INTERNATIONAL AUDIO LABORATORIES ERLANGEN
A joint institution of Fraunhofer IIS and Universität Erlangen-Nürnberg

Winter Term 2026/27

Selected Topics in Deep Learning for Audio, Speech, and Music Processing

Lecture

Differentiable Digital Signal Processing (DDSP)

Stefan Balke, Andreas Brendel, Ben Maman, Meinard Müller

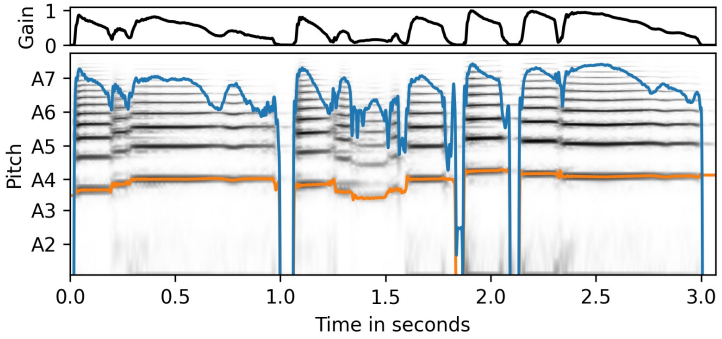
Friedrich-Alexander-Universität Erlangen-Nürnberg

Fraunhofer IIS

Differentiable Digital Signal Processing (DDSP)

Synthesizer Example

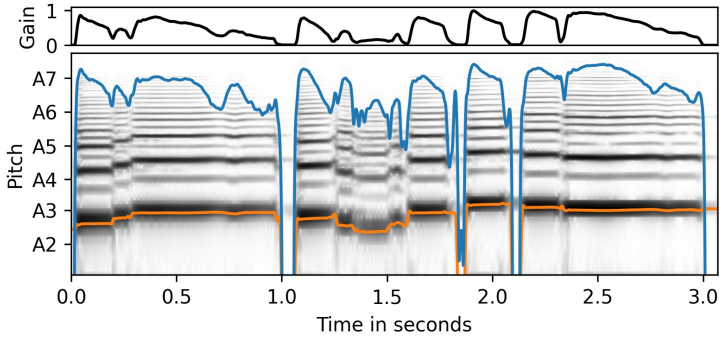






Loudness

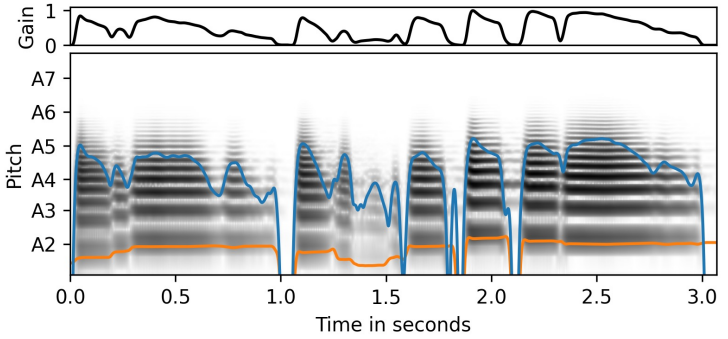






Timbre







Pitch (F0)

Differentiable Digital Signal Processing (DDSP)

Learning Objectives

1. Explain how DDSP builds on established DSP knowledge.
2. Describe how deep neural networks predict parameters for DSP components.
3. Explain how this combination enables controllable audio synthesis.
4. Learn about its limitations.

Thanks to my colleagues:



Meinard Müller



Simon Schwär



Christian Dittmar



Manuel Peters

And all the great contributions from the ISMIR and signal processing community.

© AudioLabs, 2026

Differentiable Digital Signal Processing (DDSP)

DDSP is an umbrella term that describes the combination of DSP building blocks with machine learning approaches (e.g., DNNs).

Differentiable Digital Signal Processing (DDSP)

DDSP is an umbrella term that describes the combination of DSP building blocks with machine learning approaches (e.g., DNNs).

- **Controllability:** Control signals can have a semantic meaning
- **Interpretability:** Used components are designed for a clear task (e.g., filtering)
- **Transfer:** We can link DSP knowledge with recent technology (ML)
- **Efficiency:** Less training data, smaller models, less compute

Differentiable Digital Signal Processing (DDSP)

DDSP is an umbrella term that describes the combination of DSP building blocks with machine learning approaches (e.g., DNNs).

When you think of DSP, which fundamental building blocks come to your mind?



Differentiable Digital Signal Processing (DDSP)

Fundamental Building Blocks

Building block	Typical parameters	Example use (audio)
Gain and mixing	Gains, mixing weights	Volume, crossfades, combining sources
Filters		
Oscillators		
Noise generators and shaping		
Delays and delay lines		
Time–frequency transforms		

© AudioLabs, 2026

Differentiable Digital Signal Processing (DDSP)

Fundamental Building Blocks

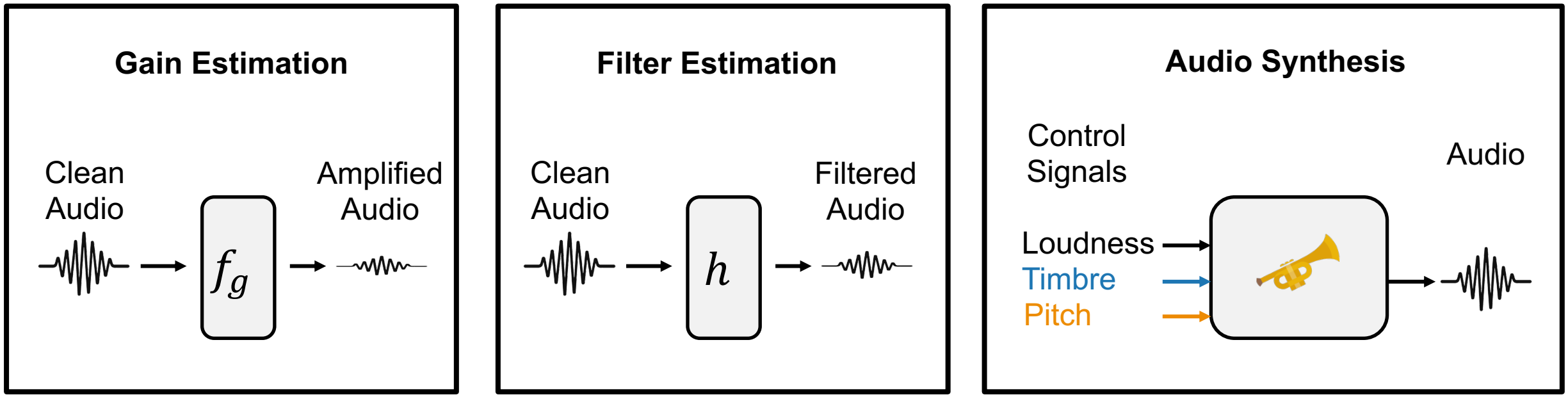
Building block	Typical parameters	Example use (audio)
Gain and mixing	Gains, mixing weights	Volume, crossfades, combining sources
Filters	FIR coefficients, cutoff, resonance	Equalization
Oscillators	Frequency, amplitude, phase	Sinusoidal and harmonic synthesis
Noise generators and shaping	Noise level, spectral envelope	Sound creation
Delays and delay lines	Delay time, feedback	Echo, Reverb
Time–frequency transforms	Window, hop size, transform size	Spectral analysis and processing

© AudioLabs, 2026

Differentiable Digital Signal Processing (DDSP)

DDSP is an umbrella term that describes the combination of DSP building blocks with machine learning approaches (e.g., DNNs).

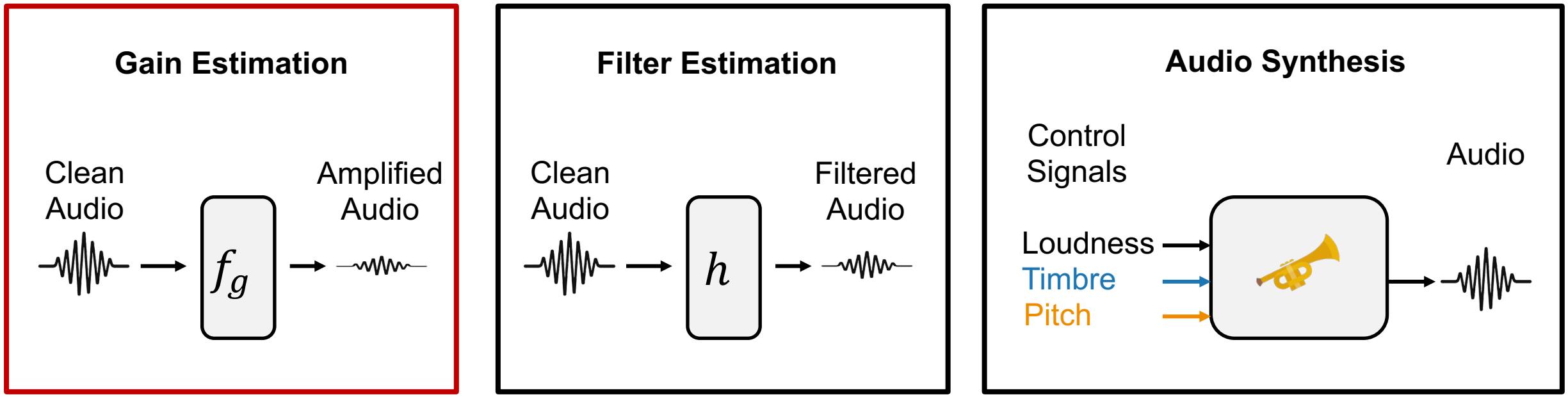
Three example scenarios:



Differentiable Digital Signal Processing (DDSP)

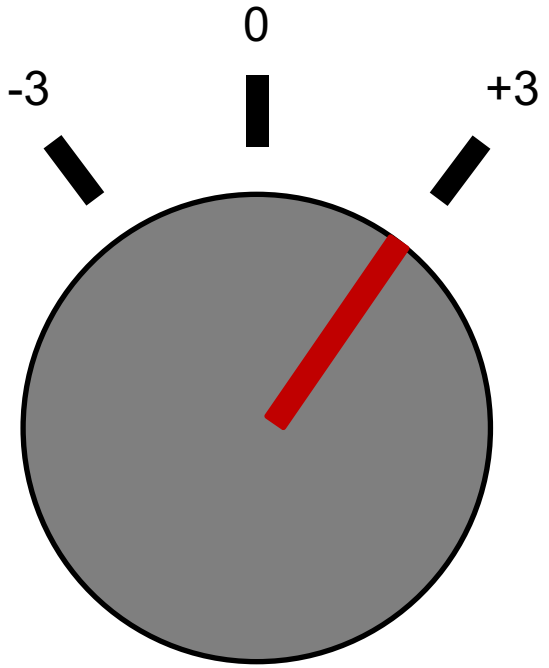
DDSP is an umbrella term that describes the combination of DSP building blocks with machine learning approaches (e.g., DNNs).

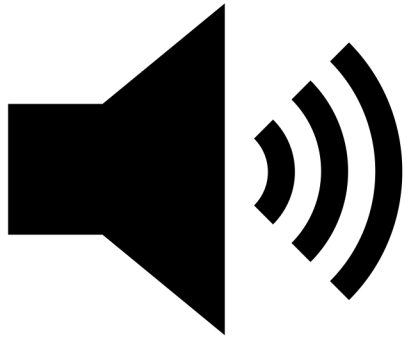
Three example scenarios:



DDSP: Gain Estimation







x

$x_1 = 2$

$f_\theta : \mathbb{R} \mapsto \mathbb{R}$

y

$y_1 = 6$

DDSP: Gain Estimation

- **Linear Model Assumption:**

$$\hat{y} = f_\theta(x) = \theta x$$

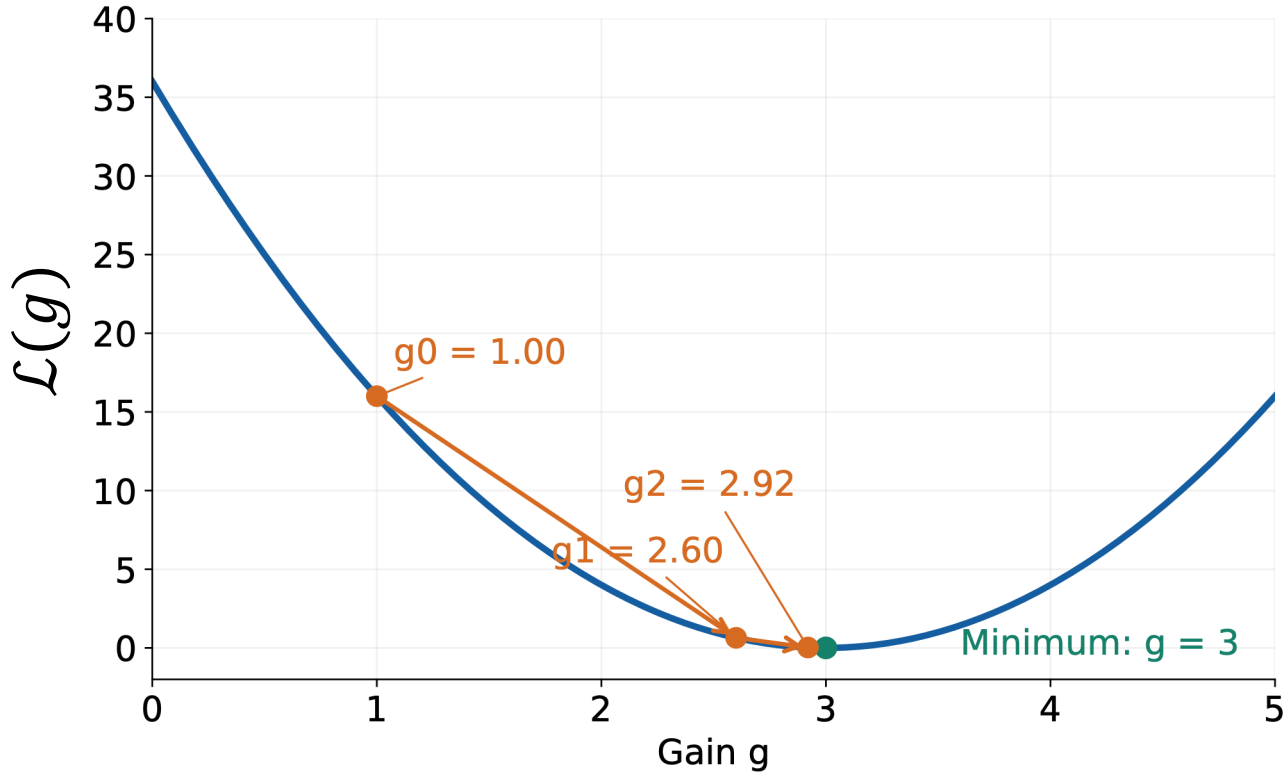
- **Loss Term (Squared Error):**

$$\mathcal{L}(\hat{y}, y) = (\hat{y} - y)^2 = (\theta x - y)^2$$

- **Gradient of the Loss w.r.t. θ:**

$$\frac{\partial \mathcal{L}}{\partial \theta} = \frac{\partial \mathcal{L}}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial \theta} = 2(\theta x - y) \cdot x$$

