Modeling nonlinear audio effects with end-to-end deep neural networks

Marco A. Martínez Ramírez, Joshua D. Reiss

May 16, 2019

Deep learning architectures for audio processing

We research deep learning architectures for audio processing in the context of audio effects modeling.

Deep learning architectures for audio processing

We research deep learning architectures for audio processing in the context of audio effects modeling.

Audio effects modeling is the process of emulating the sound of an analog reference device.

Deep learning architectures for audio processing

We research deep learning architectures for audio processing in the context of audio effects modeling.

Audio effects modeling is the process of emulating the sound of an analog reference device.

We aim to find a **general-purpose** deep learning architecture for generic black-box modeling of nonlinear and linear audio effects.

Nonlinear audio effects

Nonlinear audio effects are widely used by musicians and sound engineers.

Nonlinear audio effects

Nonlinear audio effects are widely used by musicians and sound engineers.

Mainly used for aesthetic reasons and are usually applied to electric musical instruments.

Nonlinear audio effects

Nonlinear audio effects are widely used by musicians and sound engineers.

Mainly used for aesthetic reasons and are usually applied to electric musical instruments.

The nonlinear behaviour of circuit components leads to **amplitude and harmonic distortion**.

Nonlinear audio effects

Nonlinear audio effects are widely used by musicians and sound engineers.

Mainly used for aesthetic reasons and are usually applied to electric musical instruments.

The nonlinear behaviour of circuit components leads to **amplitude and harmonic distortion**.

Most common nonlinear effects:

- *distortion*, *overdrive*, *amplifiers*

Nonlinear audio effects modeling

Most existing methods for nonlinear modeling are often either **simplified** or **optimized** to a very specific circuit.

Nonlinear audio effects modeling

Most existing methods for nonlinear modeling are often either **simplified** or **optimized** to a very specific circuit.

Difficult to generalize the methods among different audio effects.

Nonlinear audio effects modeling

Most existing methods for nonlinear modeling are often either **simplified** or **optimized** to a very specific circuit.

Difficult to generalize the methods among different audio effects.

Accentuated when we consider that each effect unit also contains components other than the nonlinearity.

Aim

Given an **arbitrary** nonlinear audio effect, our task is to train a deep neural network to learn and apply the specific transformation.

Aim

Given an **arbitrary** nonlinear audio effect, our task is to train a deep neural network to learn and apply the specific transformation.

Using an end-to-end architecture, where raw audio is both the input and the output of the system.

Aim

Given an **arbitrary** nonlinear audio effect, our task is to train a deep neural network to learn and apply the specific transformation.

Using an end-to-end architecture, where raw audio is both the input and the output of the system.

We show the model performing nonlinear modeling:

- *distortion*
- *overdrive*
- *amplifiers*
- *combinations of linear and nonlinear audio effects*

Dataset

The audio is obtained from **IDMT-SMT-Audio-Effects**.

Dataset

The audio is obtained from **IDMT-SMT-Audio-Effects**.

Individual 2-second notes of guitars and bass.

Dataset

The audio is obtained from **IDMT-SMT-Audio-Effects**.

Individual 2-second notes of guitars and bass.

We use unprocessed and processed recordings with *distortion*, *overdrive* and *amplifier emulation (EQ)*.

Dataset

The audio is obtained from **IDMT-SMT-Audio-Effects**.

Individual 2-second notes of guitars and bass.

We use unprocessed and processed recordings with *distortion*, *overdrive* and *amplifier emulation (EQ)*.

We also apply a custom audio effects chain (FxChain):

- *lowshelf filter* (gain=+20dB)
- *highshelf filter* (gain=-20dB)
- *overdrive* (gain=+30dB)

Dataset

The audio is obtained from **IDMT-SMT-Audio-Effects**.

Individual 2-second notes of guitars and bass.

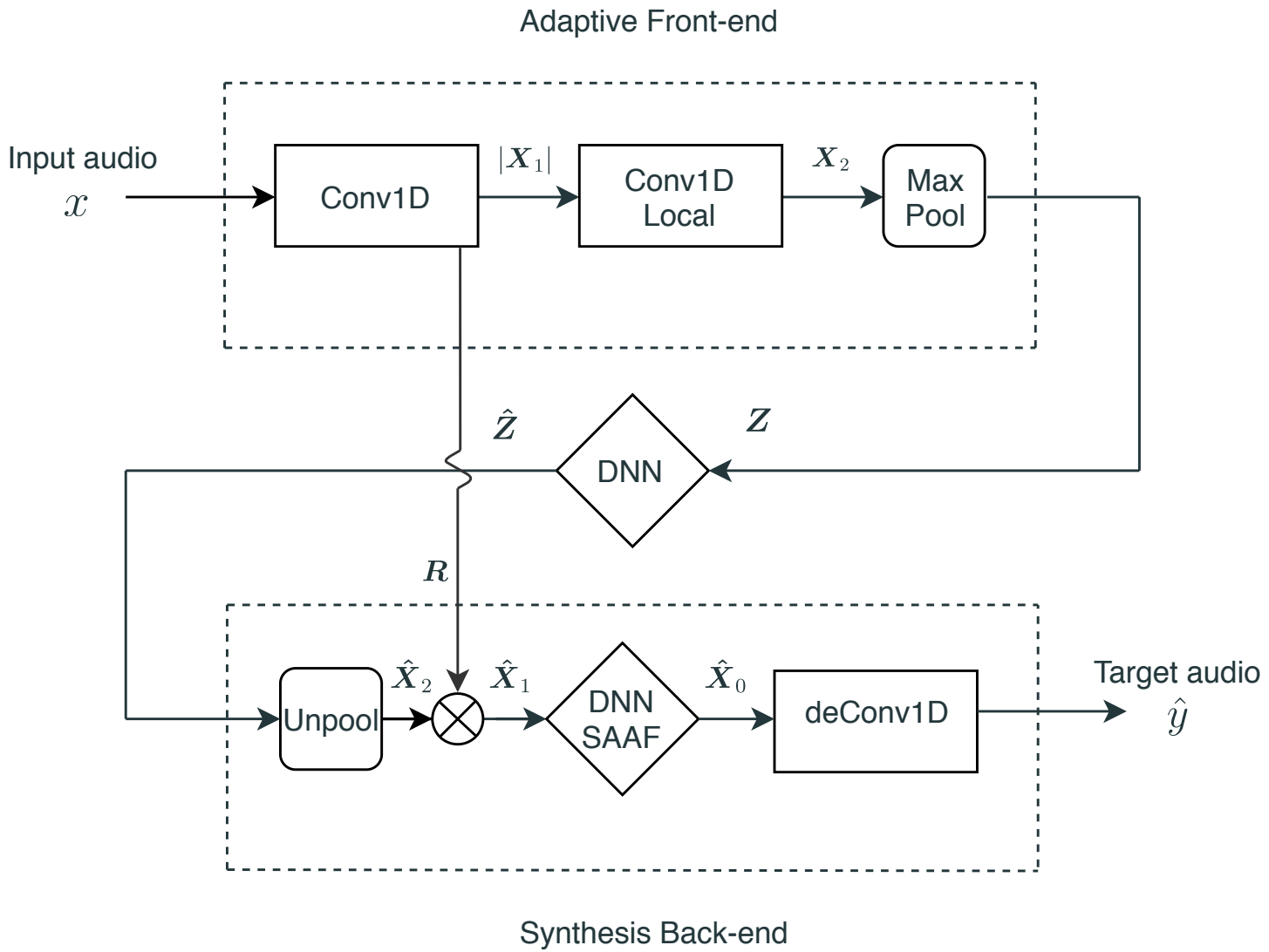
We use unprocessed and processed recordings with *distortion*, *overdrive* and *amplifier emulation (EQ)*.

We also apply a custom audio effects chain (FxChain):

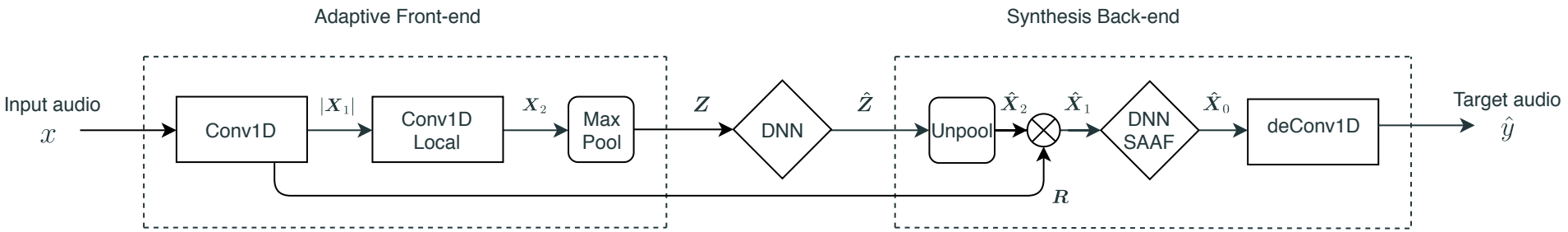
- *lowshelf filter* (gain=+20dB)
- *highshelf filter* (gain=-20dB)
- *overdrive* (gain=+30dB)

We use **624** raw and distorted notes for each audio effect setting.

Model


Adaptive Front-end
Input audio
x
Conv1D
$|X_1|$
Conv1D Local
X_2
Max Pool
$\hat{Z}$
DNN
Z
R
Unpool
$\hat{X}_2$
$\hat{X}_1$
DNN SAAF
$\hat{X}_0$
deConv1D
Target audio
$\hat{y}$
Synthesis Back-end

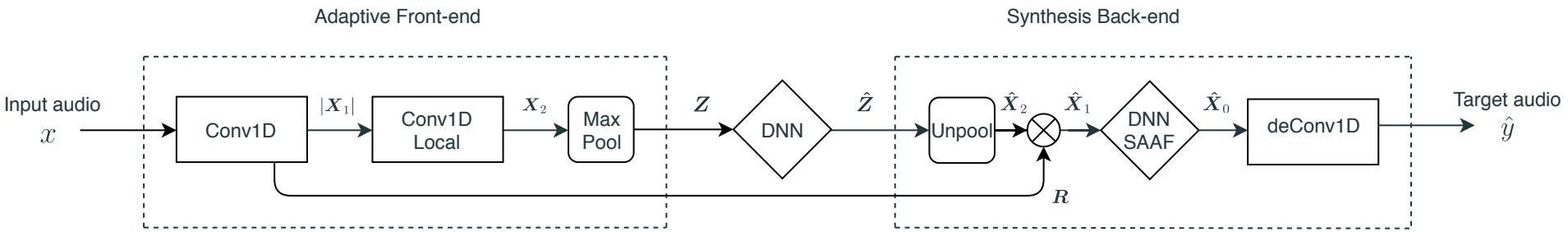
Training



The training of the model is performed in two steps.

- The **first** step is to train only the convolutional layers for an **unsupervised learning** task.

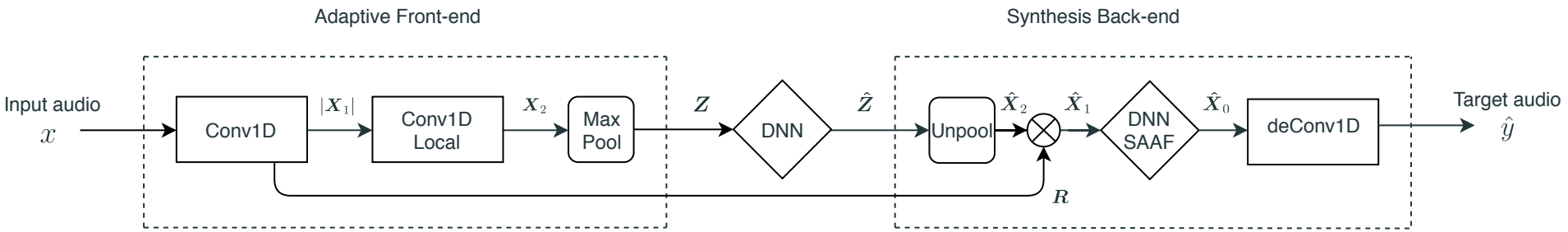
Training



The training of the model is performed in two steps.

