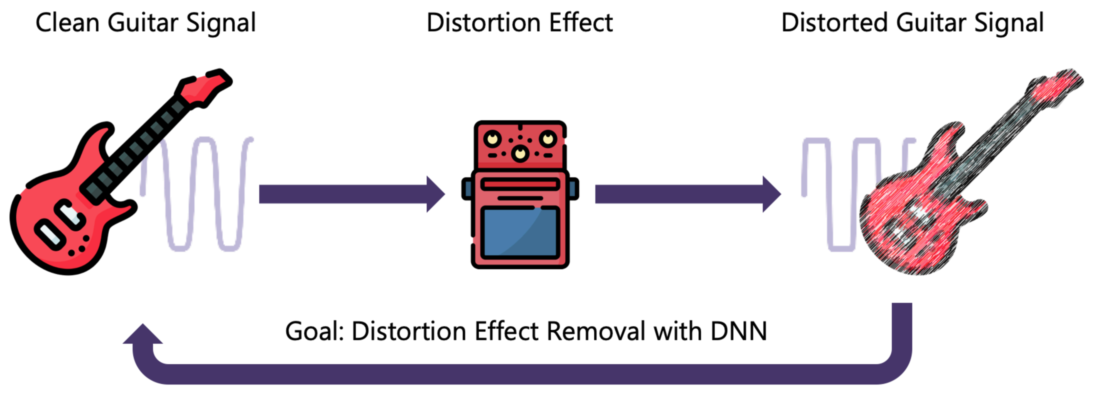
Distortion Audio Effects: Learning How to Recover the Clean Signal

Johannes Imort[♮], Giorgio Fabbro[♯], Marco A. Martínez Ramírez[♭], Stefan Uhlich[♯], Yuichiro Koyama[♭], Yuki Mitsufuji[♭]

[♮]RWTH Aachen University, Germany • [♯]Sony Europe B.V., Stuttgart, Germany • [♭]Sony Group Corporation, Tokyo, Japan

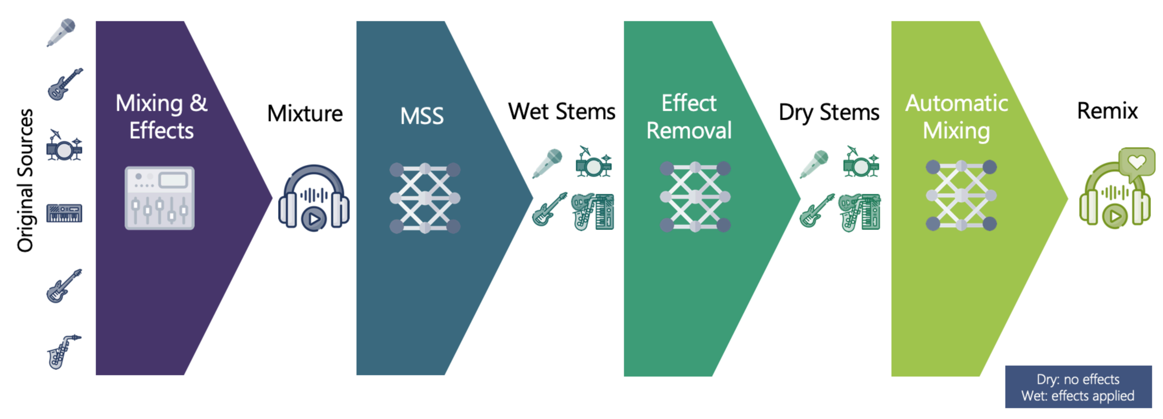
SONY

Introduction



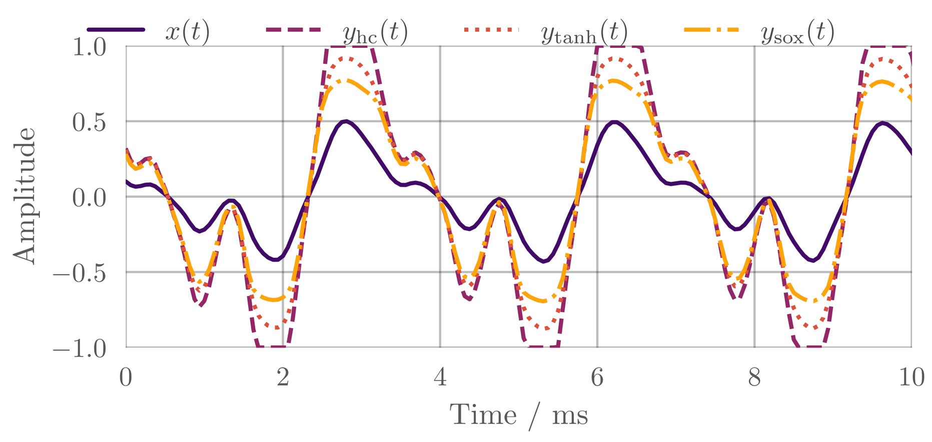
Research question: Can distortion effect removal be solved by DNNs designed for music source separation?