3. **Minimize loss with Gradient Descent.**



Analytical Solution:

- $x = 2,\ y = 6 \rightarrow \mathcal{L}(\theta) = (2\theta - 6)^2$
- $\frac{\partial \mathcal{L}}{\partial \theta} = 2(\theta x - y) \cdot x = 4(2\theta - 6)$
- $4(2\theta - 6) = 0 \rightarrow \theta = 3.$

© AudioLabs, 2026

DDSP: Gain Estimation

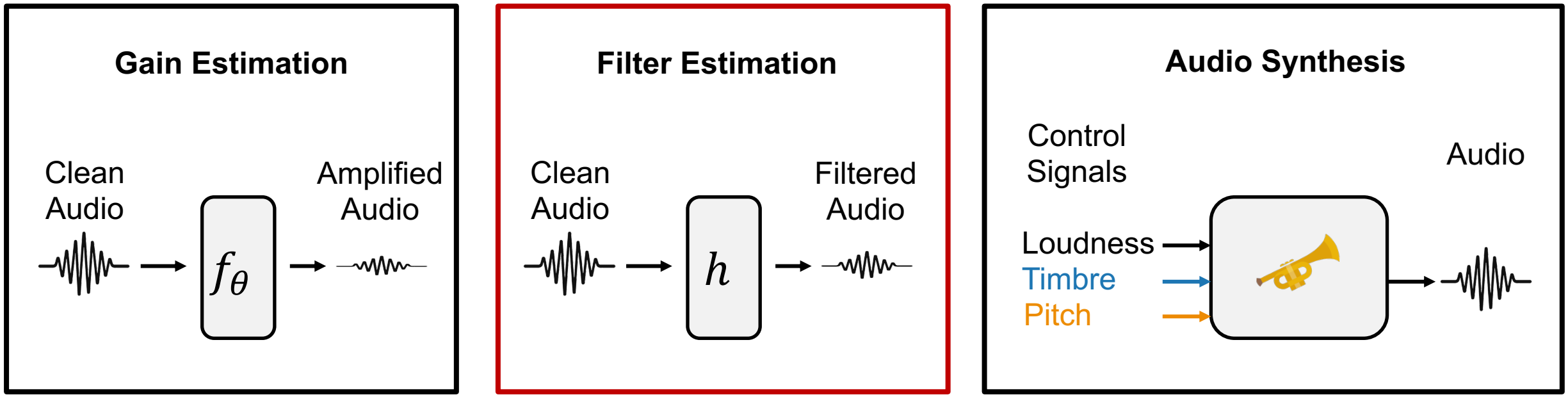
Recap of what we just did:

- Estimated the parameter of a linear model
- In this toy example, analytical solution was possible
  → **Result:** All inputs will be multiplied by 3
- Let us extend the model to more parameters!

Differentiable Digital Signal Processing (DDSP)

DDSP is an umbrella term that describes the combination of DSP building blocks with machine learning approaches (e.g., DNNs).

Three example scenarios:

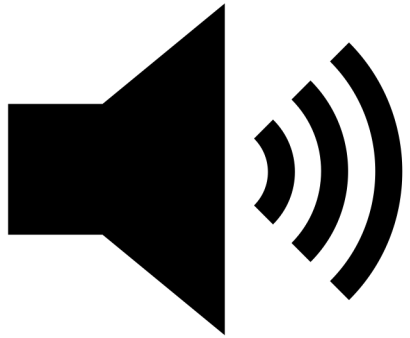


DDSP: Filter Estimation





FIR Filter with
K Coefficients



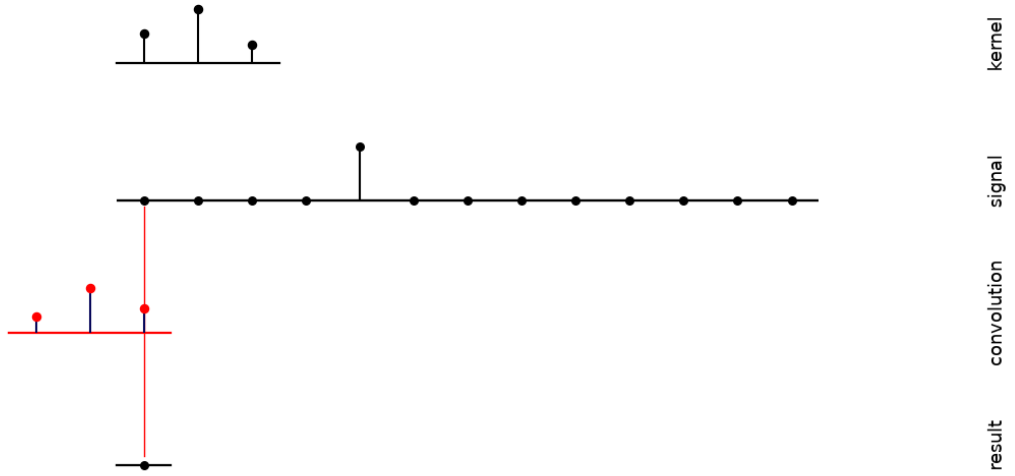
x h y

DDSP: Filter Estimation

- **Linear Model Assumption:**

$$\hat{y}(\mathrm{t}) = (h * x)(t) = \sum_{k=0}^{K-1} h(k)\, x(t-k)$$

Convolution



DDSP: Filter Estimation

Input: $x(0) = 1, x(1) = 2, x(2) = 3$

Filter: $h(0) = 2, h(1) = 1$

Task: Calculate the outputs for (a) $y(1)$ and (b) $y(2)$.

(a)

$$y(1) = \sum_{k=0}^{K-1} h(k)\, x(1-k) = h(0) \cdot x(1) + h(1) \cdot x(0) = 2 \cdot 2 + 1 \cdot 1 = 5$$

(b)

$$y(2) = \sum_{k=0}^{K-1} h(k)\, x(2-k) = h(0) \cdot x(2) + h(1) \cdot x(1) = 2 \cdot 3 + 1 \cdot 2 = 8$$

DDSP: Filter Estimation

- **Linear Model Assumption:**

$$\hat{y}(\mathrm{t}) = h(t) * x(t) = \sum_{k=0}^{K-1} h(k)\, x(t-k)$$

- **Gradient:**

$$\frac{\partial \mathcal{L}}{\partial h(\mathrm{k})} = \frac{2}{T} \sum_{\mathrm{t}=0}^{\mathrm{T}-1} \big(\hat{y}(t) - y(t)\big)\, x(\mathrm{t}-\mathrm{k})$$

- **Loss Term (Mean Squared Error):**

$$\mathcal{L}(\hat{y}, y) = \frac{1}{T} \sum_{\mathrm{t}=0}^{\mathrm{T}-1} (\hat{y}(t) - y(t))^2$$

→ Use Gradient descent to minimize!

DDSP: Filter Estimation

Example with $\mathbf{K = 3}$:

$$\hat{y}(\mathrm{t}) = h(t) * x(t) = h(0)x(t) + h(1)x(t-1) + h(2)x(\mathrm{t}-2)$$

For $t = t_0$:

$$\hat{y}(t_0) = h(0)x(t_0) + h(1)x(t_0 - 1) + h(2)x(t_0 - 2)$$

Loss for $t = t_0$:

$$\mathcal{L}_{t_0} = (\hat{y}(t_0) - y(t_0))^2 = \big(h(0)x(t_0) + h(1)x(t_0 - 1) + h(2)x(t_0 - 2) - y(t_0)\big)^2$$

DDSP: Filter Estimation

Loss for $t = t_0$:

$$\mathcal{L}_{t_0} = (\hat{y}(t_0) - y(t_0))^2 = \big(h(0)x(t_0) + h(1)x(t_0 - 1) + h(2)x(t_0 - 2) - y(t_0)\big)^2$$

Partial Derivatives:

- $\frac{\partial \mathcal{L}_{t_0}}{\partial h(0)} = \frac{\partial \mathcal{L}}{\partial \hat{y}(t_0)} \frac{\partial \hat{y}(t_0)}{\partial h(0)} = 2\left(\hat{y}(t_0) - y(t_0)\right) x(t_0)$
- $\frac{\partial \mathcal{L}_{t_0}}{\partial h(1)} = \frac{\partial \mathcal{L}}{\partial \hat{y}(t_0)} \frac{\partial \hat{y}(t_0)}{\partial h(1)} = 2\left(\hat{y}(t_0) - y(t_0)\right) x(t_0 - 1)$
- $\frac{\partial \mathcal{L}_{t_0}}{\partial h(2)} = \frac{\partial \mathcal{L}}{\partial \hat{y}(t_0)} \frac{\partial \hat{y}(t_0)}{\partial h(2)} = 2\left(\hat{y}(t_0) - y(t_0)\right) x(t_0 - 2)$

© AudioLabs, 2026

DDSP: Filter Estimation

Partial Derivatives for $k = 0, \dots, K-1$:

$$\frac{\partial \mathcal{L}_{t_0}}{\partial h(k)} = 2\big(\hat{y}(t_0) - y(t_0)\big)x(t_0 - k)$$

Apply to all t (take the average):

$$\frac{\partial \mathcal{L}}{\partial h(k)} = \frac{1}{T}\sum_{t=0}^{T-1}\frac{\partial \mathcal{L}_t}{\partial h(k)} = \frac{2}{T}\sum_{t=0}^{T-1}\big(\hat{y}(t) - y(t)\big)\,x(t-k)$$

for $k = 0, \dots, K-1$

Very similar to the LMS algorithm

Widrow and Hoff:, Adaptive switching circuits, 1960 IRE WESCON Convention Record, New York: IRE, pp. 96–104, 1960.

DDSP: Filter Estimation

Trumpet Example

$x(t)$: Clean trumpet signal

$\mathrm{y}(t)$: Trumpet recorded with a mute

$h(t)$: FIR filter approximation of the spectral difference





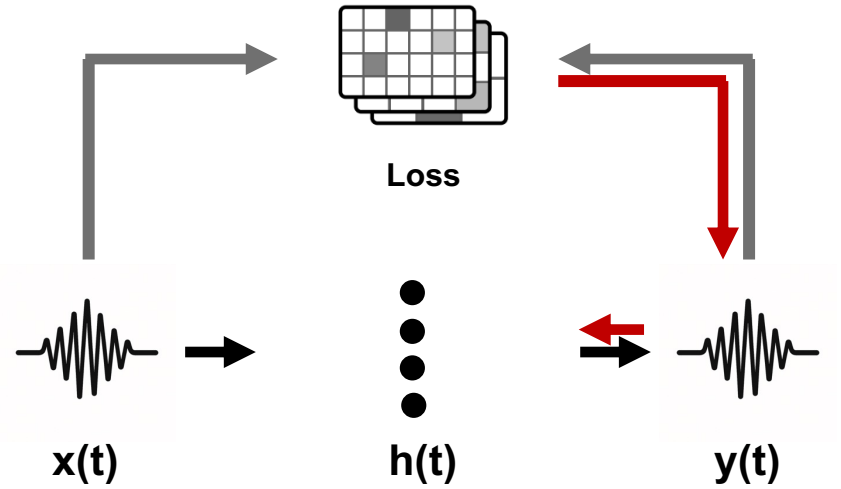
DDSP: Filter Estimation
FIR Filter Coefficients (Direct)

Inspired by: B. Hayes, J. Shier, C. Yu, D. Südholt, R. Diaz: Introduction to DDSP for Audio Synthesis: https://intro2ddsp.github.io/ ISMIR Tutorial, 2023, Milano, Italy.

- Use DSP for filter estimation

$$y(t) = h(t) * x(t)$$

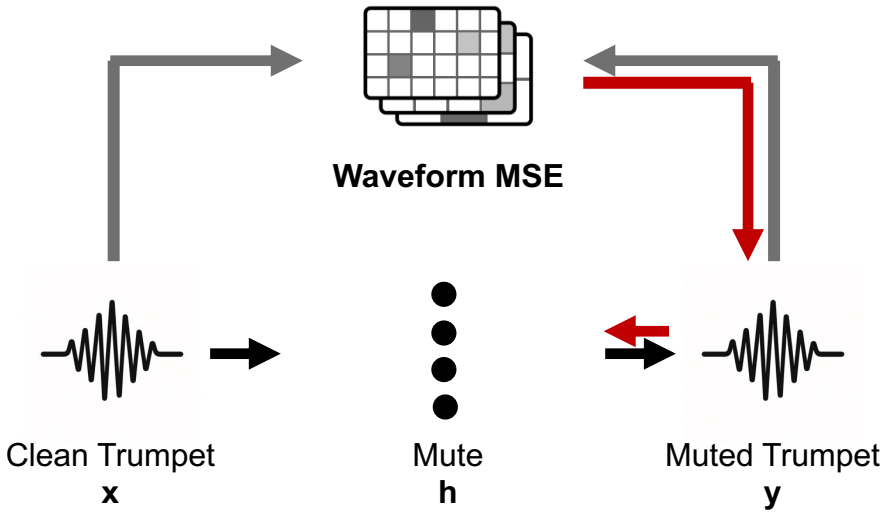
- **Task:** Estimate $h(t)$.