- The **first** step is to train only the convolutional layers for an **unsupervised learning** task.
- The **second** step consists of an end-to-end **supervised learning** task based on a given nonlinear target.

Training

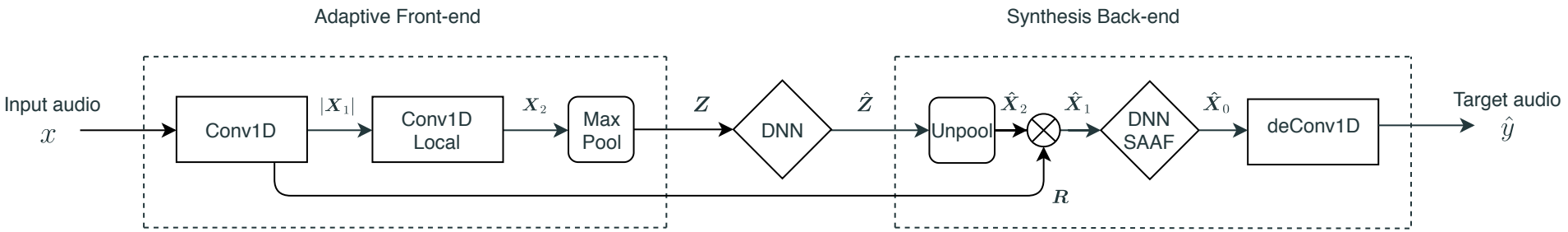


The training of the model is performed in two steps.

- The **first** step is to train only the convolutional layers for an **unsupervised learning** task.
- The **second** step consists of an end-to-end **supervised learning** task based on a given nonlinear target.

* The loss function to be minimized is the mean absolute error.

Training



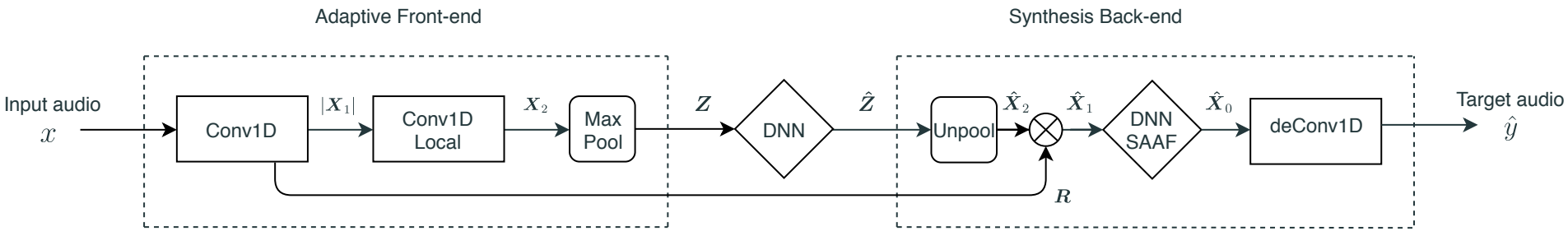
The training of the model is performed in two steps.

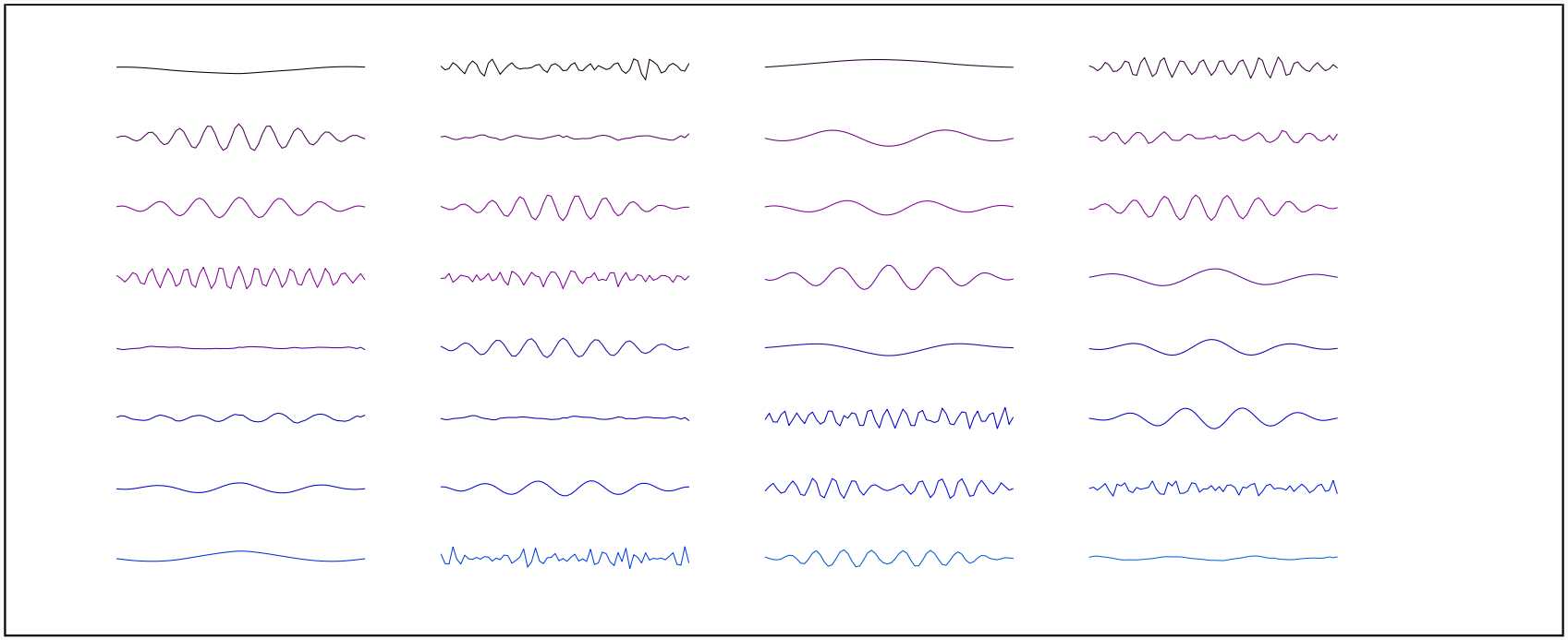
- The **first** step is to train only the convolutional layers for an **unsupervised learning** task.
- The **second** step consists of an end-to-end **supervised learning** task based on a given nonlinear target.

* The loss function to be minimized is the mean absolute error.

* Input and Target are 1024 samples with hop size of 64 samples.

