limitations of the proposed models including better generalization to new recording conditions.

7. REFERENCES

[1] Y. Luo and J. Yu, "Music source separation with band-split RNN," *arXiv preprint arXiv:2209.15174*, 2022.

[2] A. Défossez, "Hybrid spectrogram and waveform source separation," *arXiv preprint arXiv:2111.03600*, 2021.

[3] D. Stoller, S. Ewert, and S. Dixon, "Wave-u-net: A multi-scale neural network for end-to-end audio source separation," *arXiv preprint arXiv:1806.03185*, 2018.

[4] E. K. Kokkinis and J. Mourjopoulos, "Unmixing acoustic sources in real reverberant environments for close-microphone applications," *Journal of the Audio Engineering Society*, vol. 58, no. 11, pp. 907–922, 2010.

[5] E. K. Kokkinis, J. D. Reiss, and J. Mourjopoulos, "A wiener filter approach to microphone leakage reduction in close-microphone applications," *IEEE Trans. ASLP*, vol. 20, no. 3, pp. 767–779, 2011.

[6] E. Kokkinis, A. Tsilfidis, T. Kostis, and K. Karamitas, "A new DSP tool for drum leakage suppression," in *Audio Engineering Society Convention 135*. Audio Engineering Society, 2013.

[7] T. Prätzlich, R. M. Bittner, A. Liutkus, and M. Müller, "Kernel additive modeling for interference reduction in multi-channel music recordings," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 584–588.

[8] C. Févotte and J. Idier, "Algorithms for nonnegative matrix factorization with the β-divergence," *Neural Computation*, vol. 23, no. 9, pp. 2421–2456, 2011.

[9] A. Liutkus and R. Badeau, "Generalized wiener filtering with fractional power spectrograms," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 266–270.

[10] D. Di Carlo, A. Liutkus, and K. Déguemel, "Interference reduction on full-length live recordings," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 736–740.

[11] J. Reiss and C. Uhle, "Determined source separation for microphone recordings using iir filters," in *Audio Engineering Society Convention 129*. Audio Engineering Society, 2010.

[12] A. Clifford, J. D. Reiss, *et al.*, "Microphone interference reduction in live sound," in *Proceedings of the 14th International Conference on Digital Audio Effects (DAFx-11)*, 2011.

[13] F. Seipel and A. Lerch, "Multi-track crosstalk reduction using spectral subtraction," in *Audio Engineering Society Convention 144*. Audio Engineering Society, 2018.

[14] M. Zhao, D. Wang, Z. Zhang, and X. Zhang, "Music removal by convolutional denoising autoencoder in speech recognition," in *2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2015, pp. 338–341.

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

[16] H. Zhang, W. Li, J. Li, and F. Liu, "A novel speech enhancement method based on convolutional autoencoder and wavelet transform," *IEEE Access*, vol. 9, pp. 106 509–106 518, 2021.

[17] Y. Luo and N. Mesgarani, "Tasnet: time-domain audio separation network for real-time, single-channel speech separation," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 696–700.

[18] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, and R. Bittner, "Musdb18-HQ - an uncompressed version of MUSDB18," Aug. 2019. [Online]. Available: https://doi.org/10.5281/zenodo.3338373

[19] R. Scheibler, E. Bezzam, and I. Dokmanić, "Pyroomacoustics: A python package for audio room simulation and array processing algorithms," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 351–355.