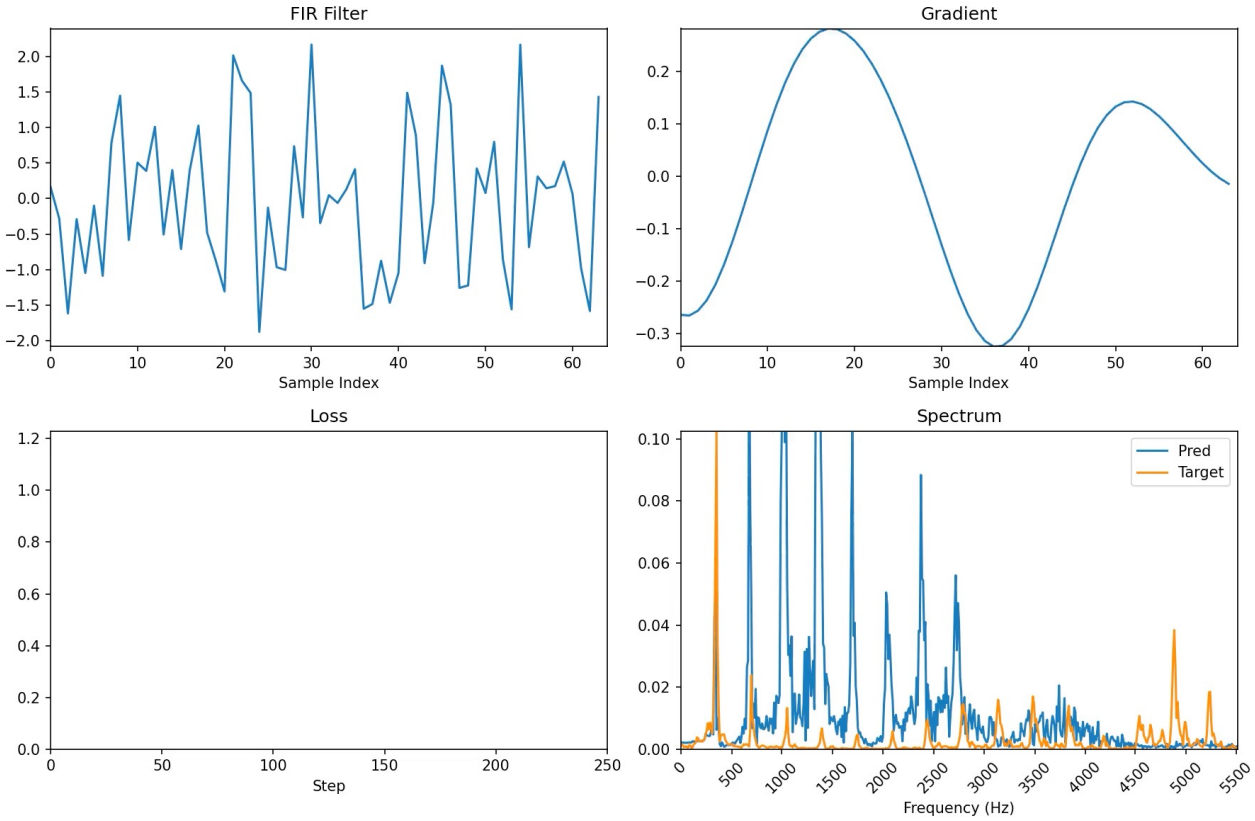
- **Approach:**
  - Model has K=64 trainable parameters
  - Followed by 1D-Convolution
  - Define Loss (e.g., MSE)
  - Gradient descent for loss minimization

DDSP: Filter Estimation

Trumpet Example

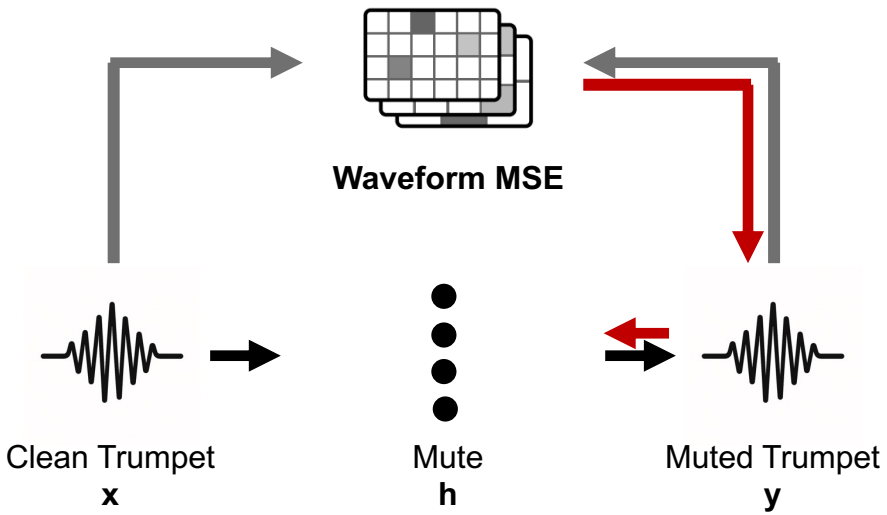


Clean trumpet filtered with learned h:

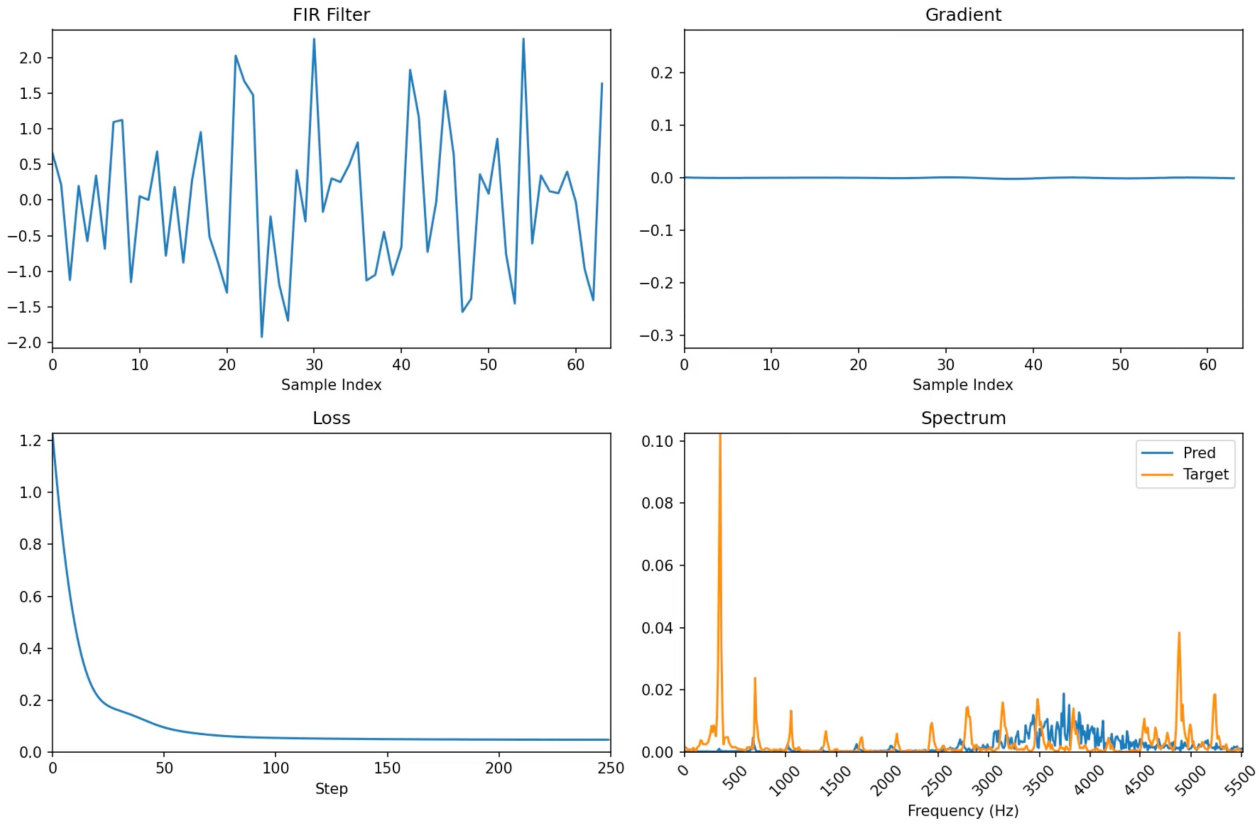


DDSP: Filter Estimation

Trumpet Example

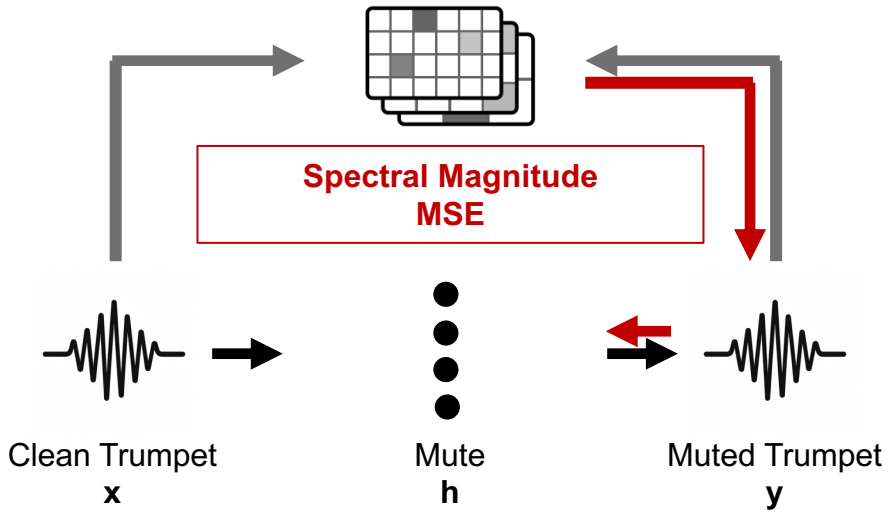


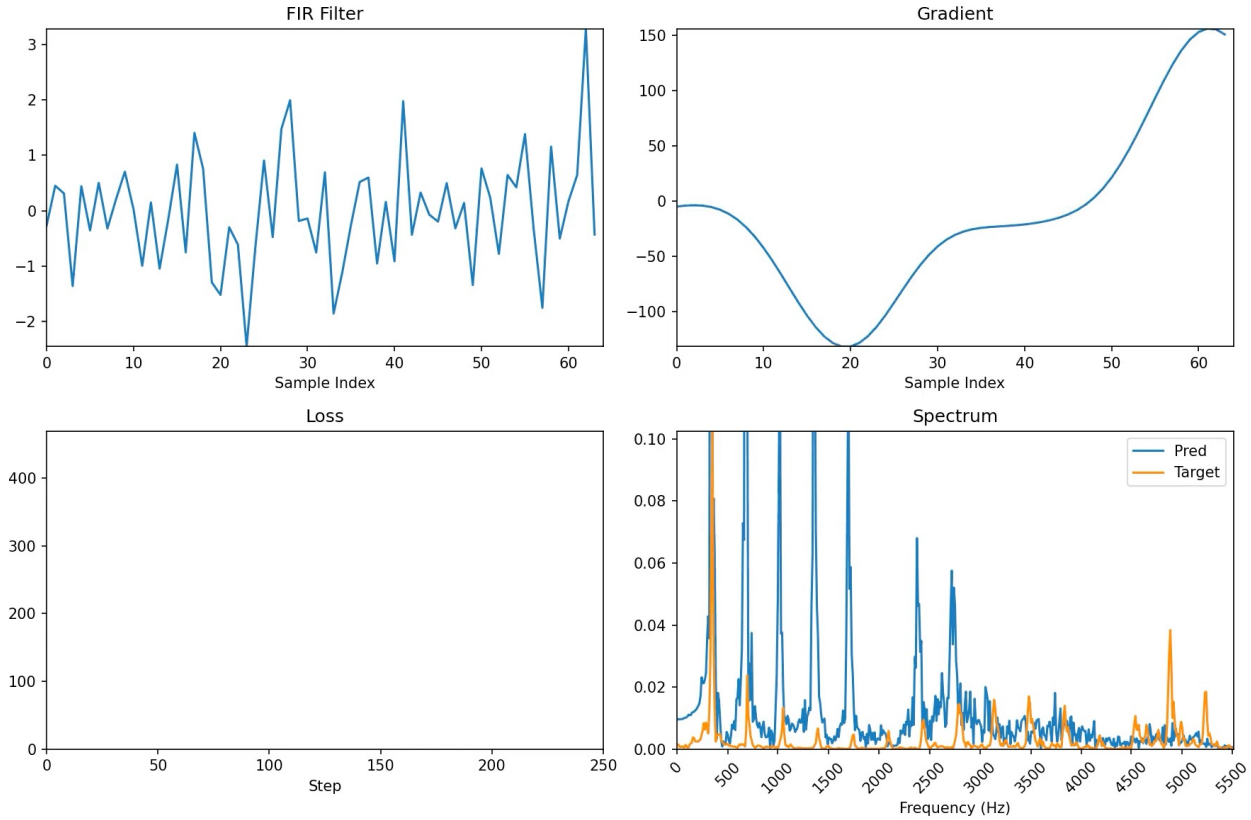
Clean trumpet filtered with learned h:



DDSP: Filter Estimation

Trumpet Example



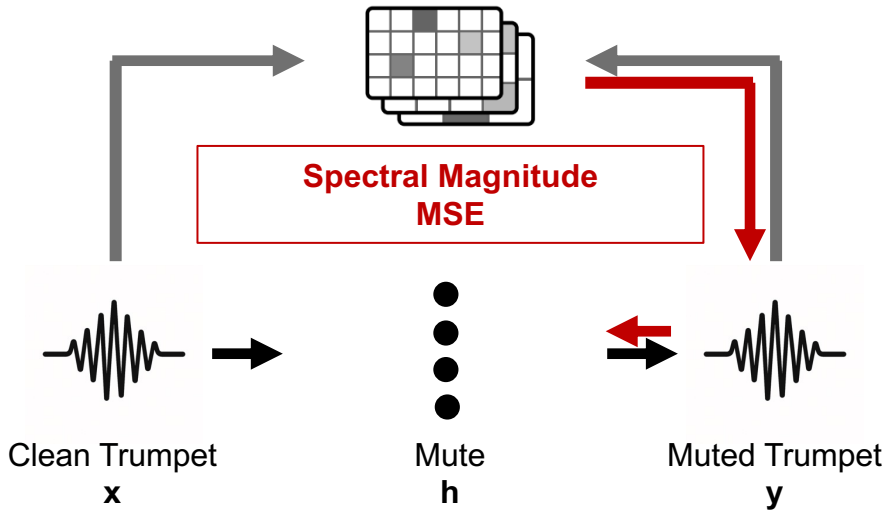


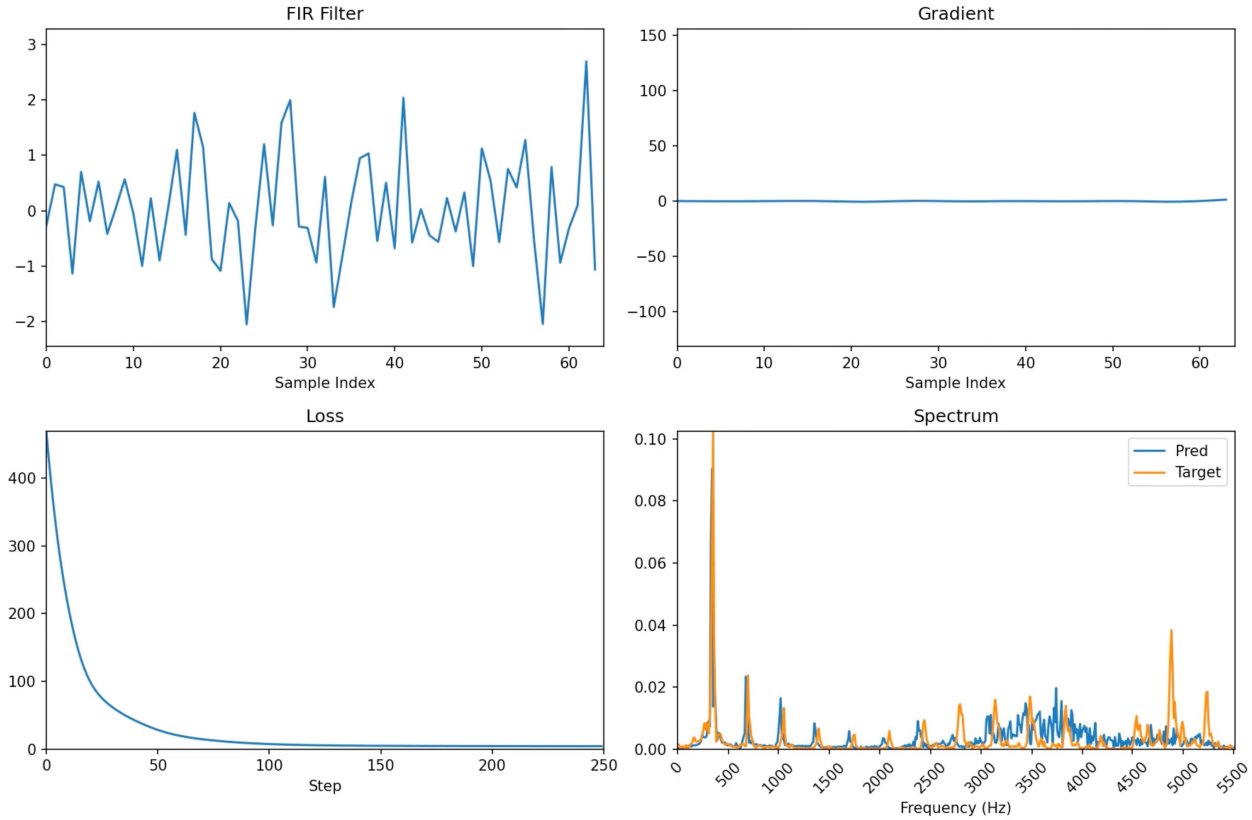
Clean trumpet filtered with learned h:

reference

DDSP: Filter Estimation

Trumpet Example



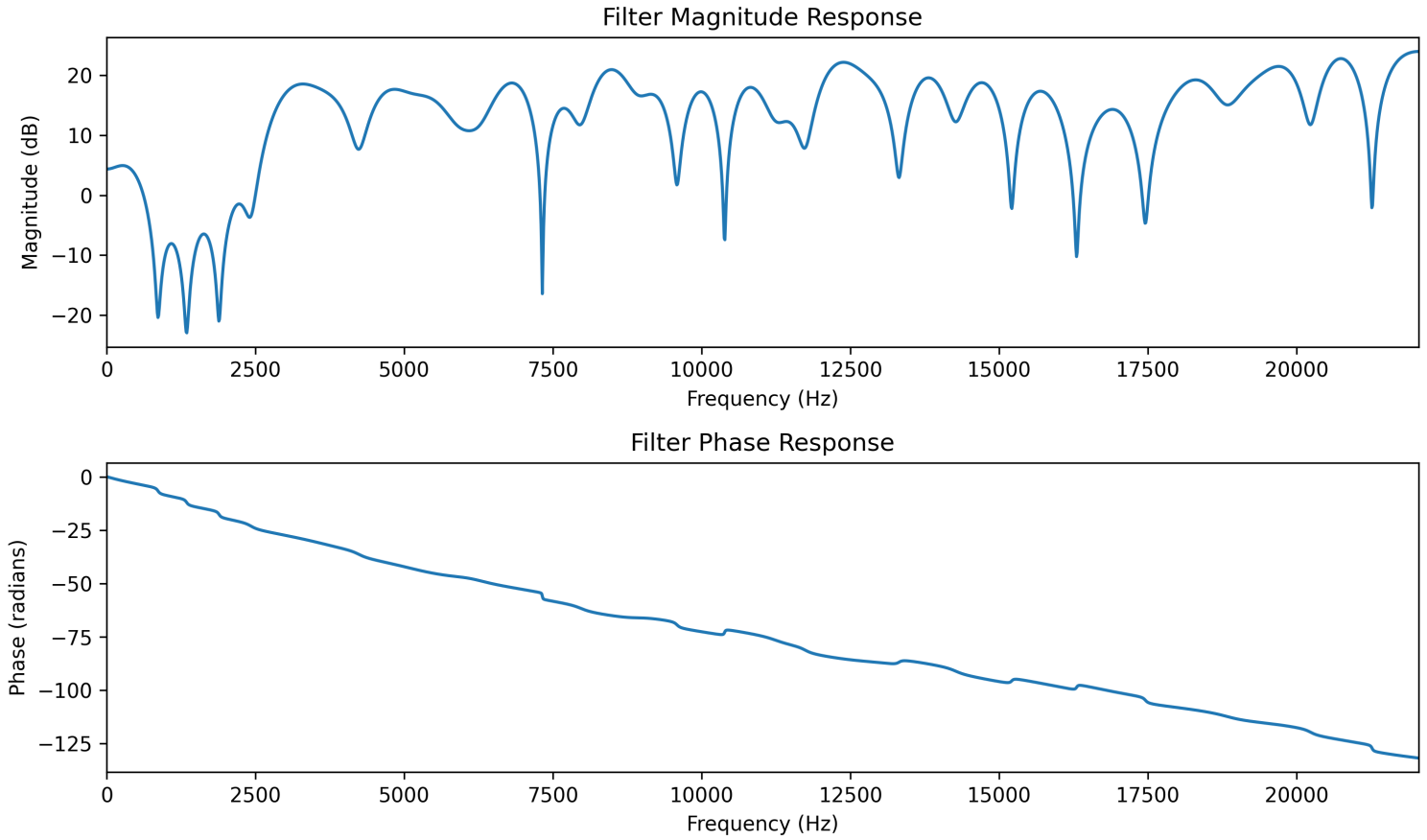


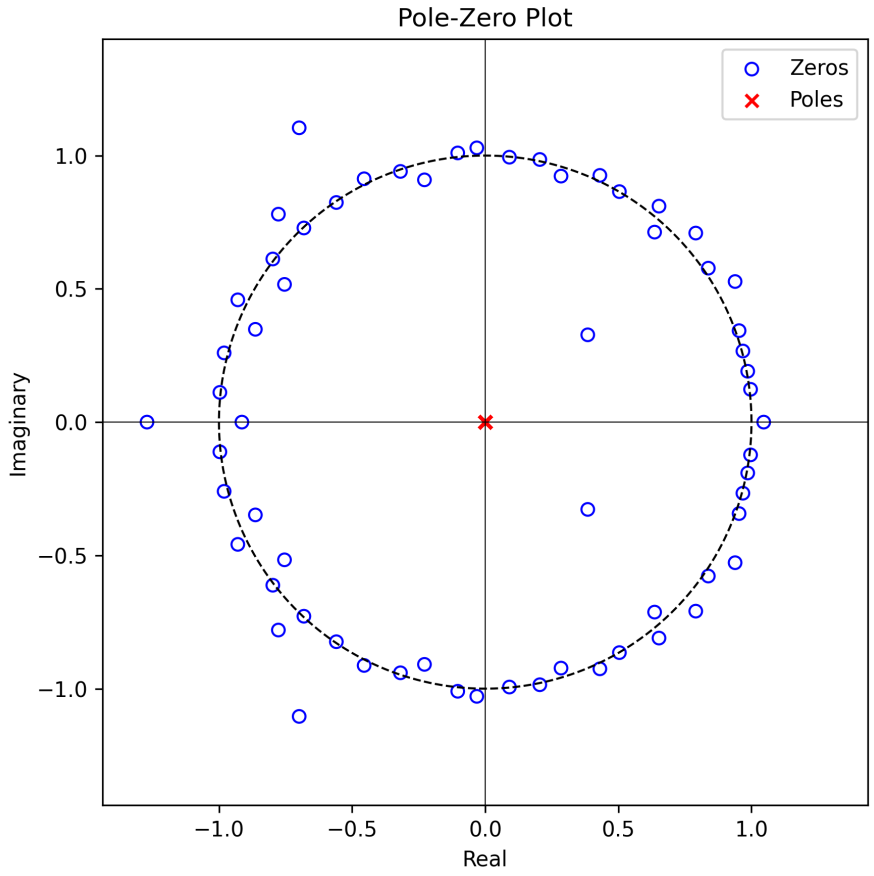
Clean trumpet filtered with learned h:

reference

DDSP: Filter Estimation

Trumpet Example





DDSP: Filter Estimation

Loss Selection



Think about the loss selection.
What impact does it have on the outcome?
Discuss with your partner, share in class.

Property	Waveform MSE	Spectral Magnitude MSE
Compares		
Matching objective		
Phase		
Time shifts		
Weighting		

DDSP: Filter Estimation

Loss Selection



Think about the loss selection.
What impact does it have on the outcome?
Discuss with your partner, share in class.

Property	**Waveform MSE**	**Spectral Magnitude MSE**
Compares	Individual audio samples	STFT magnitudes in each time-frequency bin
Matching objective		
Phase		
Time shifts		
Weighting		

DDSP: Filter Estimation

Think–Pair–Share



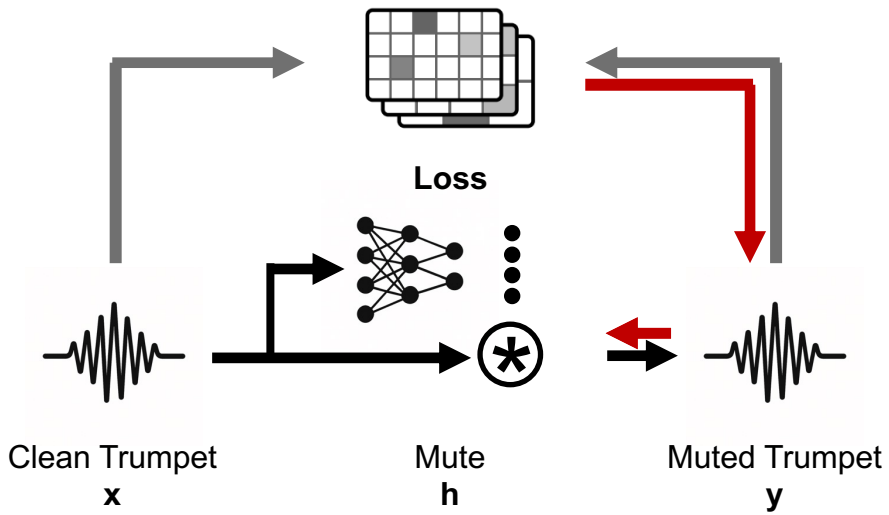
Think about the loss selection.
What impact does it have to the outcome?
Discuss with your partner, share in class.

Property	Waveform MSE	Spectral Magnitude MSE
Compares	Individual audio samples	STFT magnitudes in each time-frequency bin
Matching objective	The exact waveform	The local spectral distribution
Phase	Penalizes phase differences	Not explicitly
Time shifts	Very sensitive, even to small shifts	Often less sensitive to small shifts, but not shift-invariant
Weighting	Large sample errors dominate	Large absolute magnitude errors dominate

© AudioLabs, 2026

DDSP: Filter Estimation

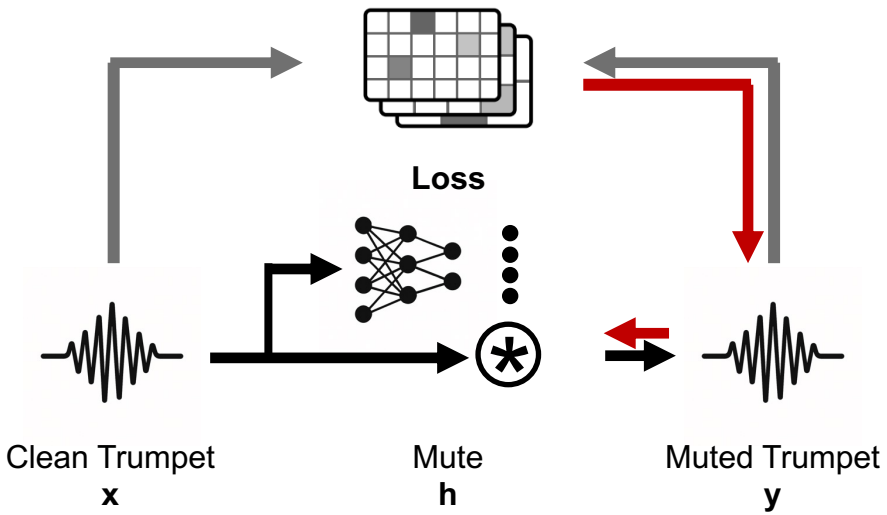
Neural Parameter Prediction

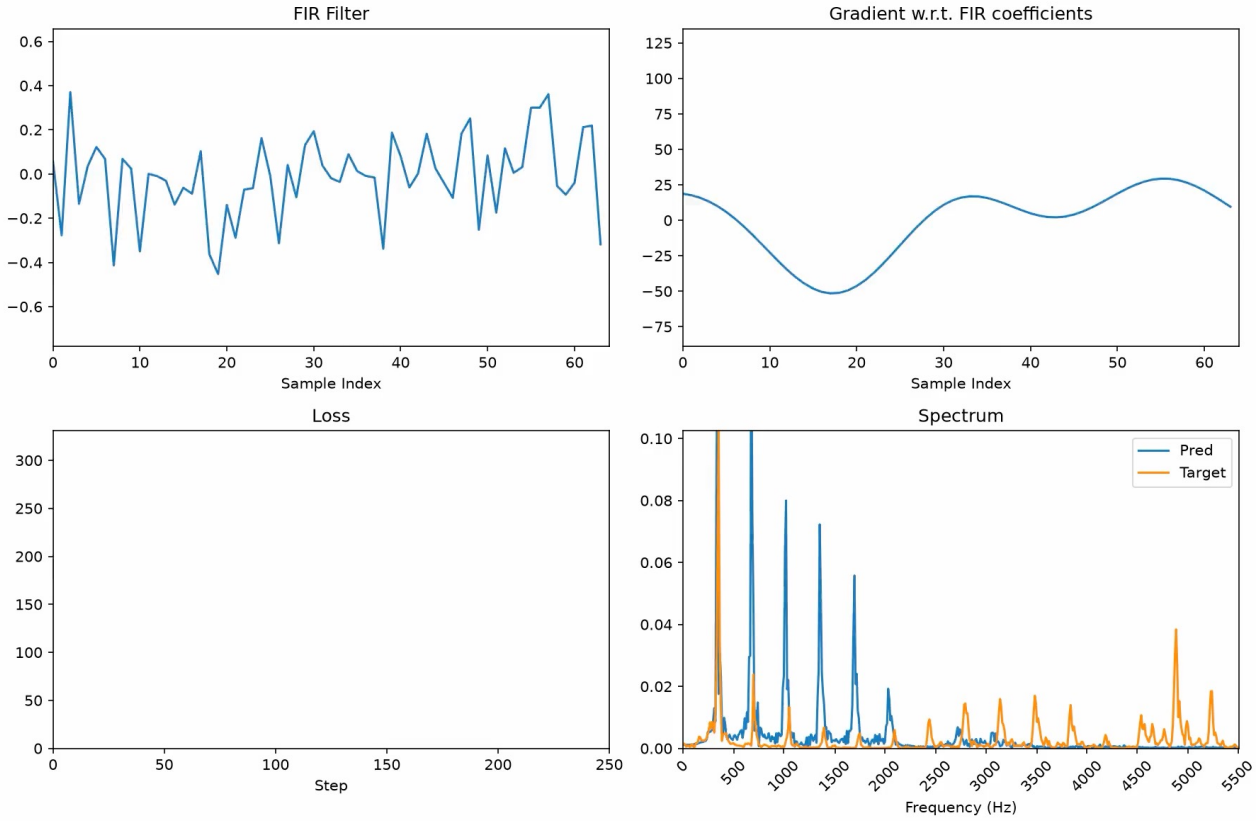


- No more direct optimization
- Neural network estimates the coefficients, i.e. there can be more hidden layers before the K-output parameters
- DNN output is followed by a 1D-Convolution