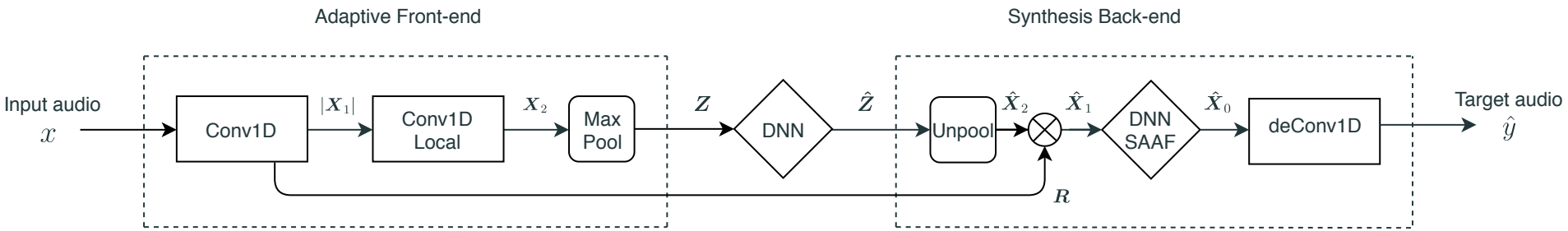
Results - unsupervised learning

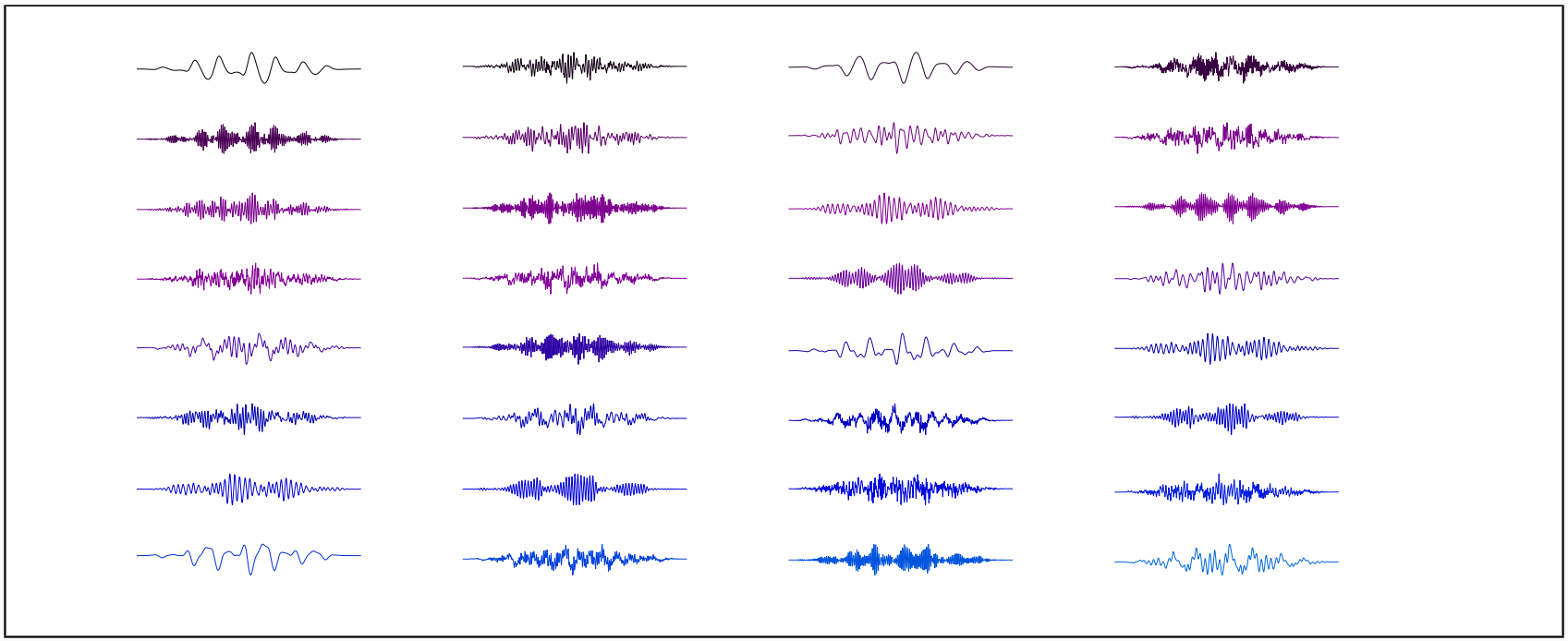




Conv1D filters

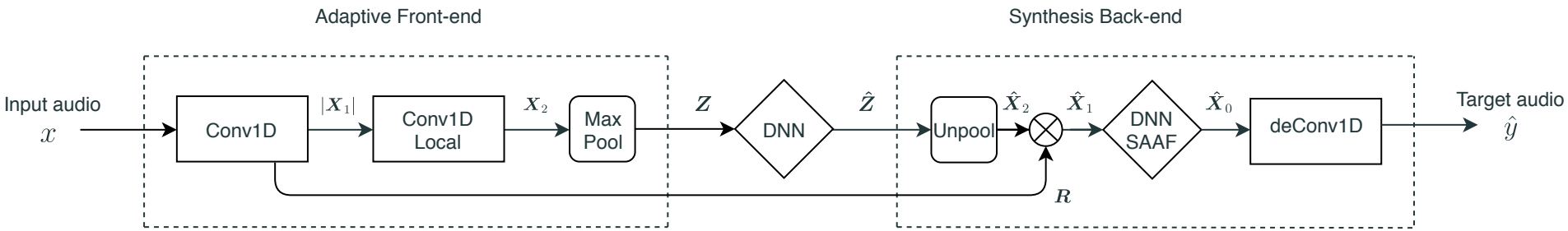
Results - unsupervised learning

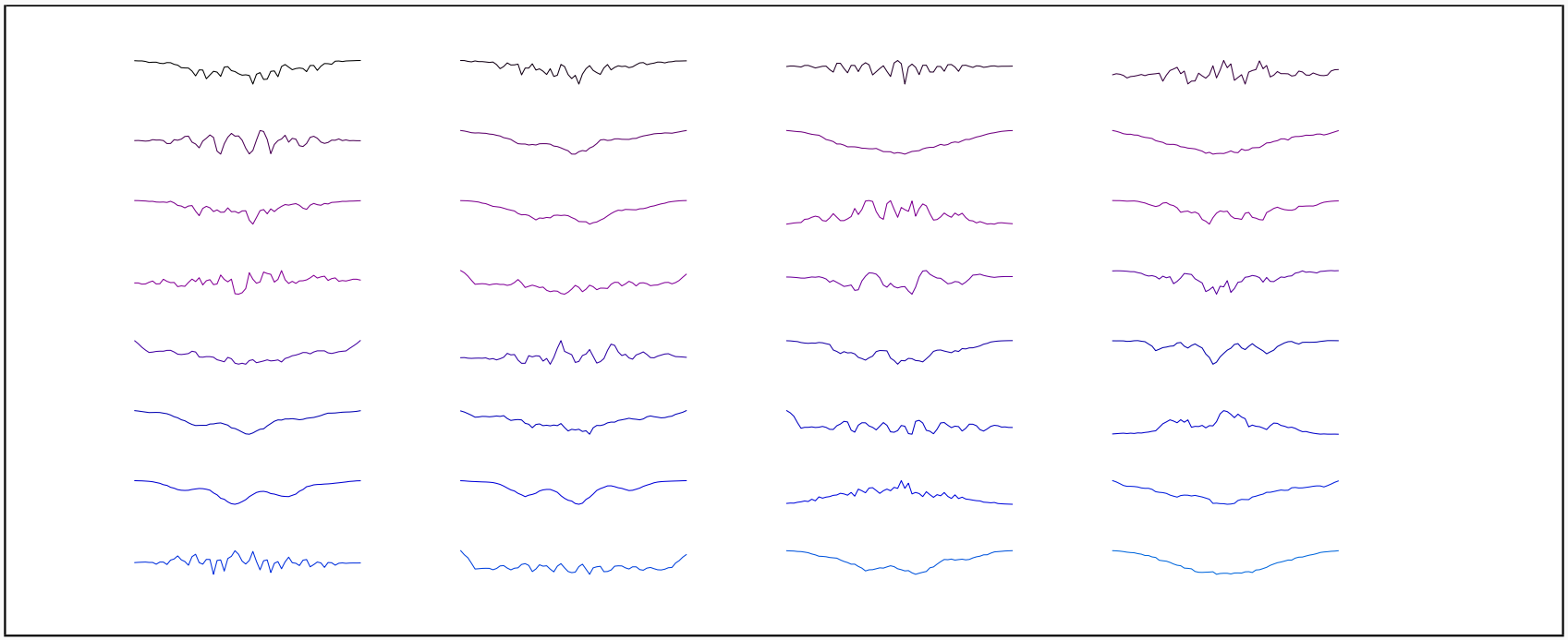




X_1

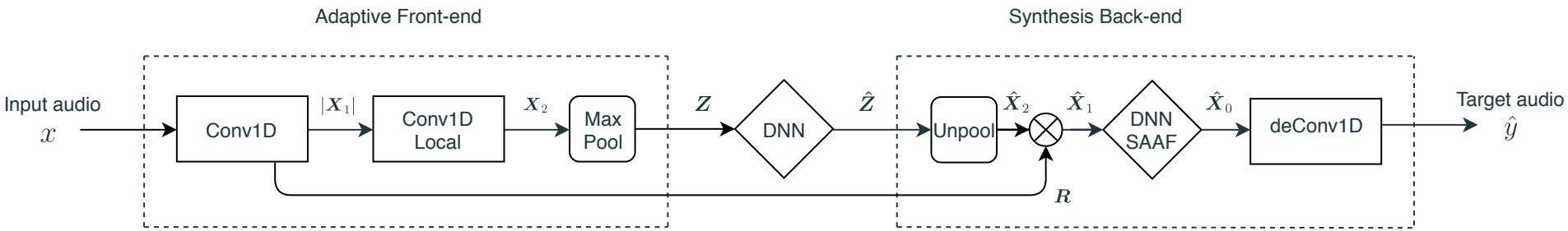
Results - unsupervised learning

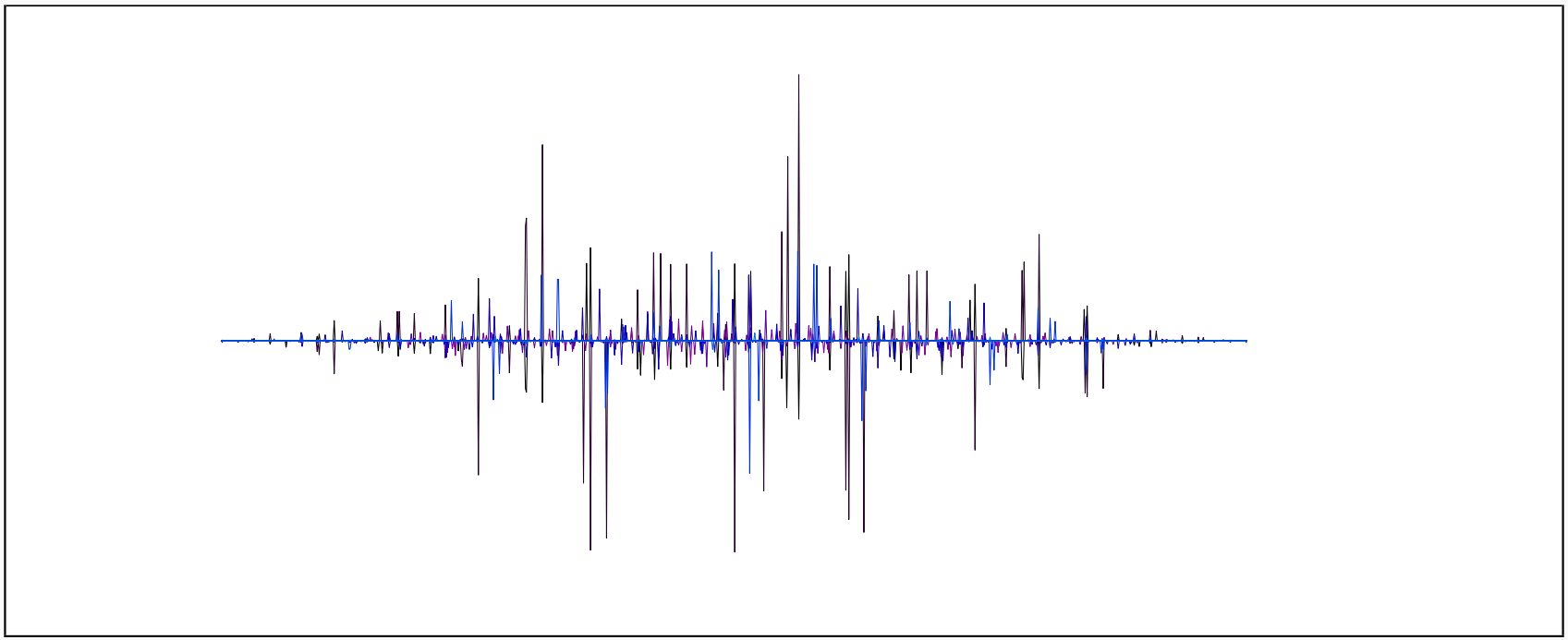




Z

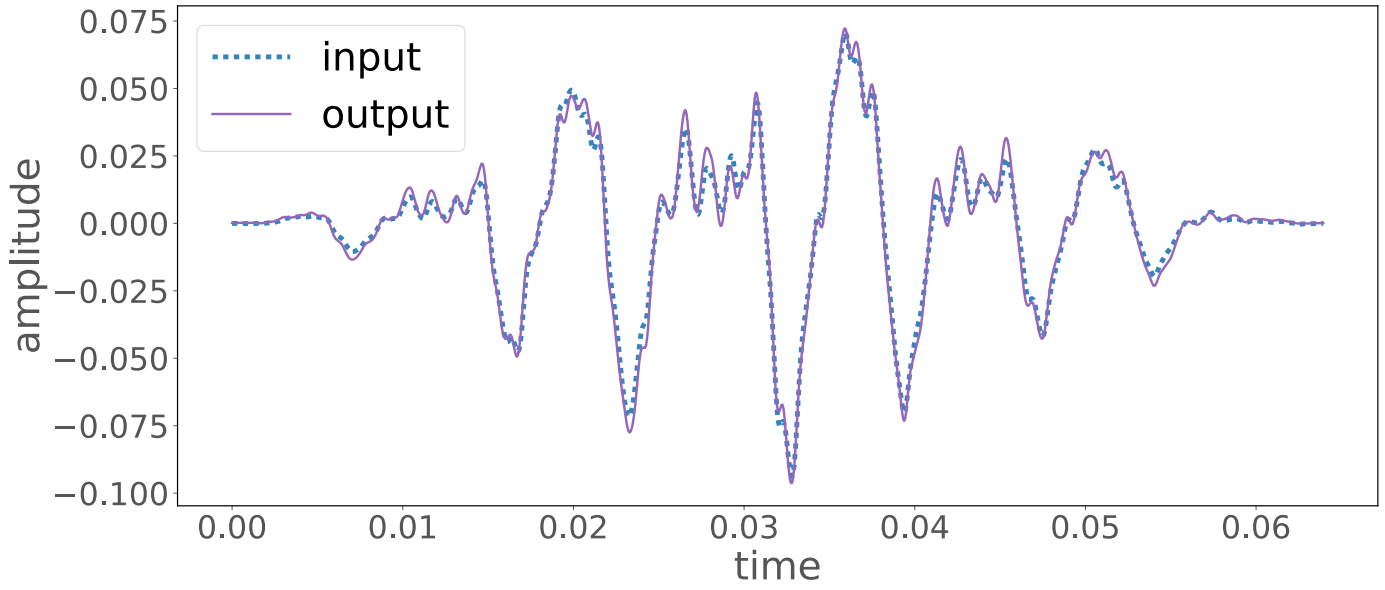
Results - unsupervised learning

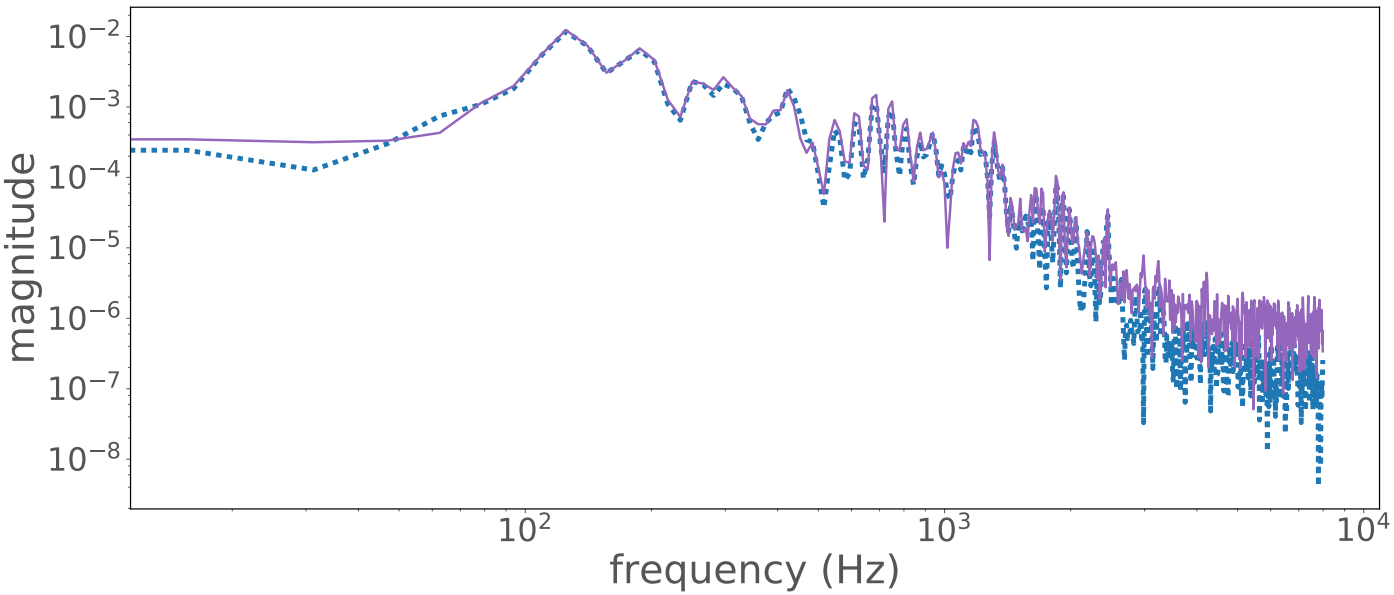




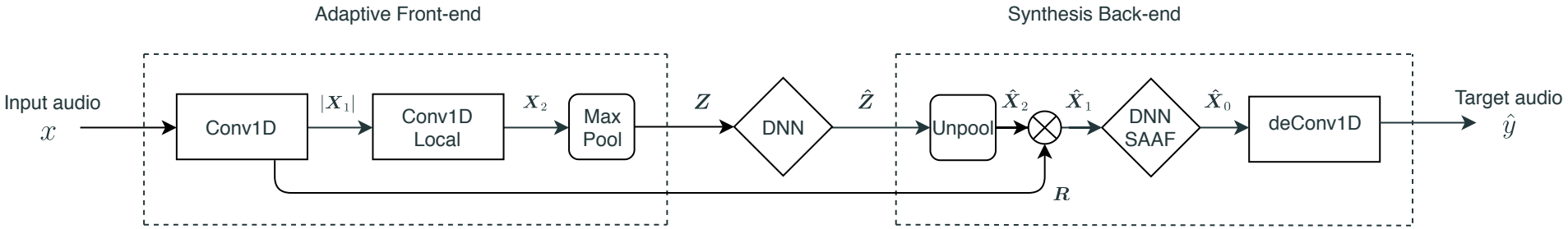
$\hat{X}_1$

Results - unsupervised learning



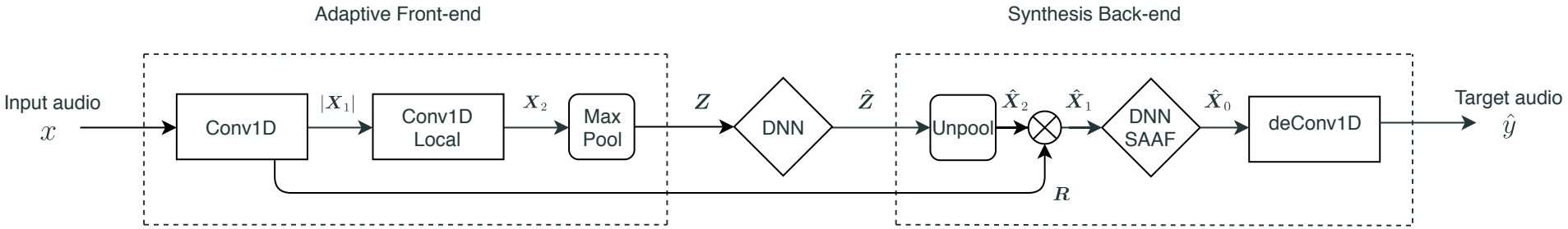


Model



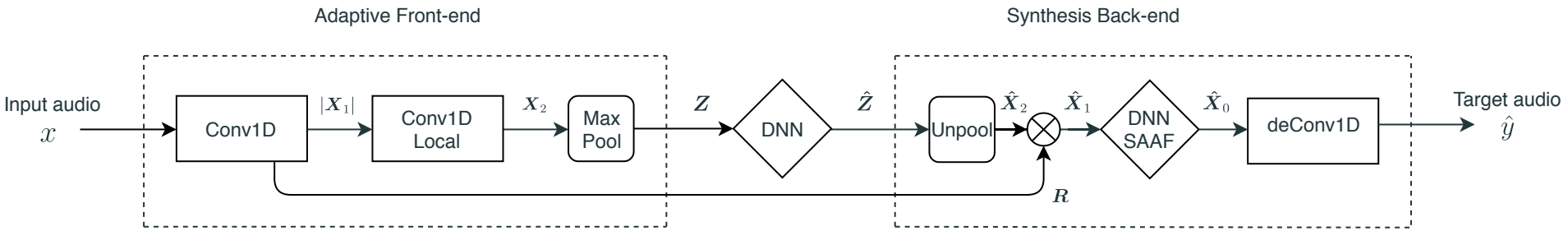
- **Latent-space DNN**

Model



- **Latent-space DNN**
  - Locally connected dense layer.
  - Fully connected dense layer.

Model

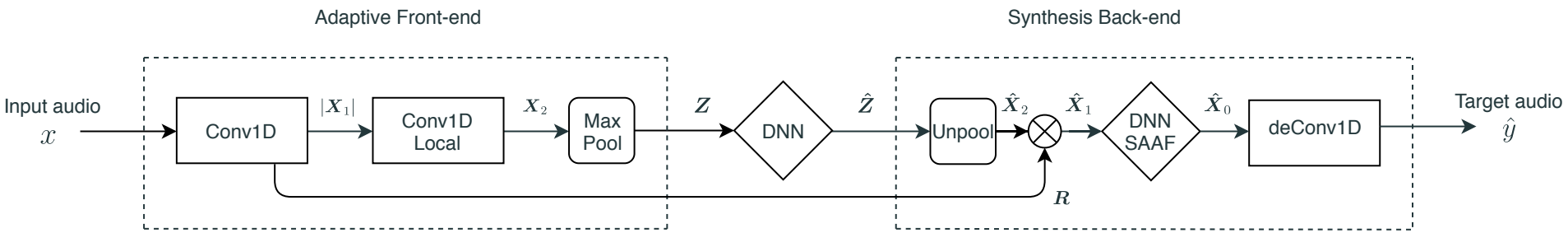


- **Latent-space DNN**