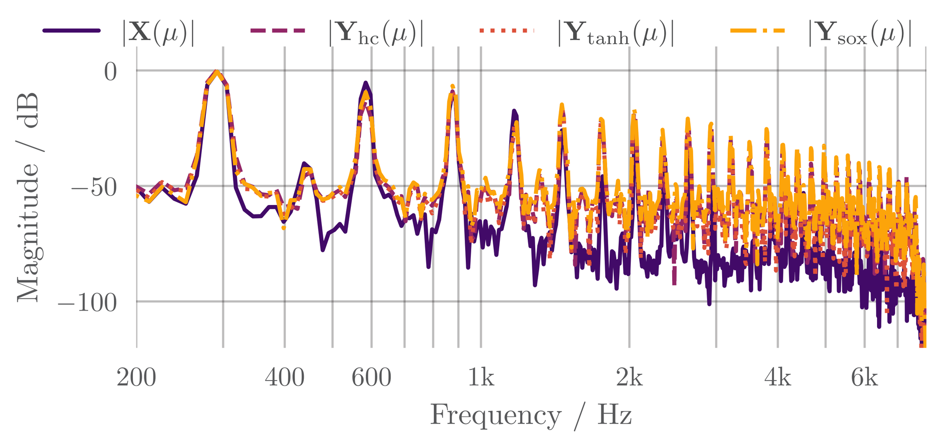
Application: Removing audio effects in music tracks is a meaningful step toward developing an automated remixing system.



Distortion Audio Effects

Hard-clipping	Simplified model; limits the amplitude when it exceeds a defined threshold (as typical for saturation in digital signal processing)	$y_{\mathrm{hc}}(x) = \begin{cases} x, & \text{if } \lvert 10^{\frac{\gamma}{20}} x \rvert \leq \theta_{\mathrm{c}} \\ \theta_{\mathrm{c}} \operatorname{sgn}(10^{\frac{\gamma}{20}} x) & \text{otherwise} \end{cases}$
Soft-clipping	Signal saturates gradually before reaching the fully saturated state (as typical for saturation in analog amplifiers)	$y_{\tanh}(x) = \tanh(10^{\frac{\gamma}{20}} x)$
SoX overdrive [1]	One example of a more complex distortion algorithm; mixes the wet and the dry signal	





Consequence: Introduction of harmonics and intermodulation distortion

Data

Clean Electric Guitar (CEG)	Polyphonic clean electric guitar samples collected from loop packages and YouTube, ~1.7h
SignalTrain [7]	Various musical audio content mixed with synthetic audio (e.g., sweeps, noise), ~24h

Processing: SoX overdrive [1] on (**Task A (CEG)**) and hard-clipping (**Task B (CEG)/C (SignalTrain)**) with uniformly sampled gain levels in the range of [20, 50] dB.

Methods

Demucs V2* [2]	Originally proposed for source separation; autoencoder architecture composed of a convolutional encoder, a BLSTM, and a convolutional decoder, linked with skip connections
Wave-U-Net* [3]	Originally proposed for source separation; U-Net for raw audio
CRAFx [4]	Originally proposed for audio effect modelling; autoencoder architecture composed of a learnable filterbank, a BLSTM, and learnable nonlinearities
Open Unmix (UMX) [5]	Originally proposed for source separation; BLSTM that operates on STFT magnitude input features; applies a learned magnitude mask to the input; reuses original phase for reconstruction
A-SPADE [6]	Sparsity-based iterative algorithm; serves as a state-of-the-art baseline