DDSP: FIR Filter Learning

Neural Parameter Prediction



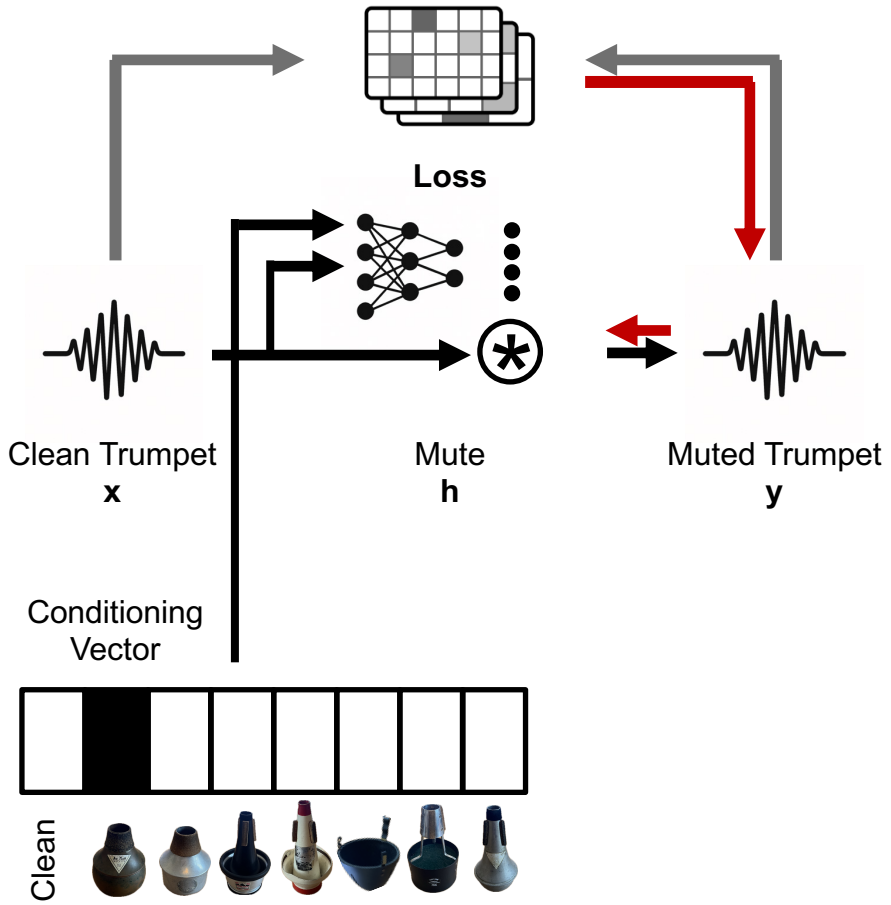


Clean trumpet filtered with learned h:

reference

DDSP: FIR Filter Learning

Neural Parameter Prediction





Ref.

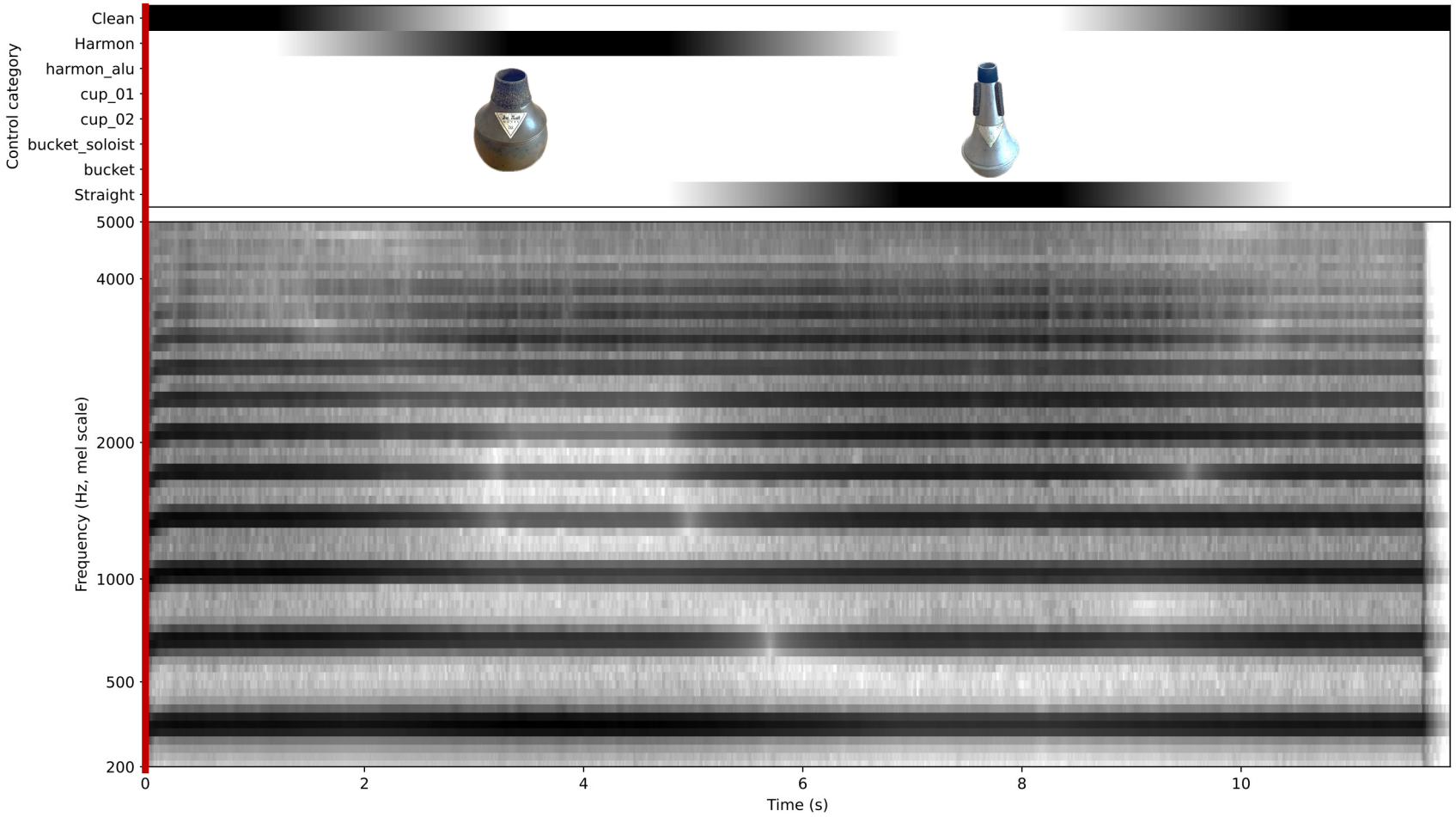
Est.

DDSP: Mute Morphing







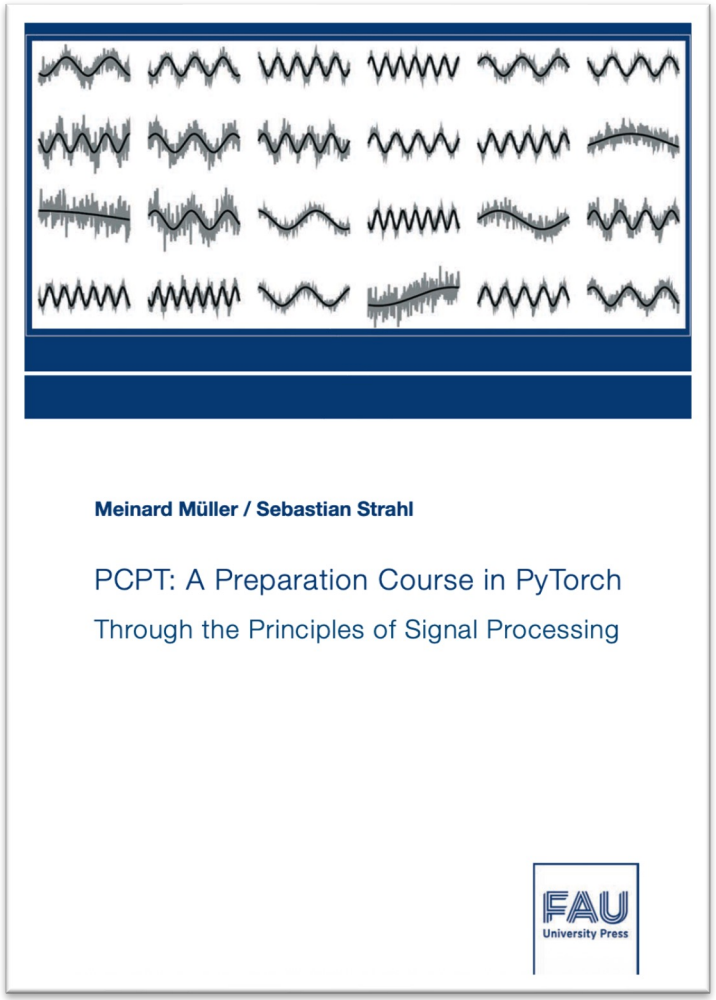


Differential Digital Signal Processing
Recursive (IIR) Filters

Further reading:

Unit 9 describes analogy IIR filter → RNN

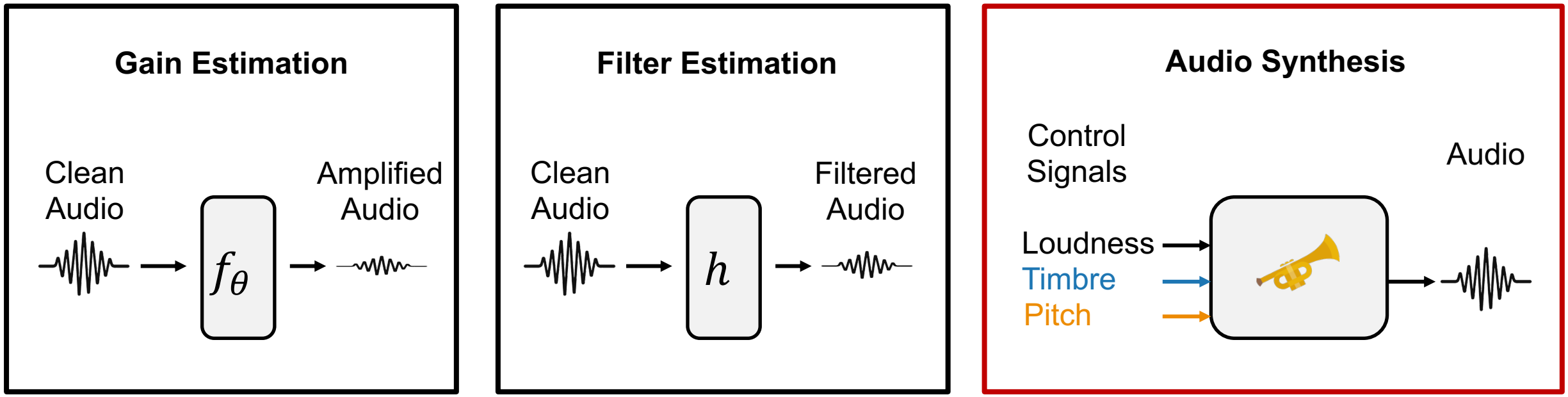
https://www.audiolabs-erlangen.de/resources/MIR/PCPT/PCPT.html



Differentiable Digital Signal Processing (DDSP)

DDSP is an umbrella term that describes the combination of DSP building blocks with machine learning approaches (e.g., DNNs).