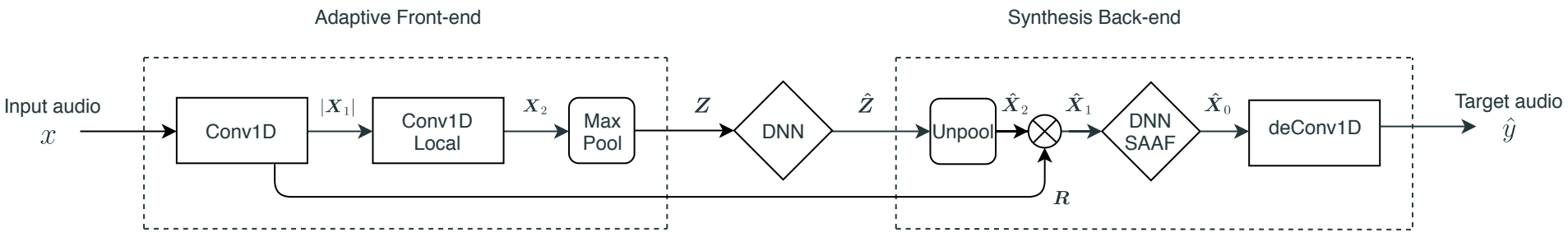
  - Locally connected dense layer.
  - Fully connected dense layer.

* Martínez Ramírez M. A. and Reiss J. D., *"End-to-end equalization with convolutional neural networks"*, 21st International Conference on Digital Audio Effects (DAFx-18).

DNN-SAAF

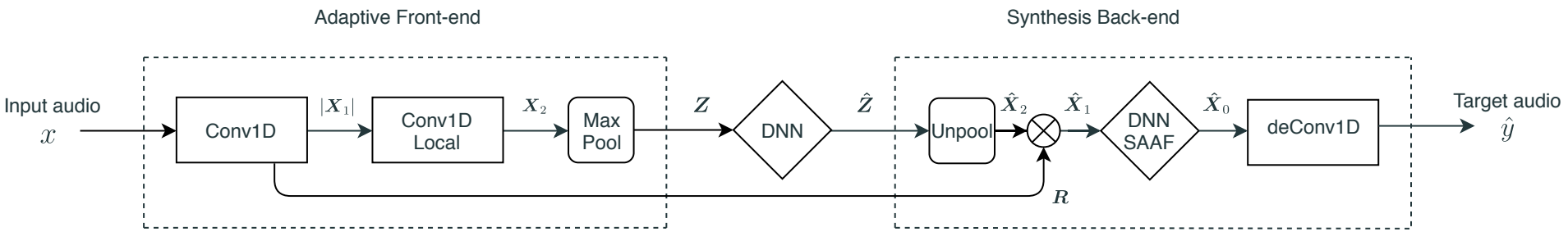

Adaptive Front-end
Synthesis Back-end
Input audio
x
Conv1D
$|X_1|$
Conv1D Local
X_2
Max Pool
Z
DNN
$\hat{Z}$
Unpool
$\hat{X}_2$
$\hat{X}_1$
DNN SAAF
$\hat{X}_0$
deConv1D
Target audio
$\hat{y}$
R

DNN-SAAF



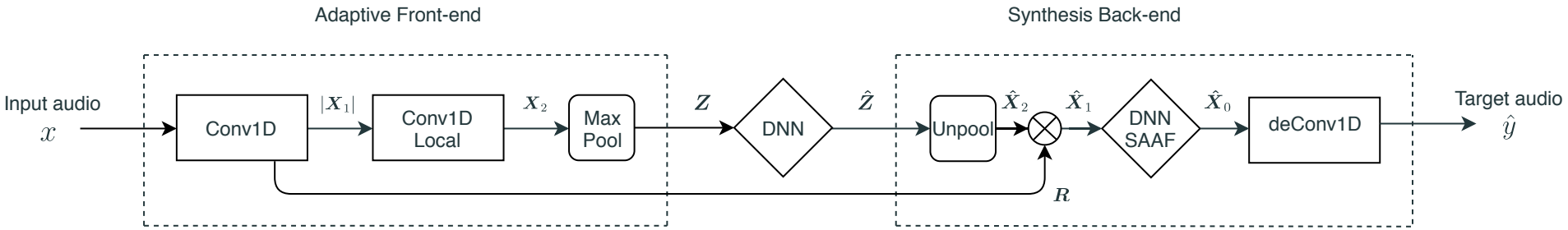
- **Smooth adaptive activation functions (SAAF)**

DNN-SAAF

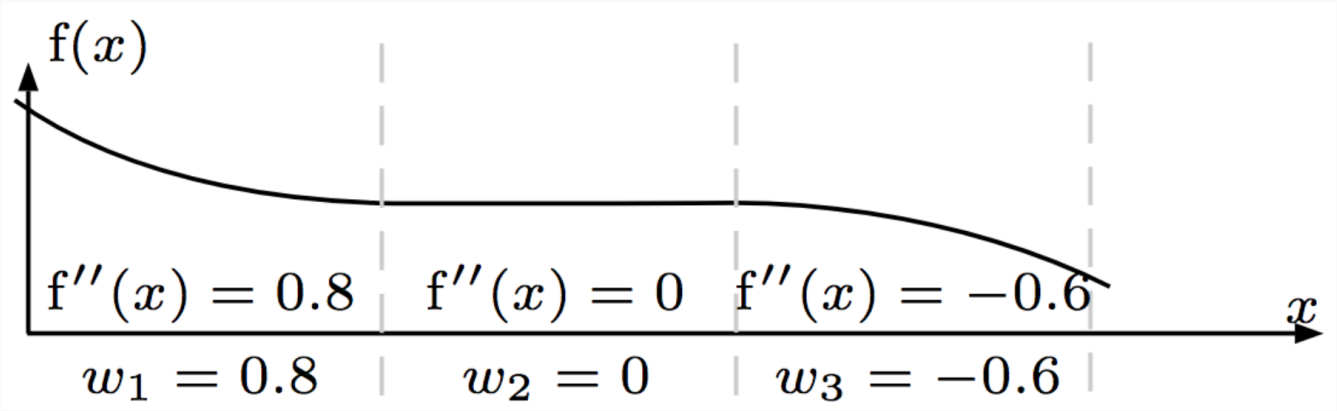


- **Smooth adaptive activation functions (SAAF)**
  - Locally connected.
  - Piecewise second order polynomials which can approximate any continuous function
  - Regularized under a *Lipschitz* constant to ensure smoothness.