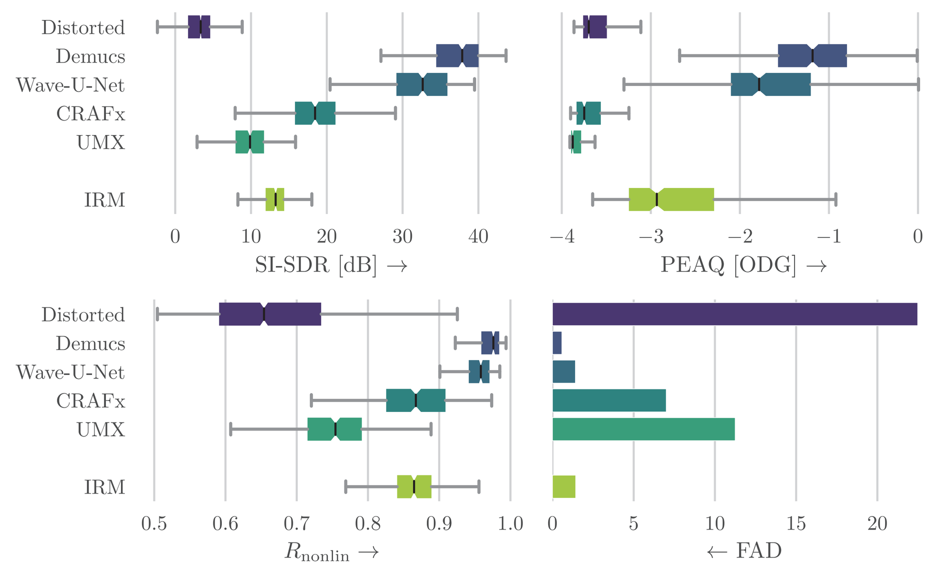
* Number of layers reduced

References

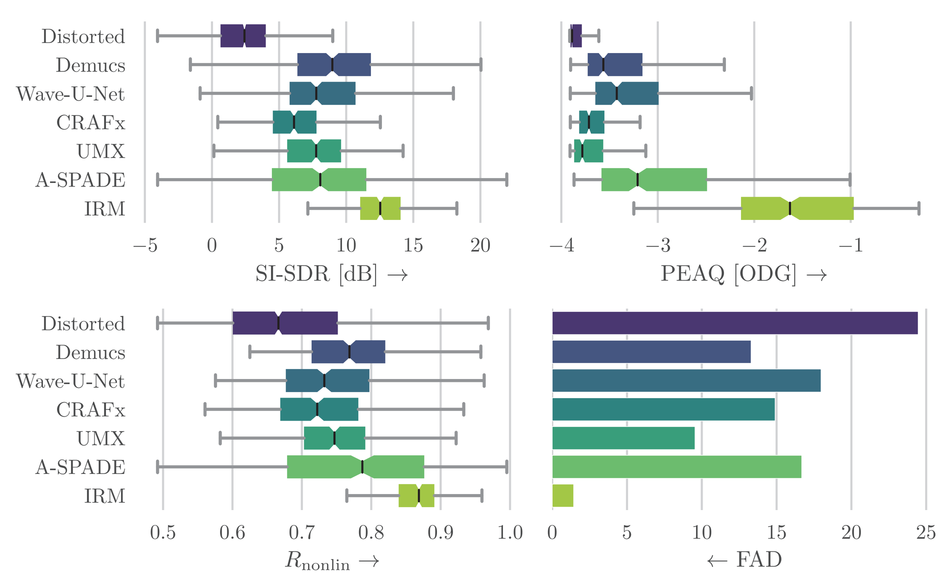
[1] C. Bagwell, et al., "Sound eXchange (SoX)," Feb. 2015, https://sox.sourceforge.net/
[2] A. Défossez, N. Usunier, L. Bottou, and F. Bach, "Music Source Separation in the Waveform Domain," arXiv:1911.13254 [cs, eess, stat], Apr. 2021.
[3] D. Stoller, S. Ewert, and S. Dixon, "Wave-U-Net: A Multi-Scale Neural Network for End-to-End Audio Source Separation," in Proceedings of the 19th ISMIR Conference, Paris, France, September 23-27, 2018.
[4] M. A. Martínez-Ramírez, E. Benetos, and J. Reiss, "Deep Learning for Black-Box Modeling of Audio Effects," Applied Sciences, vol. 10, p. 638, Jan. 2020.
[5] F.-R. Stöter, S. Uhlich, A. Liutkus, and Y. Mitsufuji, "Open-Unmix - A Reference Implementation for Music Source Separation," Journal of Open Source Software, vol. 4, no. 41, p. 1667, Sep. 2019.
[6] S. Kitic, N. Bertin, and R. Gribonval, "Sparsity and Cosparsity for Audio Declipping: A Flexible Non-convex Approach," in Latent Variable Analysis and Signal Separation, E. Vincent, A. Yeredor, Z. Koldovský, and P. Tichavský, Eds. Cham: Springer International Publishing, 2015, pp. 243–250.
[7] S. H. Hawley, B. Colburn, and S. I. Mimilakis, "SignalTrain: Profiling Audio Compressors with Deep Neural Networks," in AES 147th Convention, May 2019.
[8] P. Sobot, "pedalboard V0.3.10." 2021, Spotify AB, https://github.com/spotify/pedalboard