Three example scenarios:



Differentiable Digital Signal Processing (DDSP)

Synthesis

Input



Output



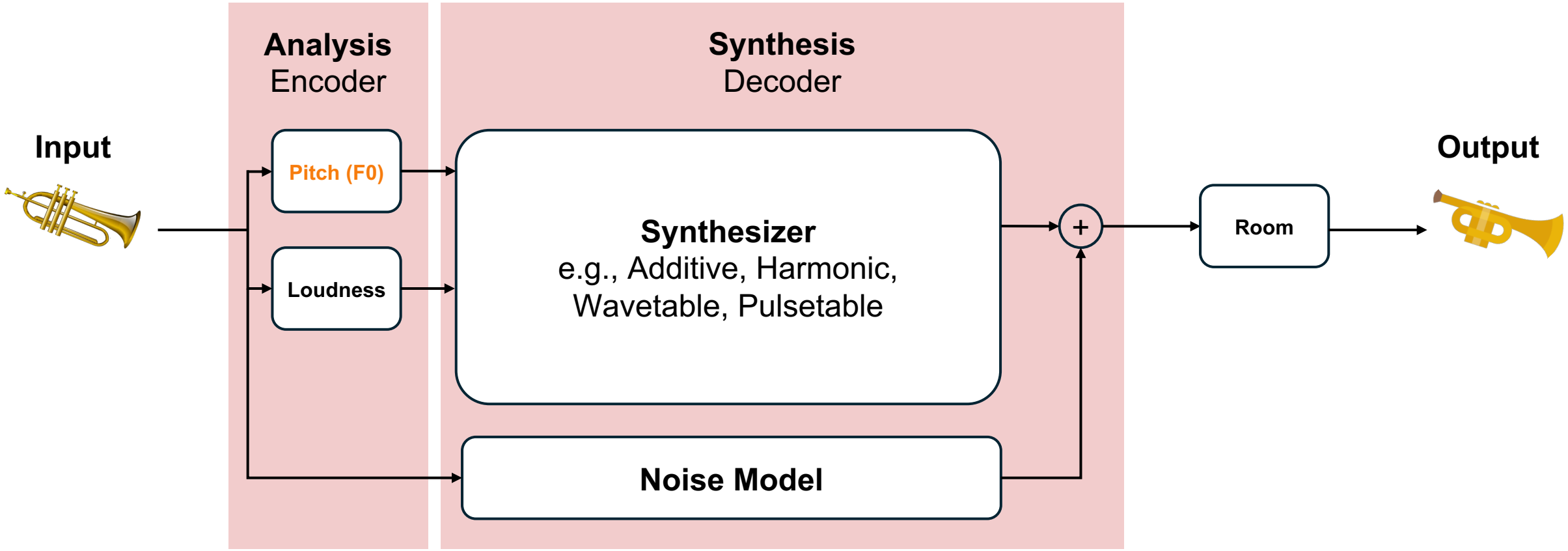
Differentiable Digital Signal Processing (DDSP)

Synthesis



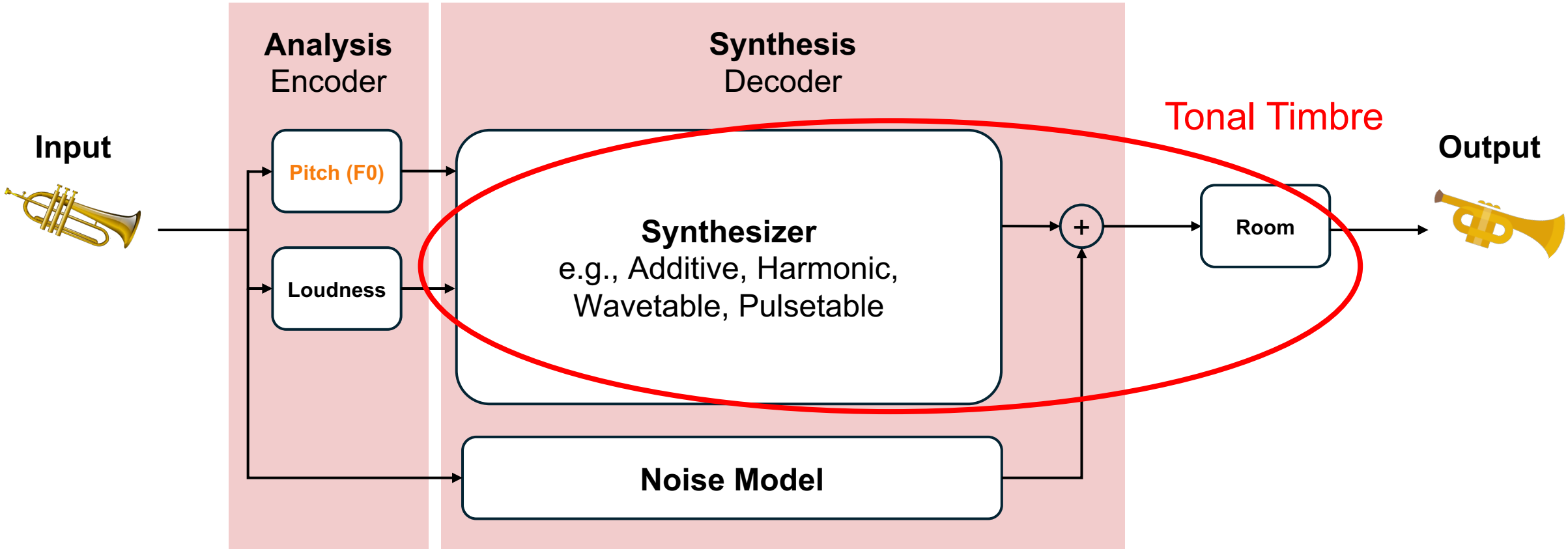
Differentiable Digital Signal Processing (DDSP)

Synthesis



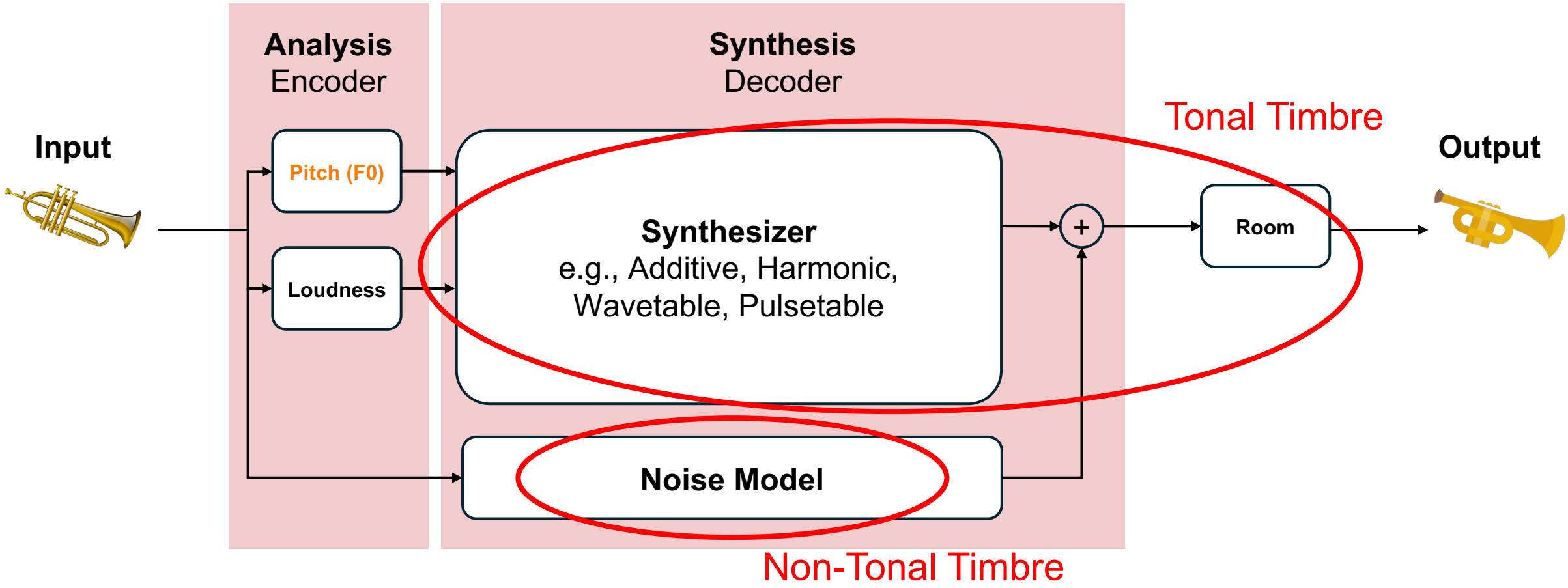
Differentiable Digital Signal Processing (DDSP)

Synthesis



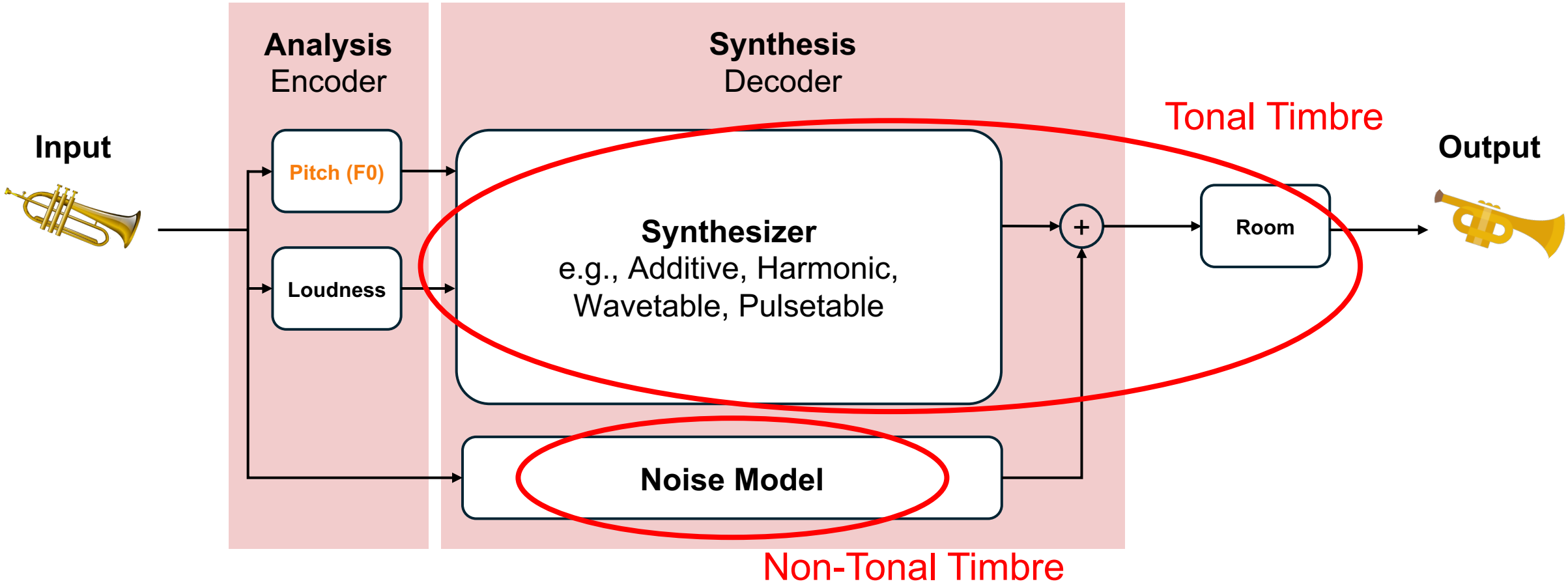
Differentiable Digital Signal Processing (DDSP)

Synthesis



Differentiable Digital Signal Processing (DDSP)