DNN-SAAF



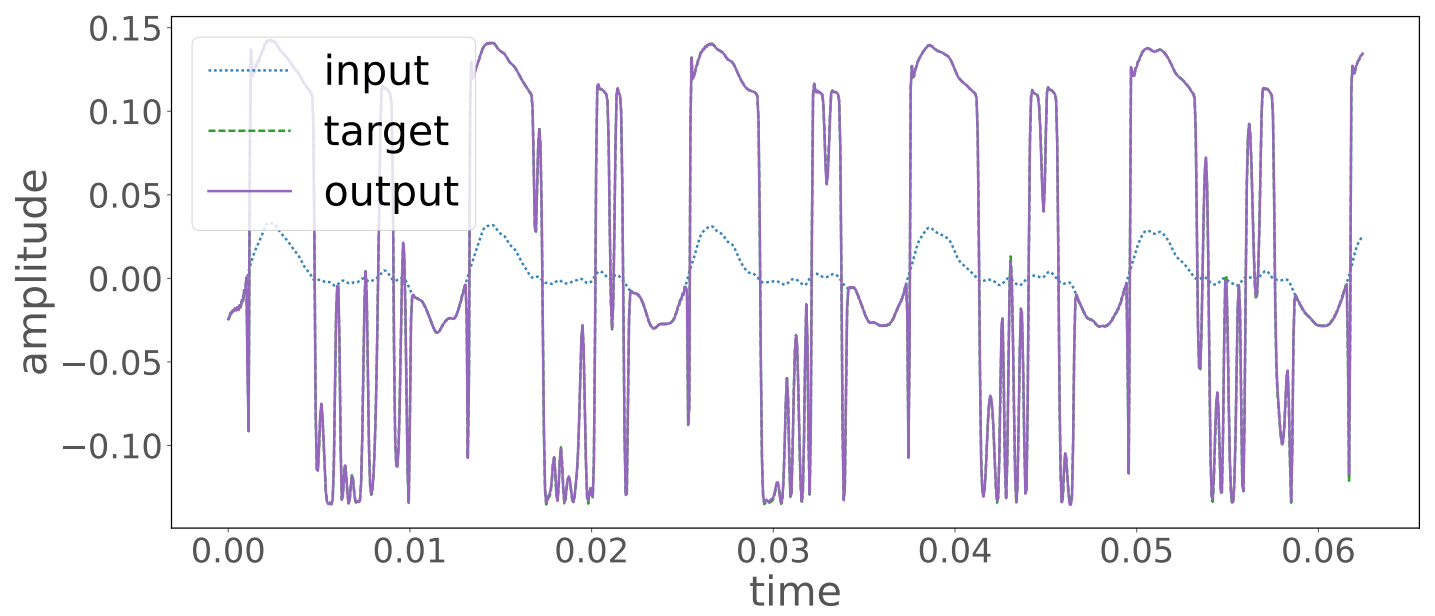
- **Smooth adaptive activation functions (SAAF)**

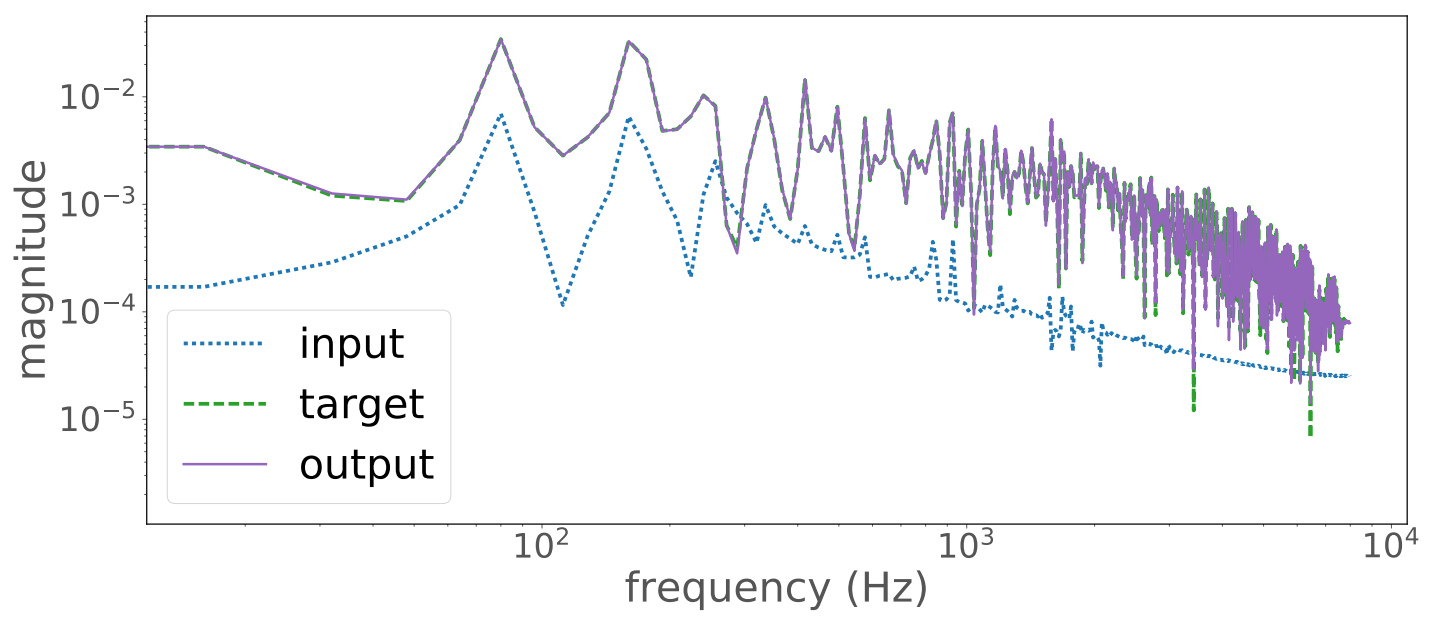


[1]

[1] Le Hou et al., Convnets with smooth adaptive activation functions for regression, in Artificial Intelligence and Statistics, 2017.

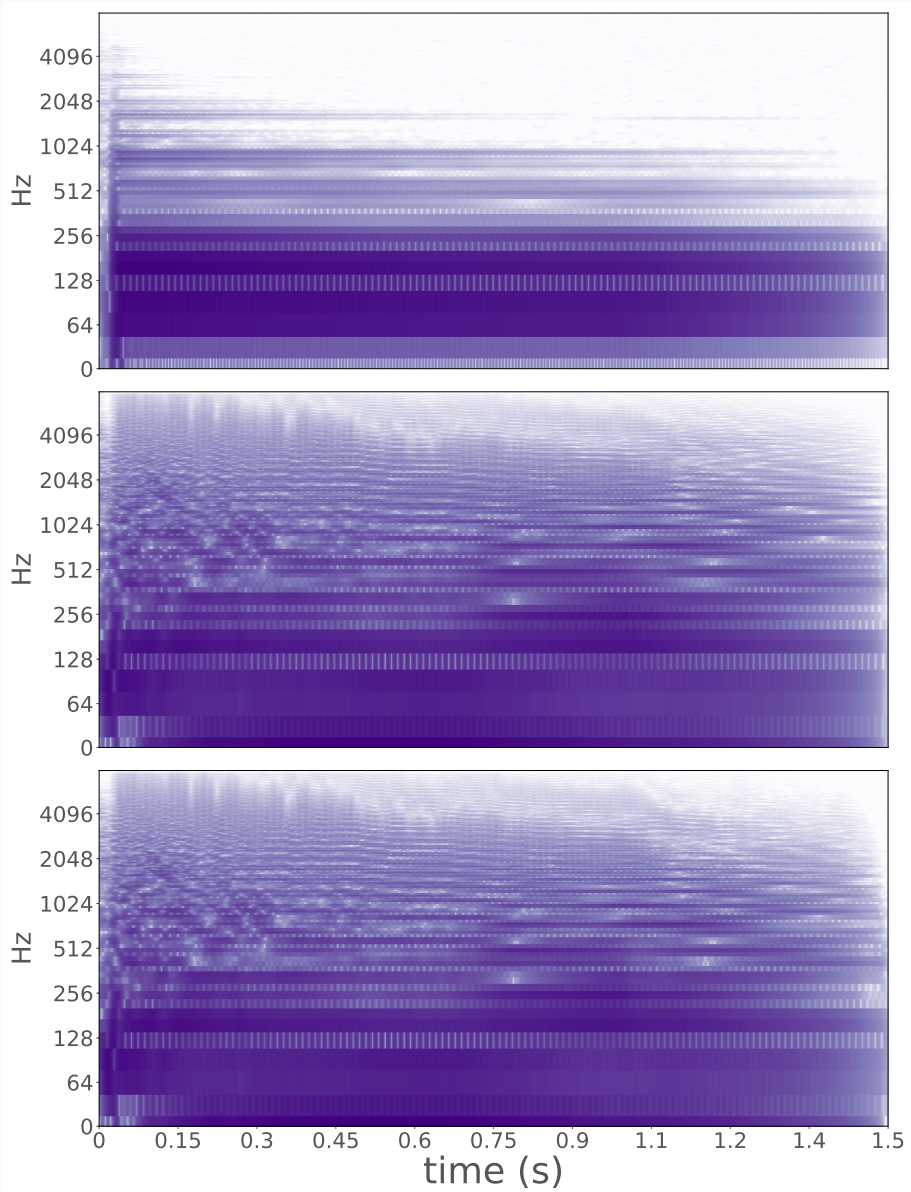
Results - distortion - bass guitar





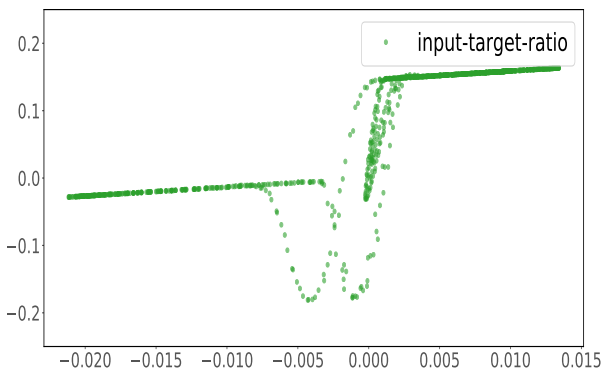
Results - distortion - bass guitar

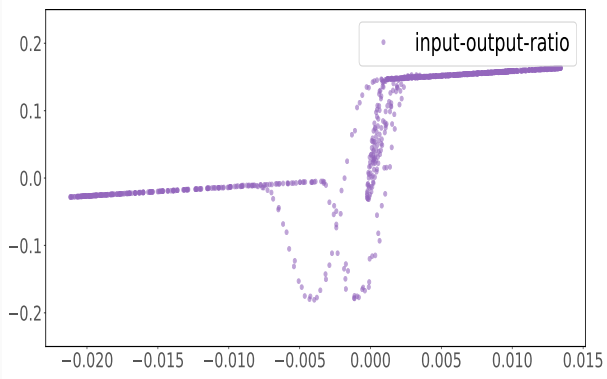
- input
- target
- output



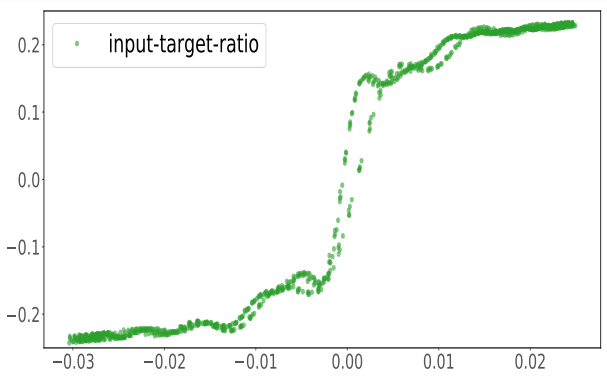
Results - distortion - waveshaping curves