Results

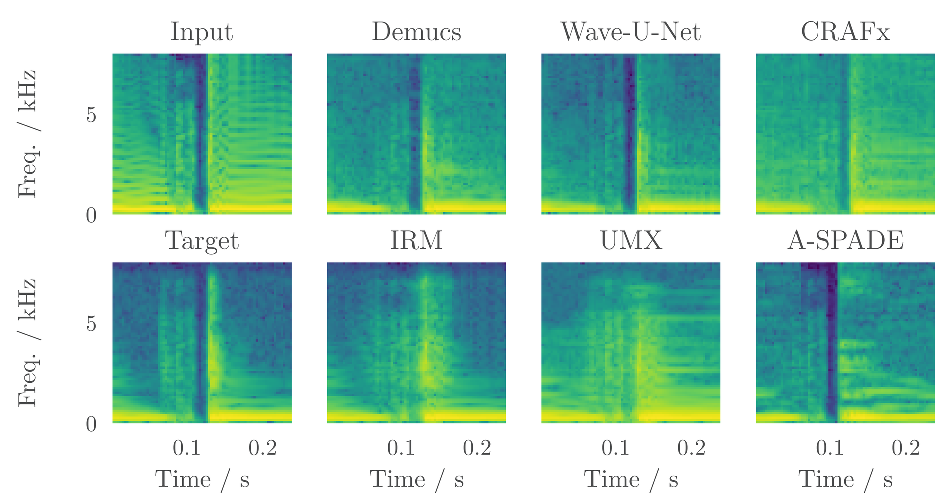
Task A (CEG): De-Overdrive Guitar



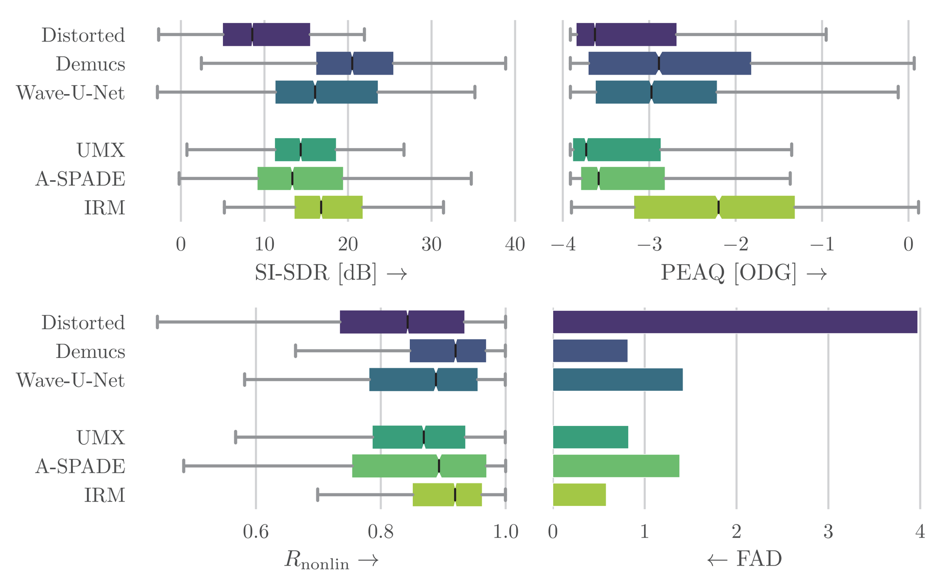
Task B (CEG): Declip Guitar



Example from test set:



Task C (SignalTrain): Declip Audio



Conclusion

- Distortion effect removal can be efficiently solved with DNNs designed for source separation, especially when the distortion algorithm to be removed blends the distorted sound with the original one
- The metrics under evaluation prove beneficial for evaluating effect removal systems
- Future work: simulate more realistic effects on larger dataset (e.g., use Pedalboard [8])