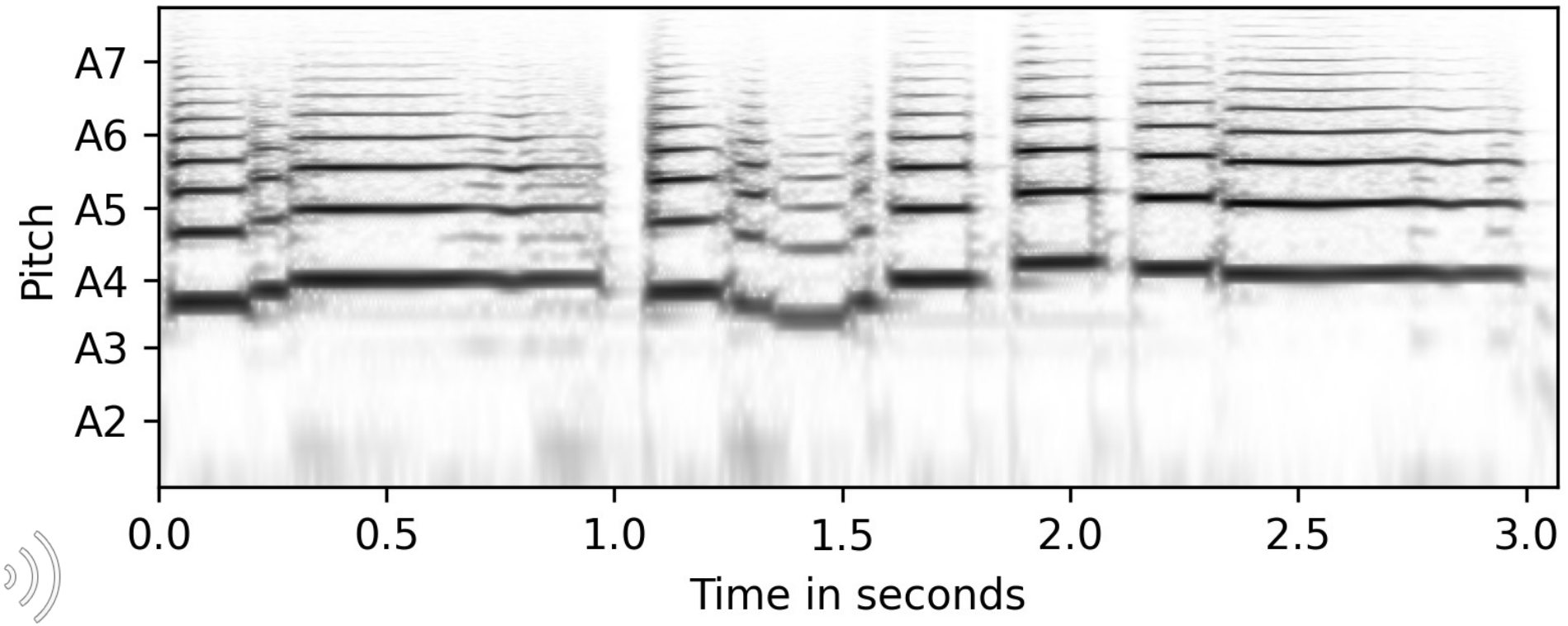
Synthesis



Wind Instrument Synthesis

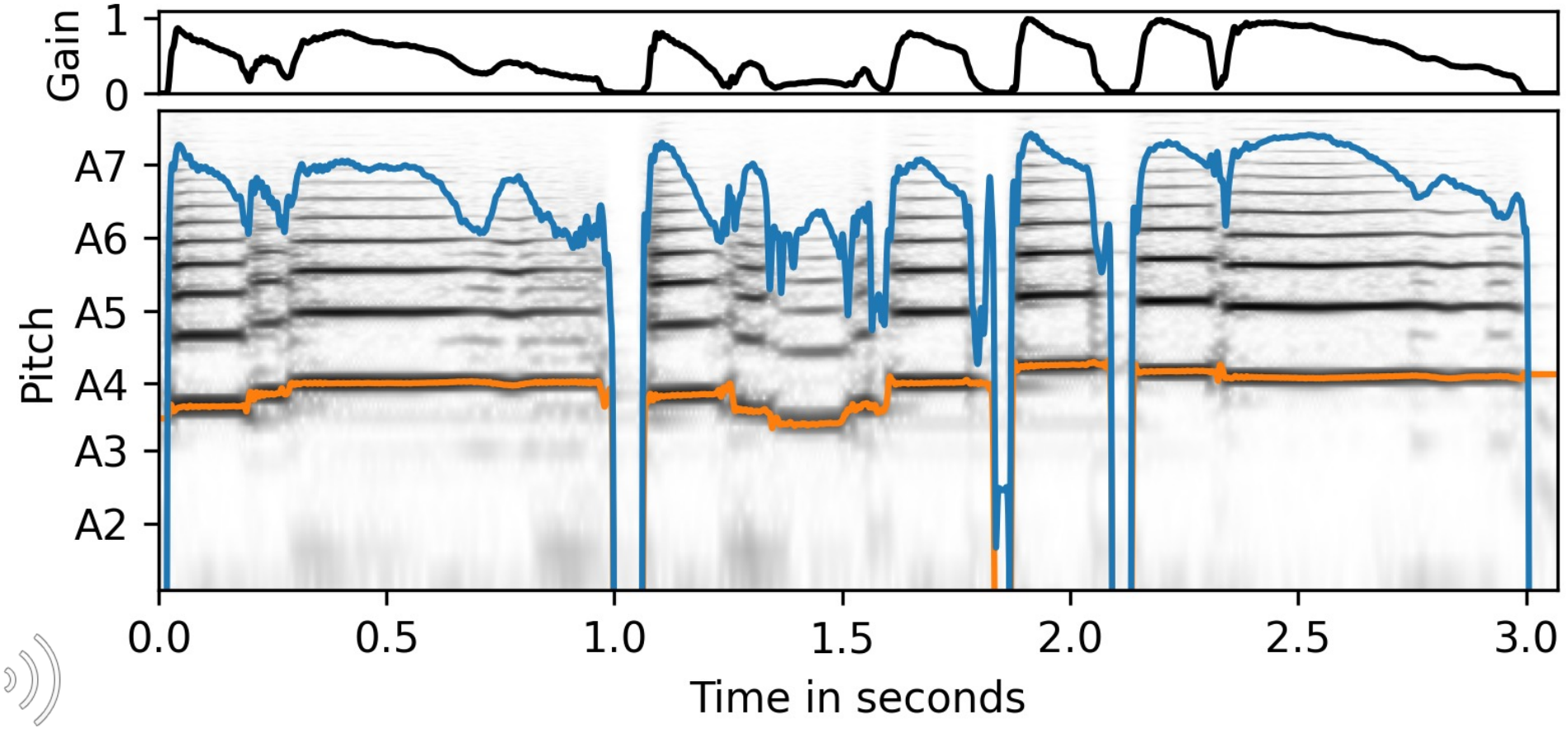
Input Signal

Dittmar, Zeitler, Balke, Schwär, Müller: PULSE-IT: Lightweight and Expressive Synthesis of Wind Instrument Playing. Late-Breaking Demo ISMIR, 2025. https://audiolabs-erlangen.de/resources/MIR/2025_DittmarZBSM_WindInstrumentSynth_ISMIR-LBD



Wind Instrument Synthesis

Extract Control Signals



Loudness

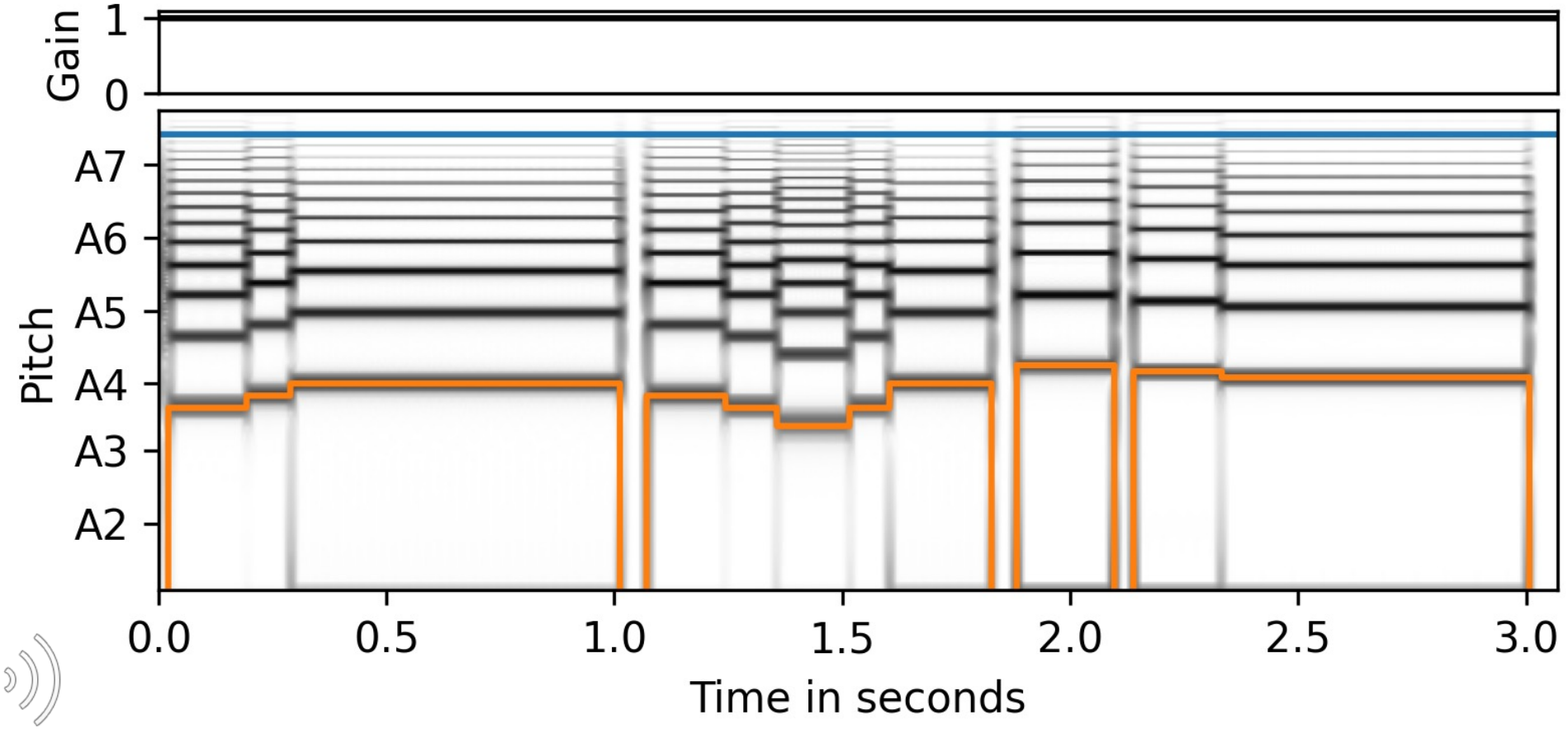
Timbre

Pitch (F0)

© AudioLabs, 2026

Wind Instrument Synthesis

Generated Control Signals



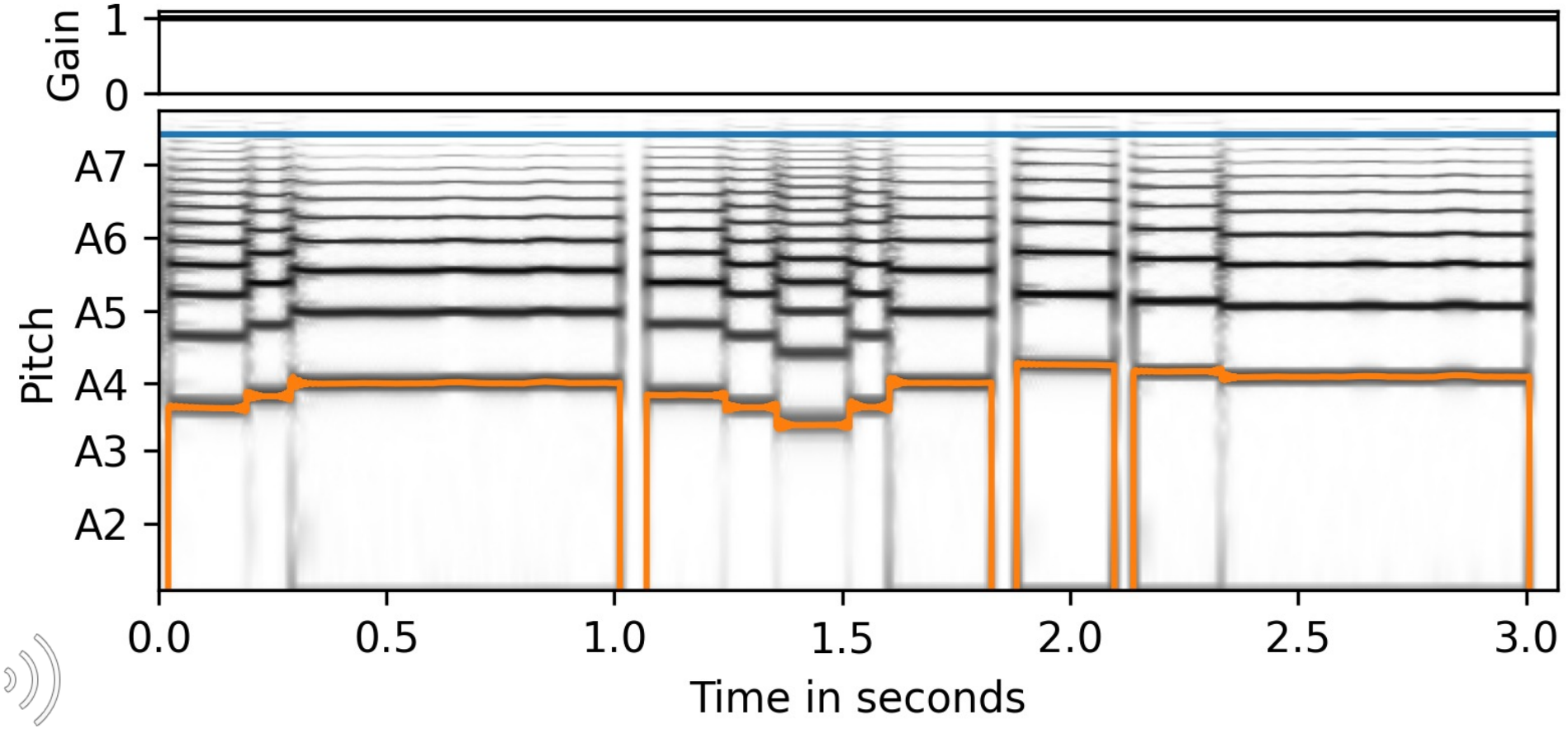
Loudness

Timbre

Pitch (F0)

Wind Instrument Synthesis

Generated Control Signals with Pitch Fluctuations



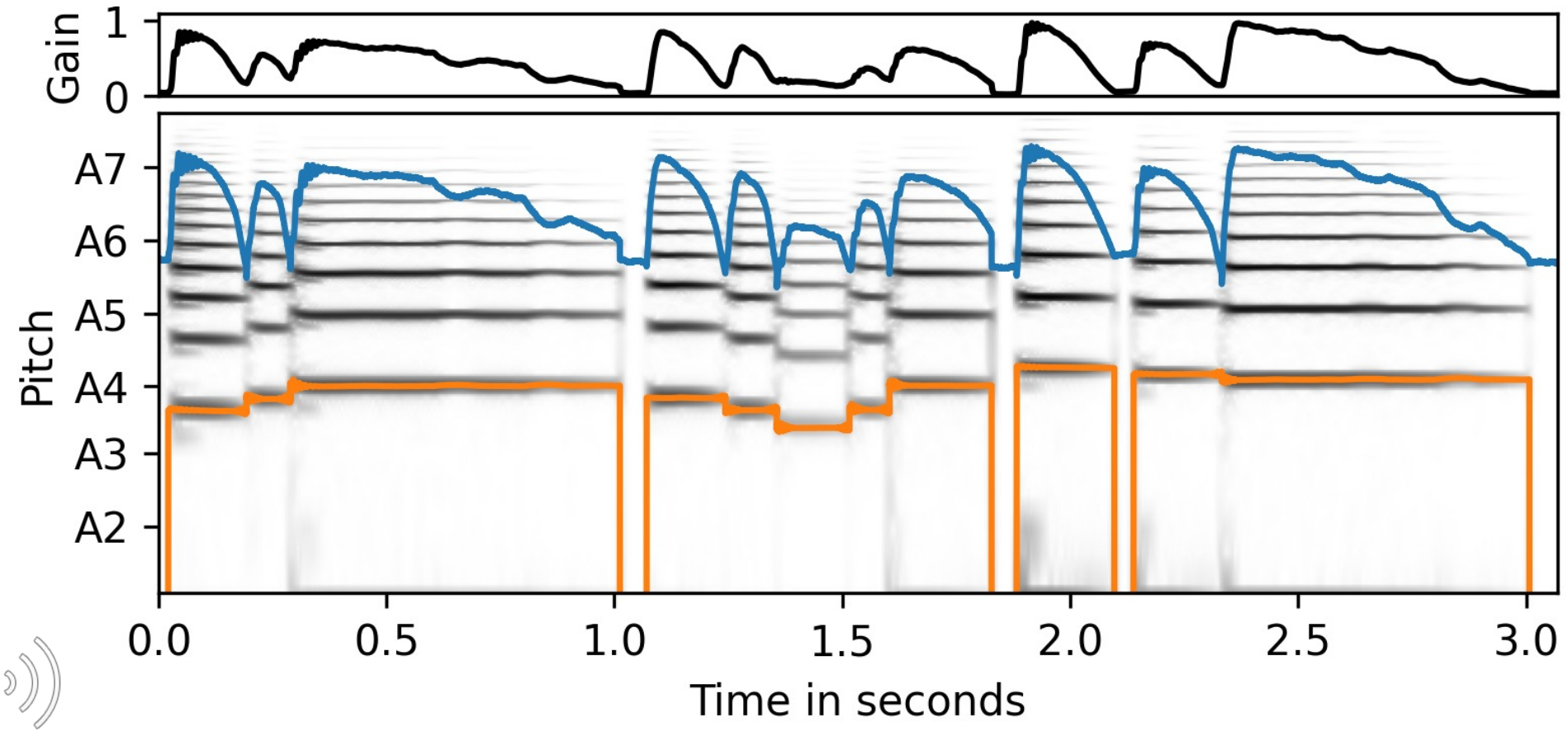
Loudness

Timbre

Pitch (F0)