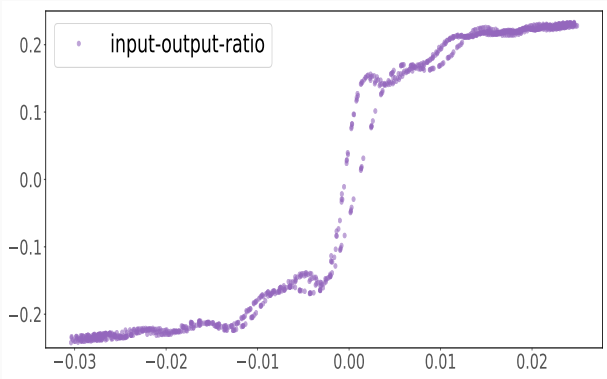
- 1st setting



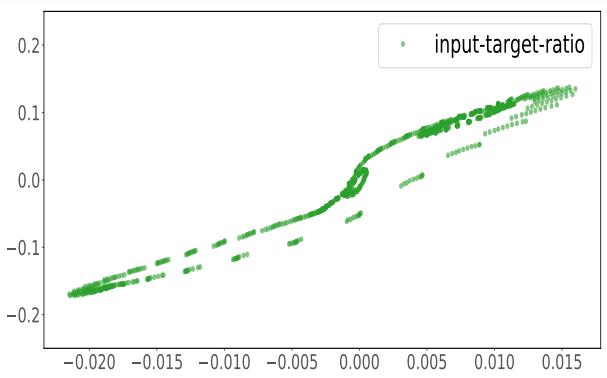


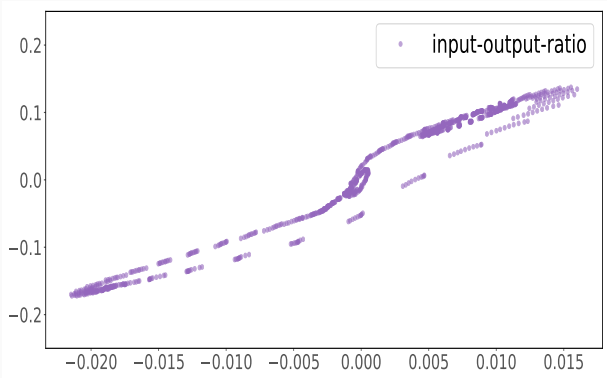
- 2nd setting



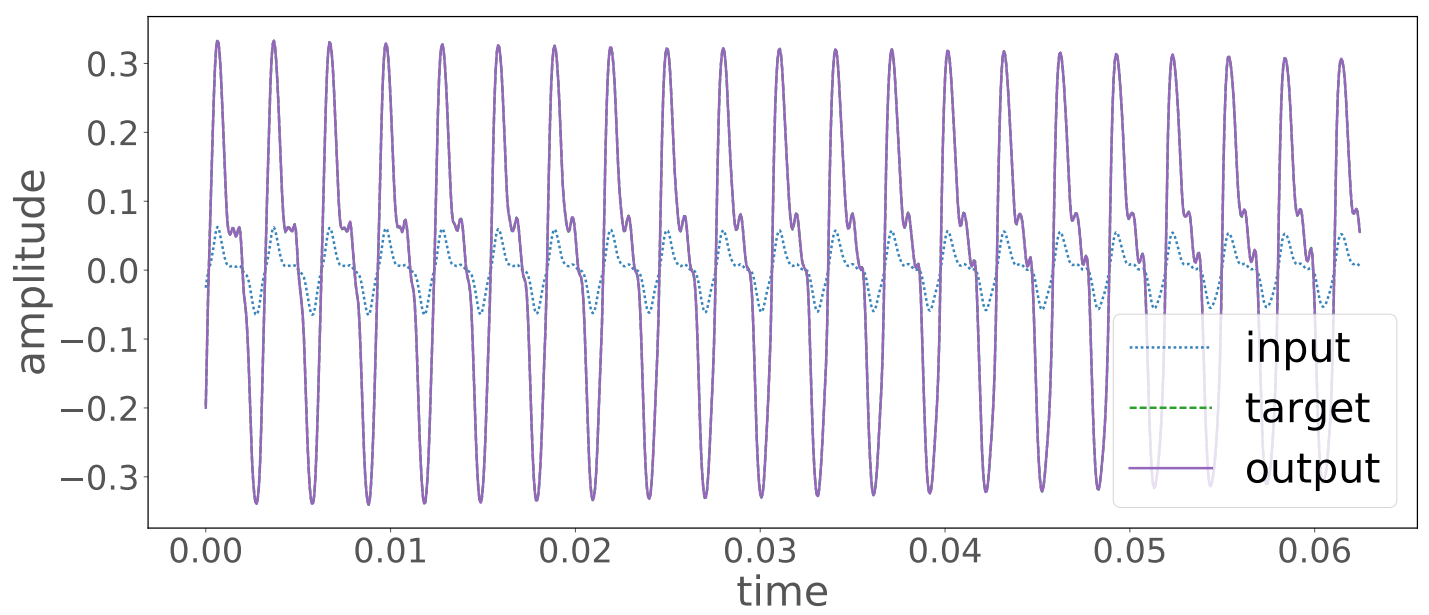


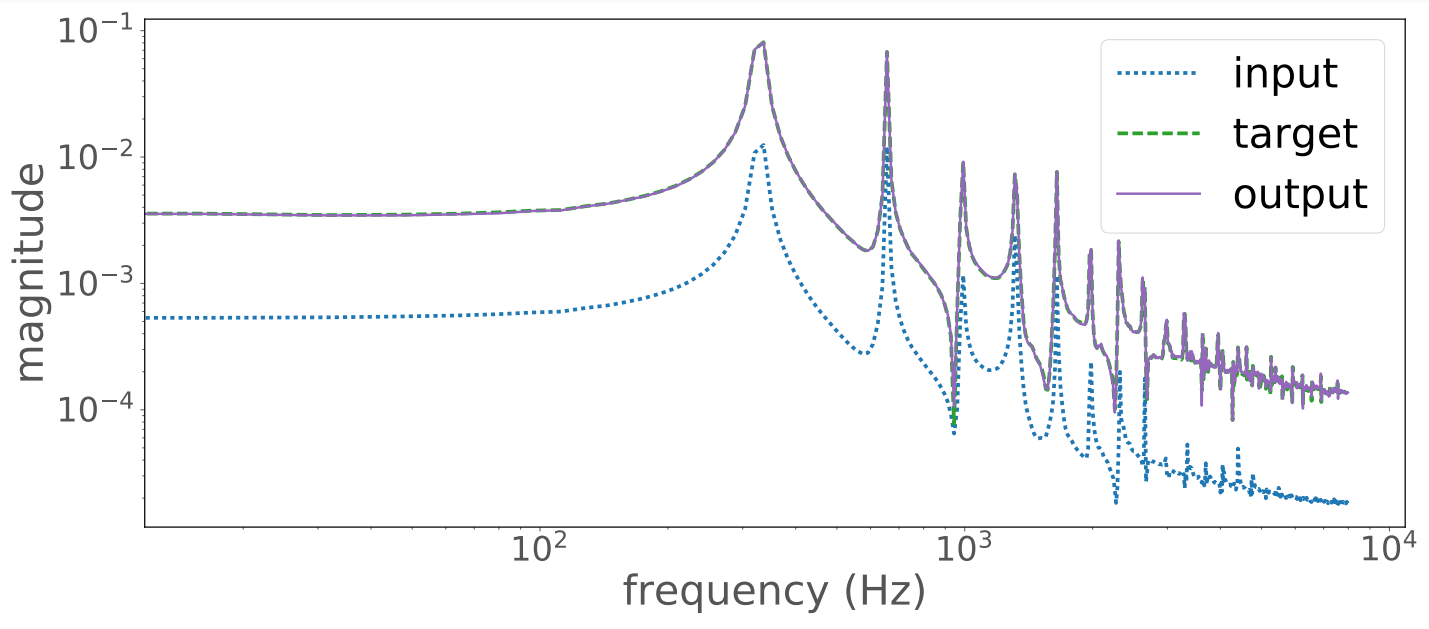
- 3rd setting





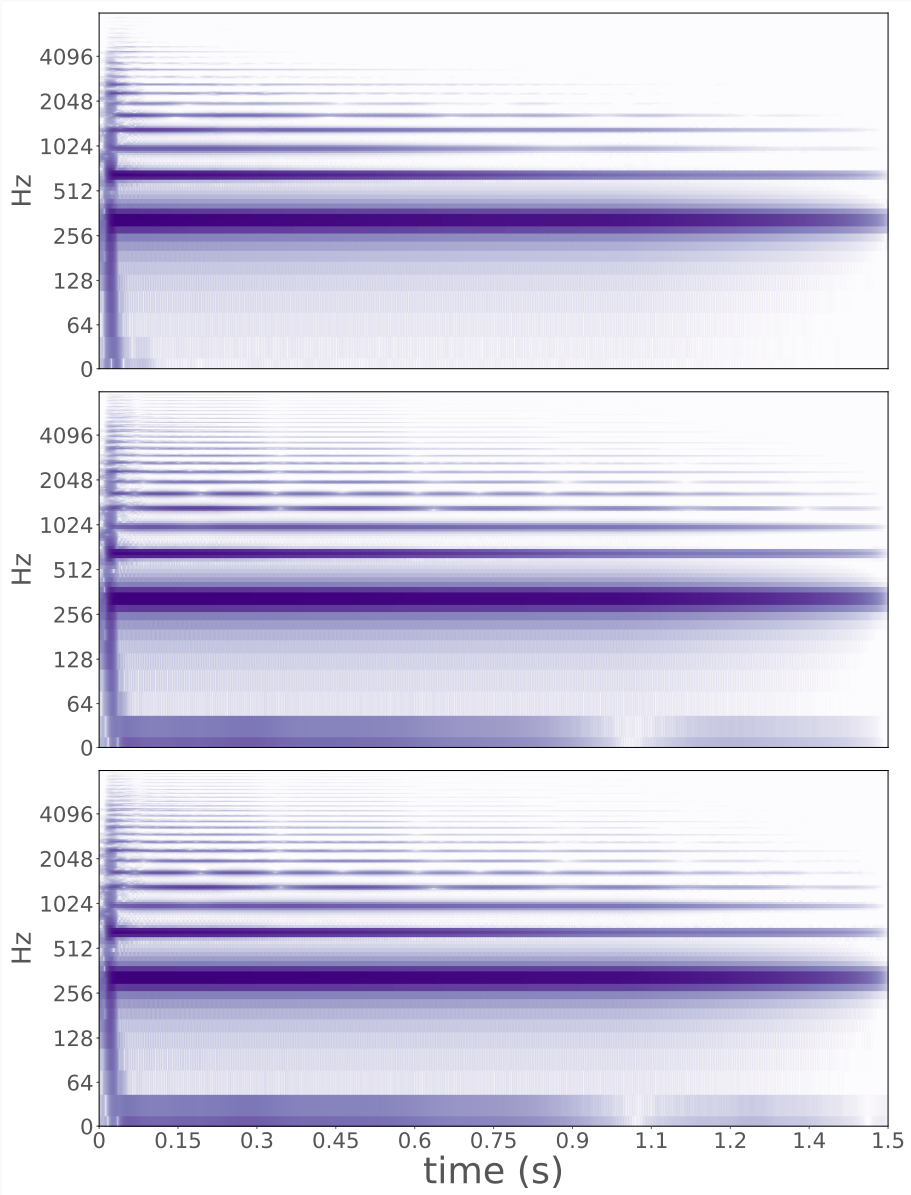
Results - overdrive - electric guitar





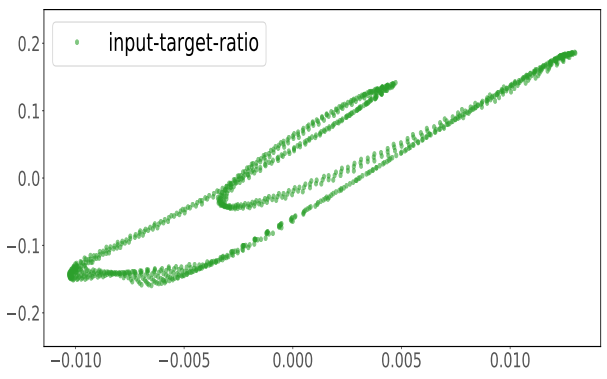
Results - overdrive - electric guitar

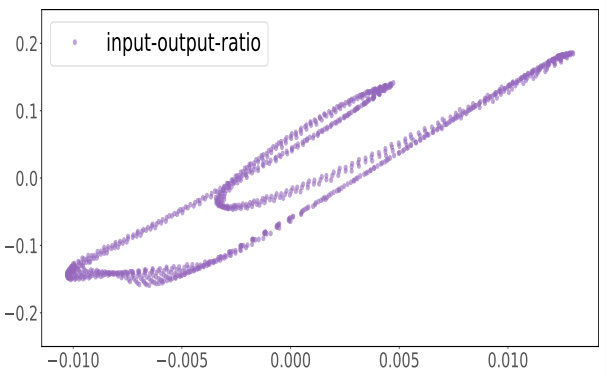
- input
- target
- output



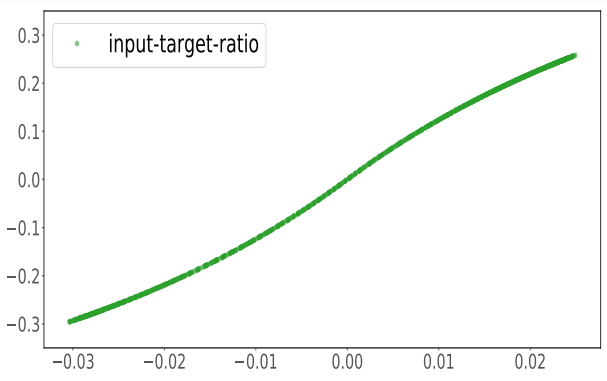
Results - overdrive - waveshaping curves

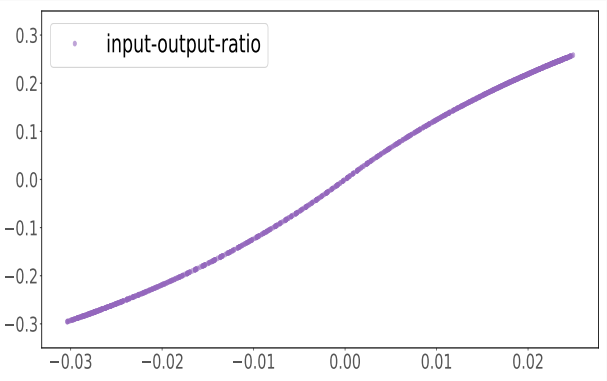
- 1st setting



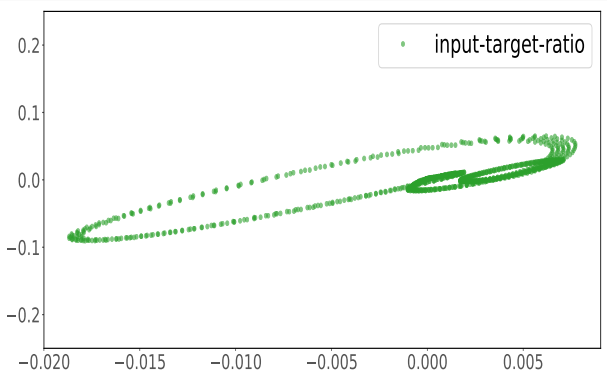


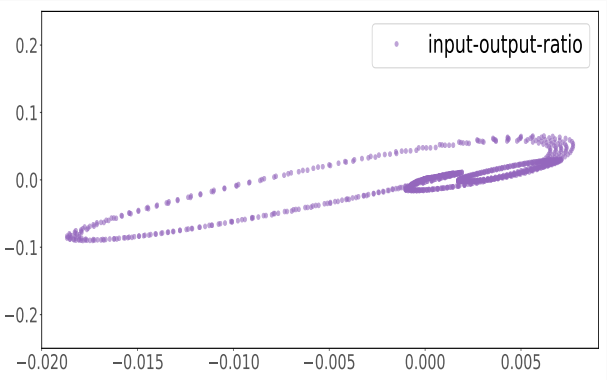
- 2nd setting



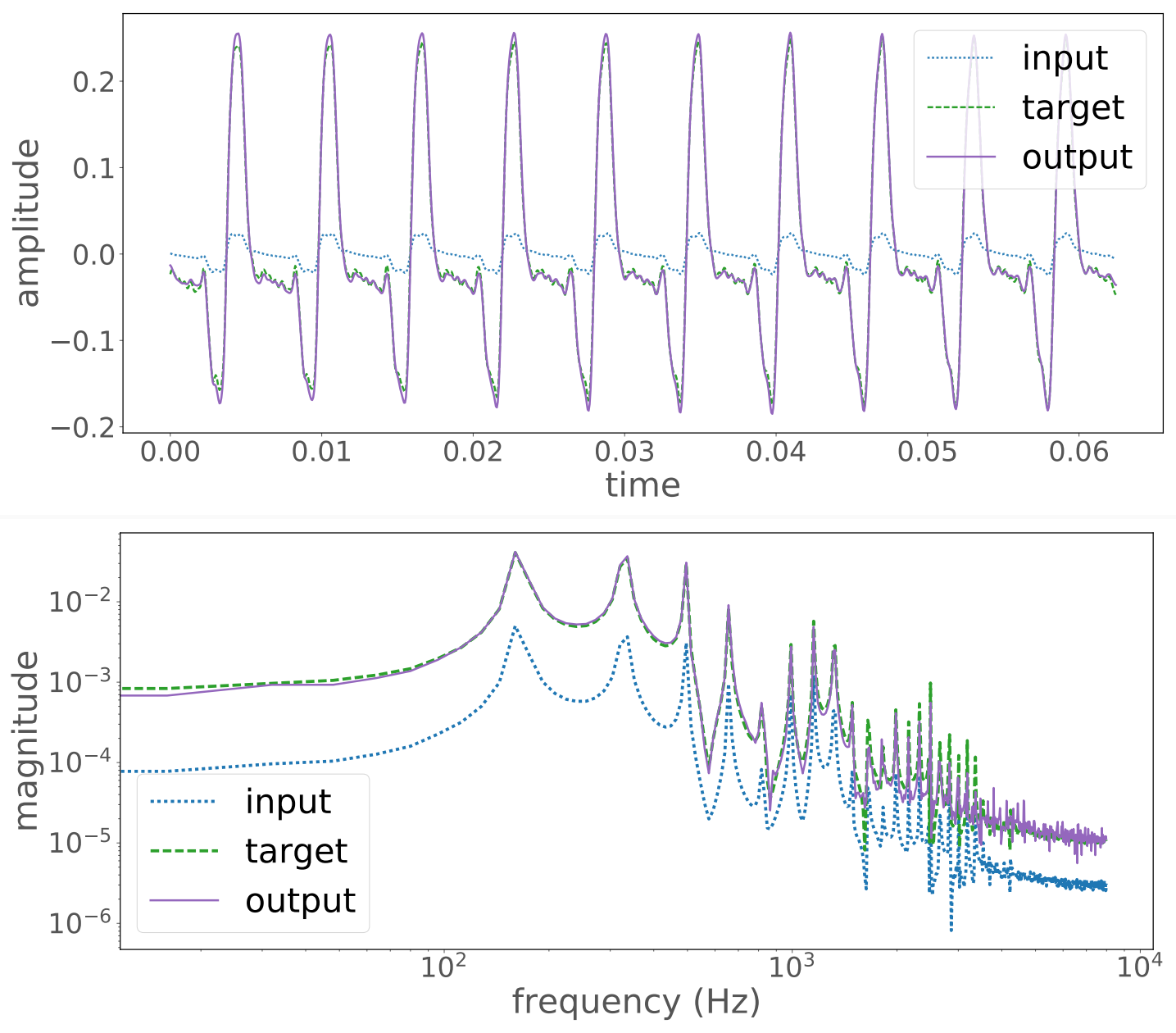


- 3rd setting



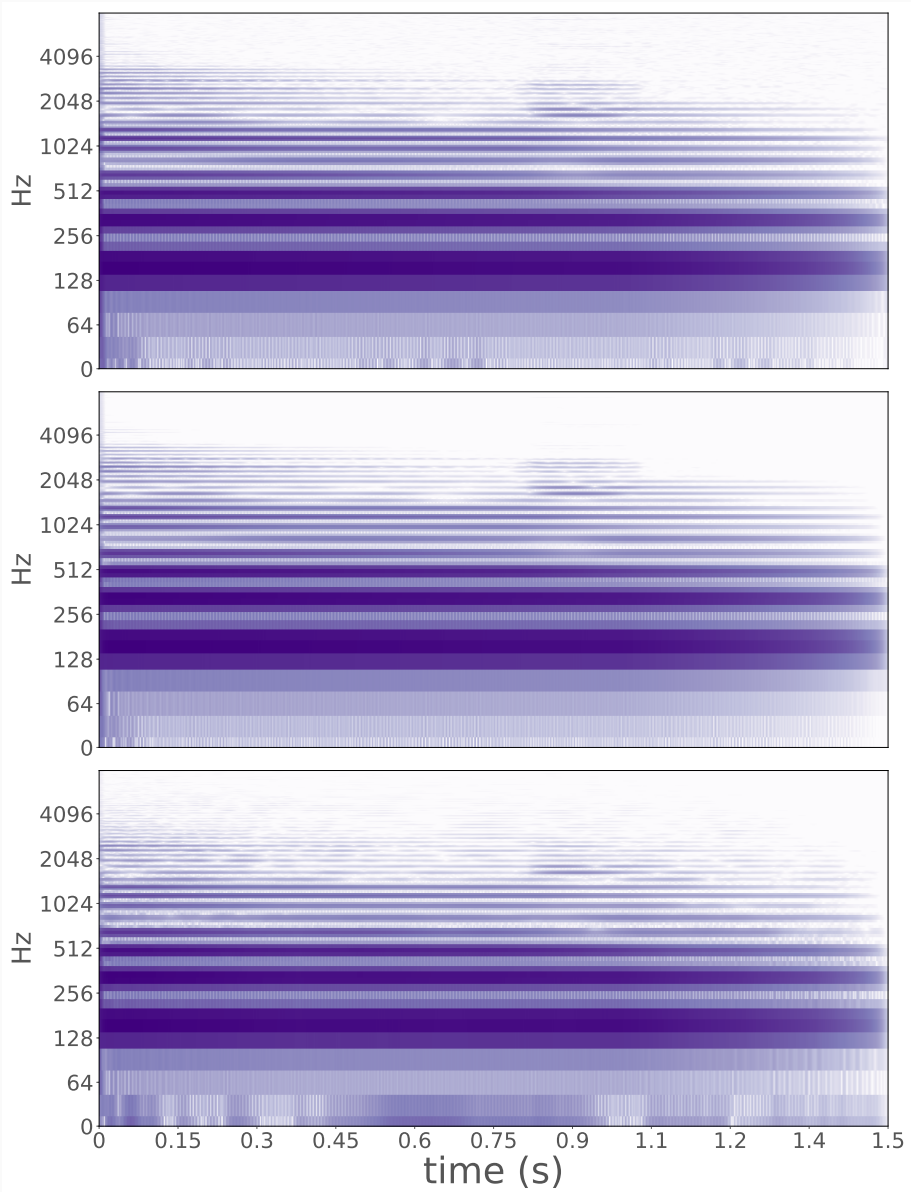


Results - amplifier emulation (EQ) - electric guitar

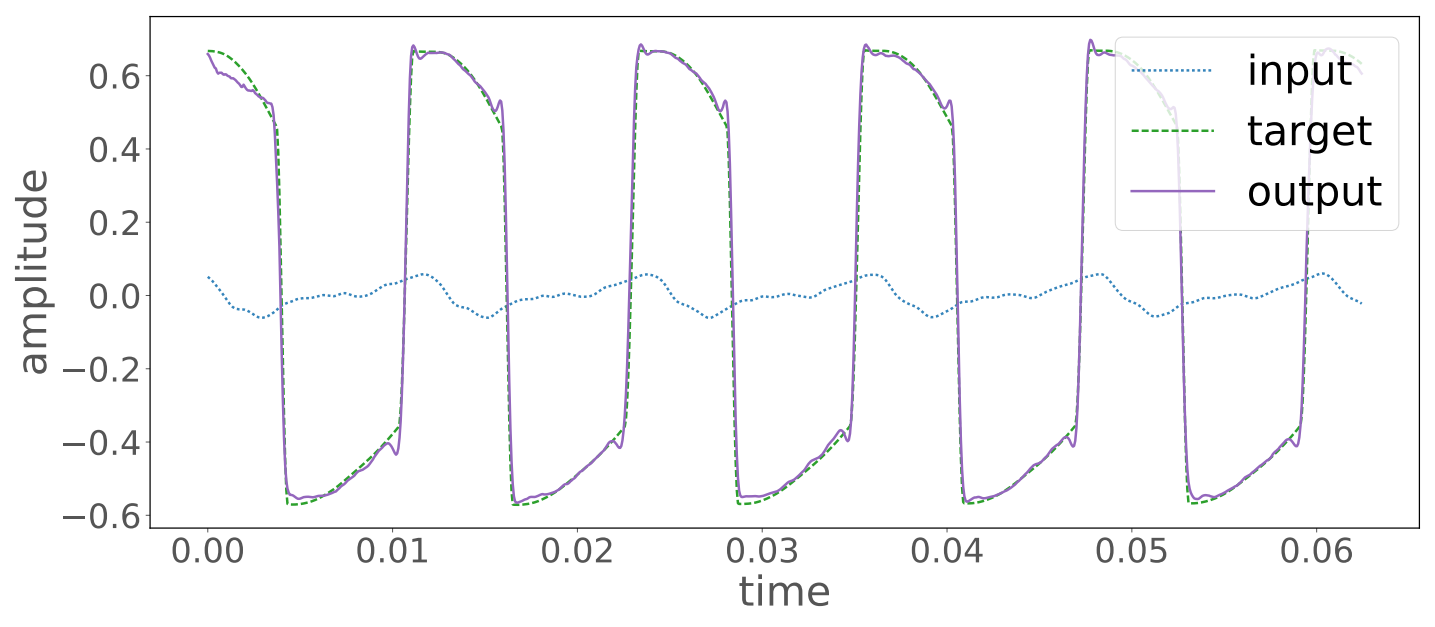


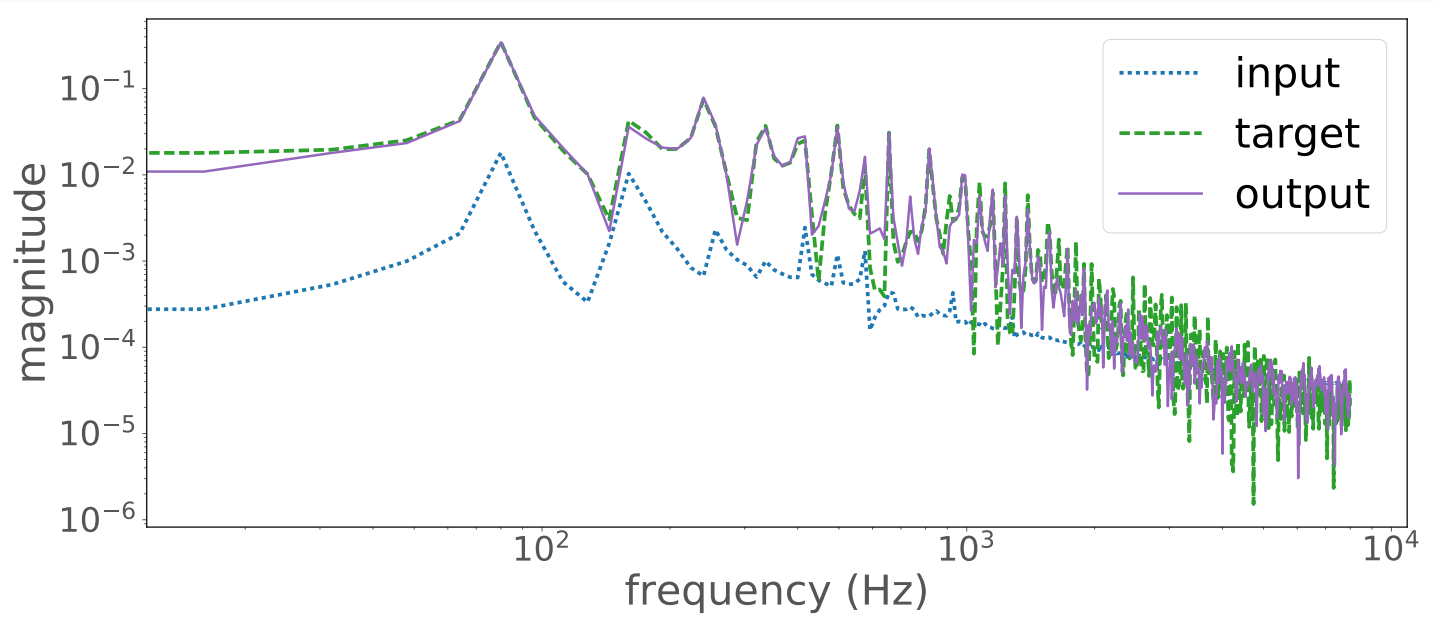
Results - amplifier emulation (EQ) - electric guitar

- input
- target
- output



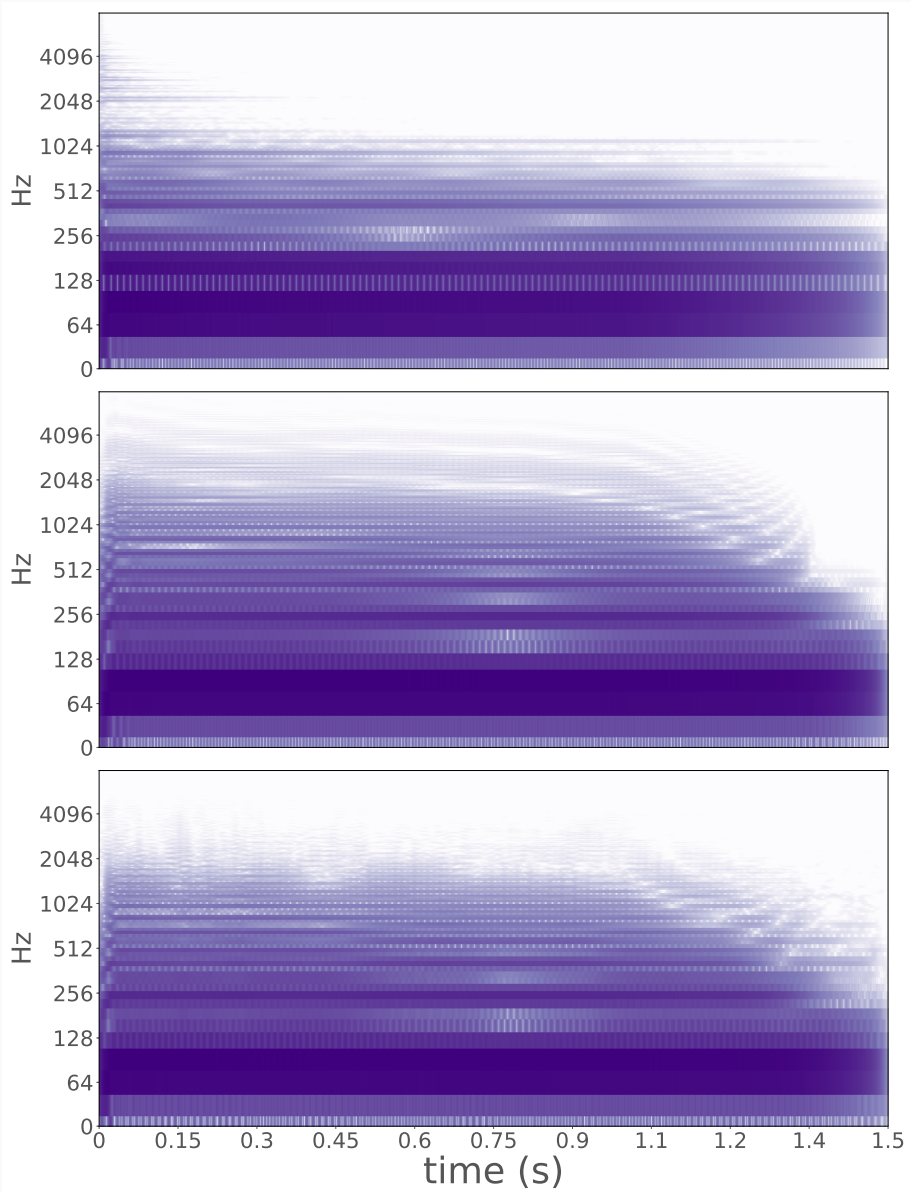
Results - FxChain - bass guitar





Results - FxChain - bass guitar

- input
- target
- output



Conclusion

Complex nonlinearities with *attack*, *release* and *filtering* settings were correctly modeled by the network.

Conclusion

Complex nonlinearities with *attack*, *release* and *filtering* settings were correctly modeled by the network.

Generalization capabilities among instruments and **optimization** towards an specific instrument were found among the trained models.

Conclusion

Complex nonlinearities with *attack*, *release* and *filtering* settings were correctly modeled by the network.

Generalization capabilities among instruments and **optimization** towards an specific instrument were found among the trained models.

We introduced a general-purpose deep learning architecture for generic black-box modeling of nonlinear and linear audio effects.

Future work

Further generalization could be explored.

- Regularizers
- Training data with a wider range of instruments.

Future work

Further generalization could be explored.

- Regularizers
- Training data with a wider range of instruments.

Exploration of RNNs to model effects with long-term memory;

Future work

Further generalization could be explored.

- Regularizers
- Training data with a wider range of instruments.

Exploration of RNNs to model effects with long-term memory;

- Martínez Ramírez M. A., Benetos E. and Reiss J. D., *"A general-purpose deep learning approach to model time-varying audio effects"*, arXiv preprint

Future work

Further generalization could be explored.

- Regularizers
- Training data with a wider range of instruments.

Exploration of RNNs to model effects with long-term memory;

- Martínez Ramírez M. A., Benetos E. and Reiss J. D., *"A general-purpose deep learning approach to model time-varying audio effects"*, arXiv preprint

Real-time implementations.

Thank you.

https://mchijmma.github.io/modeling-nonlinear/

www.m-marco.com