Wind Instrument Synthesis

Generated Control Signals with Pitch & Loudness Fluctuations



Loudness

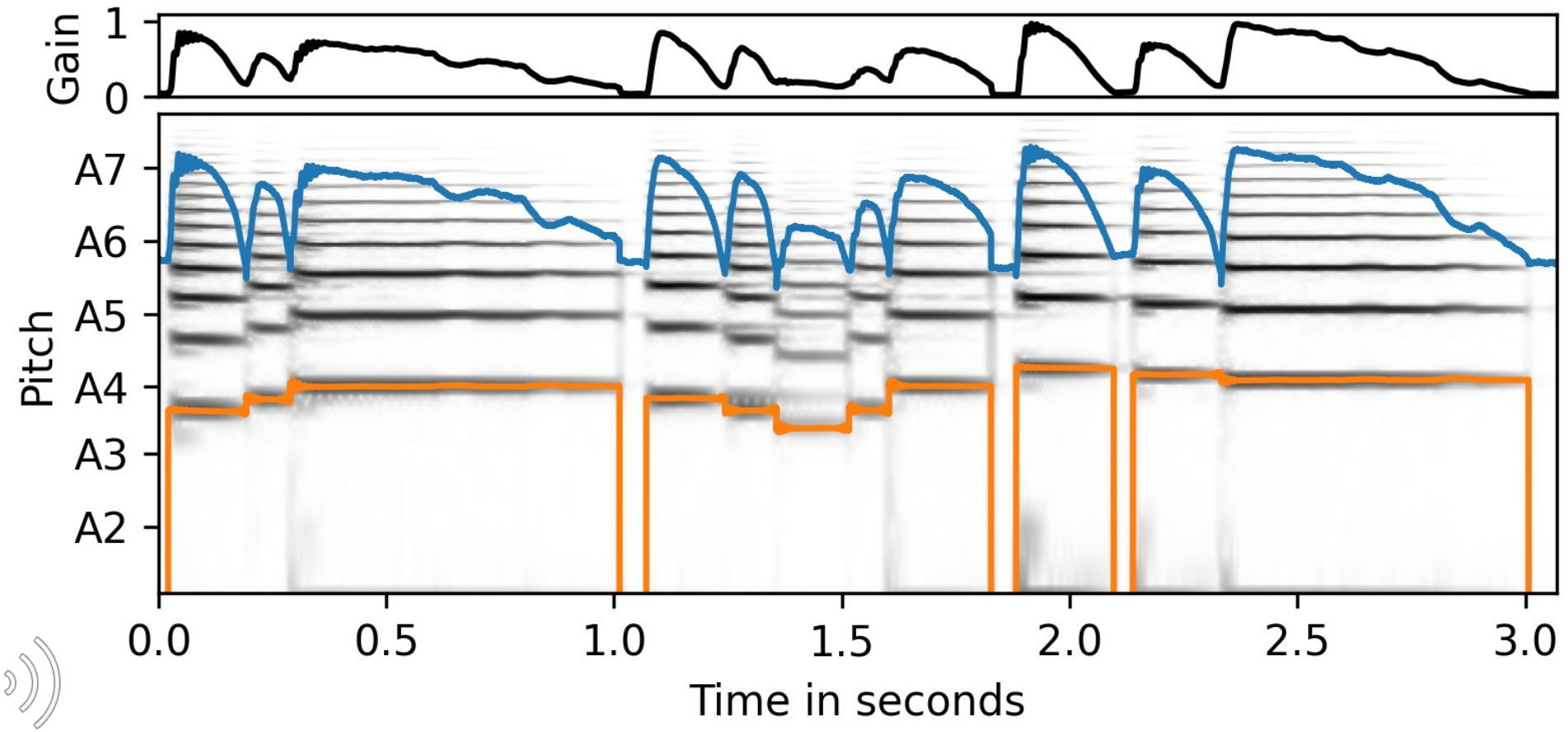
Timbre

Pitch (F0)

© AudioLabs, 2026

Wind Instrument Synthesis

Generated Control Signals with Fluctuations and Room Acoustics



Loudness

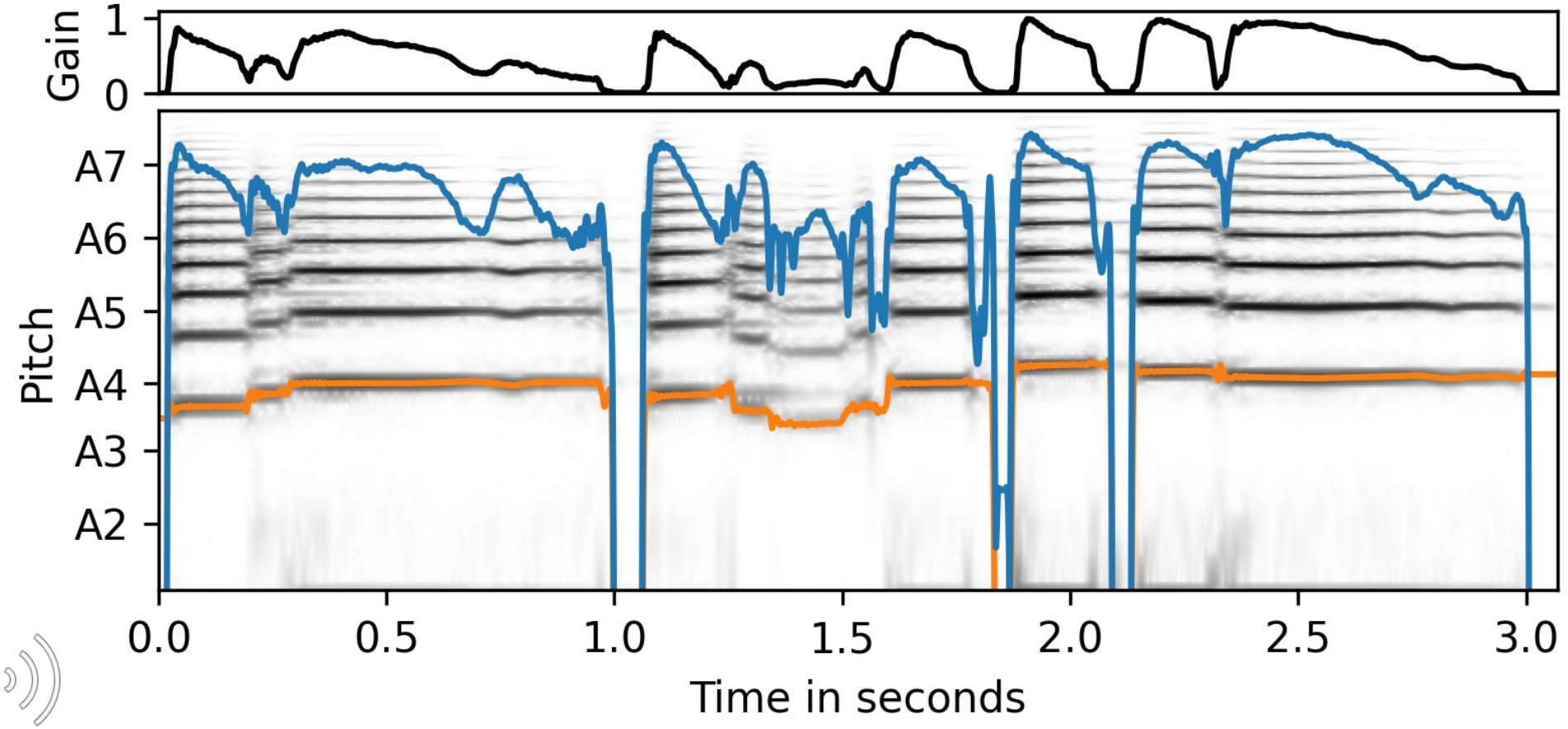
Timbre

Pitch (F0)

© AudioLabs, 2026

Wind Instrument Synthesis

Resynthesized Trumpet



Loudness

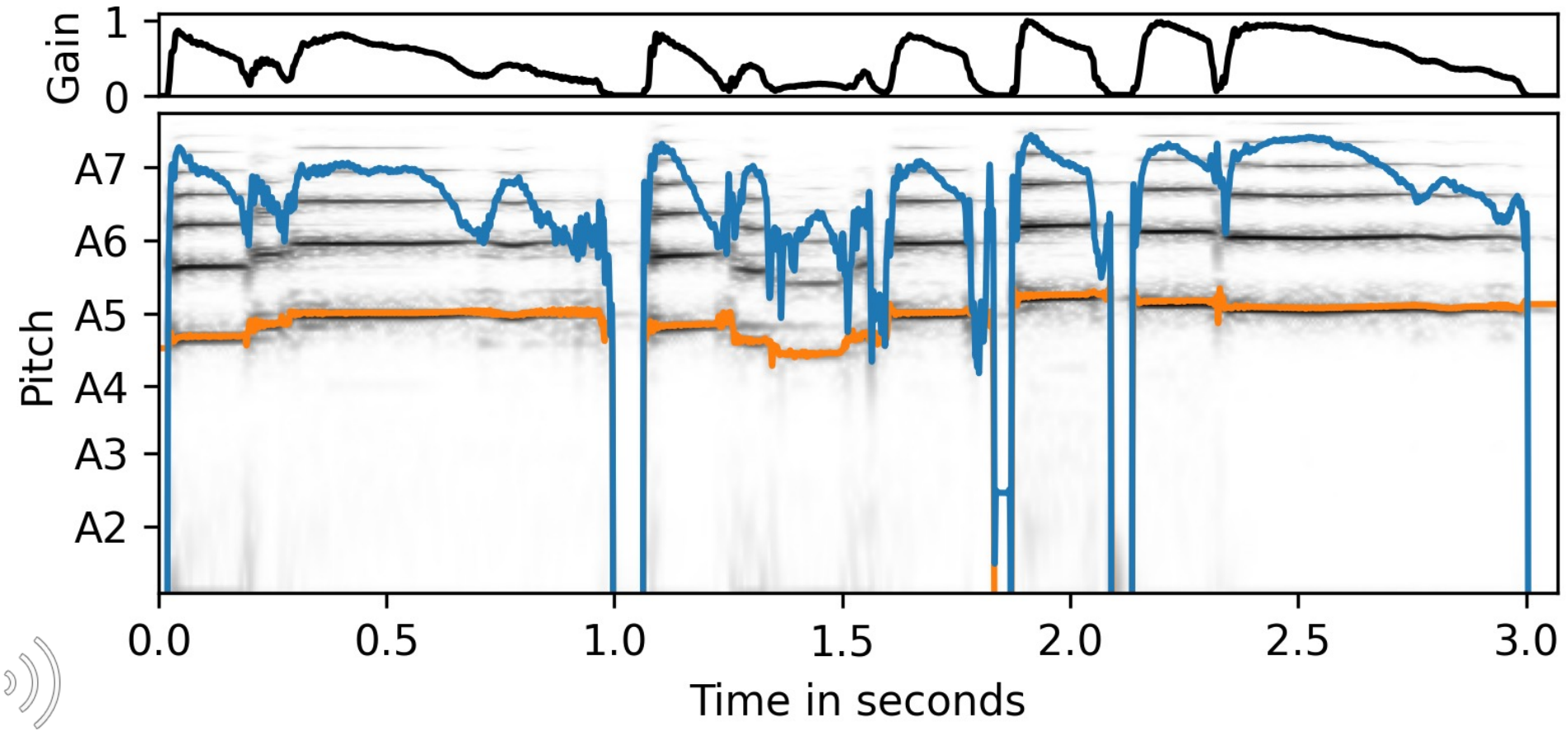
Timbre

Pitch (F0)

© AudioLabs, 2026

Wind Instrument Synthesis

Resynthesized Trumpet (One Octave Higher)



Loudness

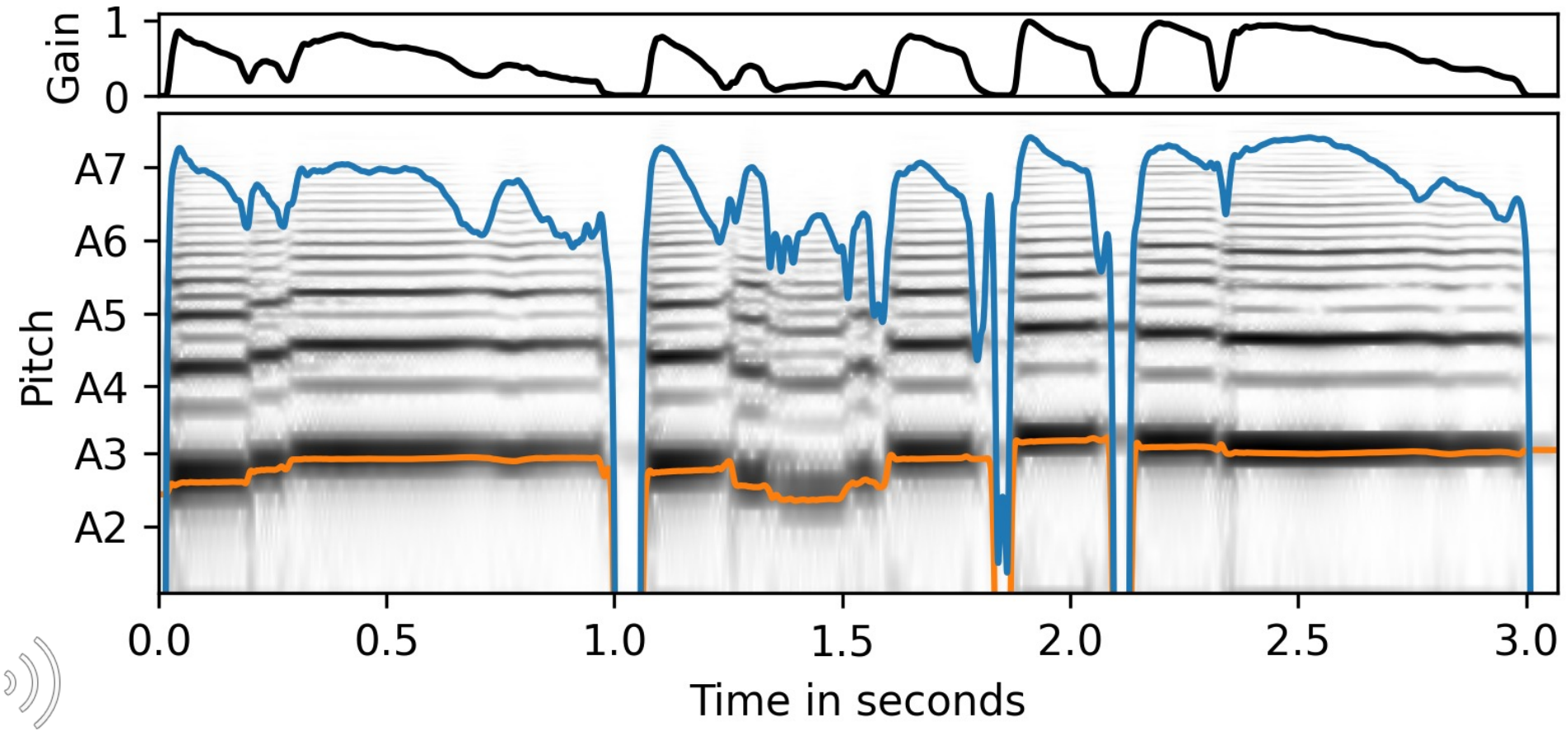
Timbre

Pitch (F0)

© AudioLabs, 2026

Wind Instrument Synthesis

Resynthesized Clarinet (One Octave Lower)



Loudness

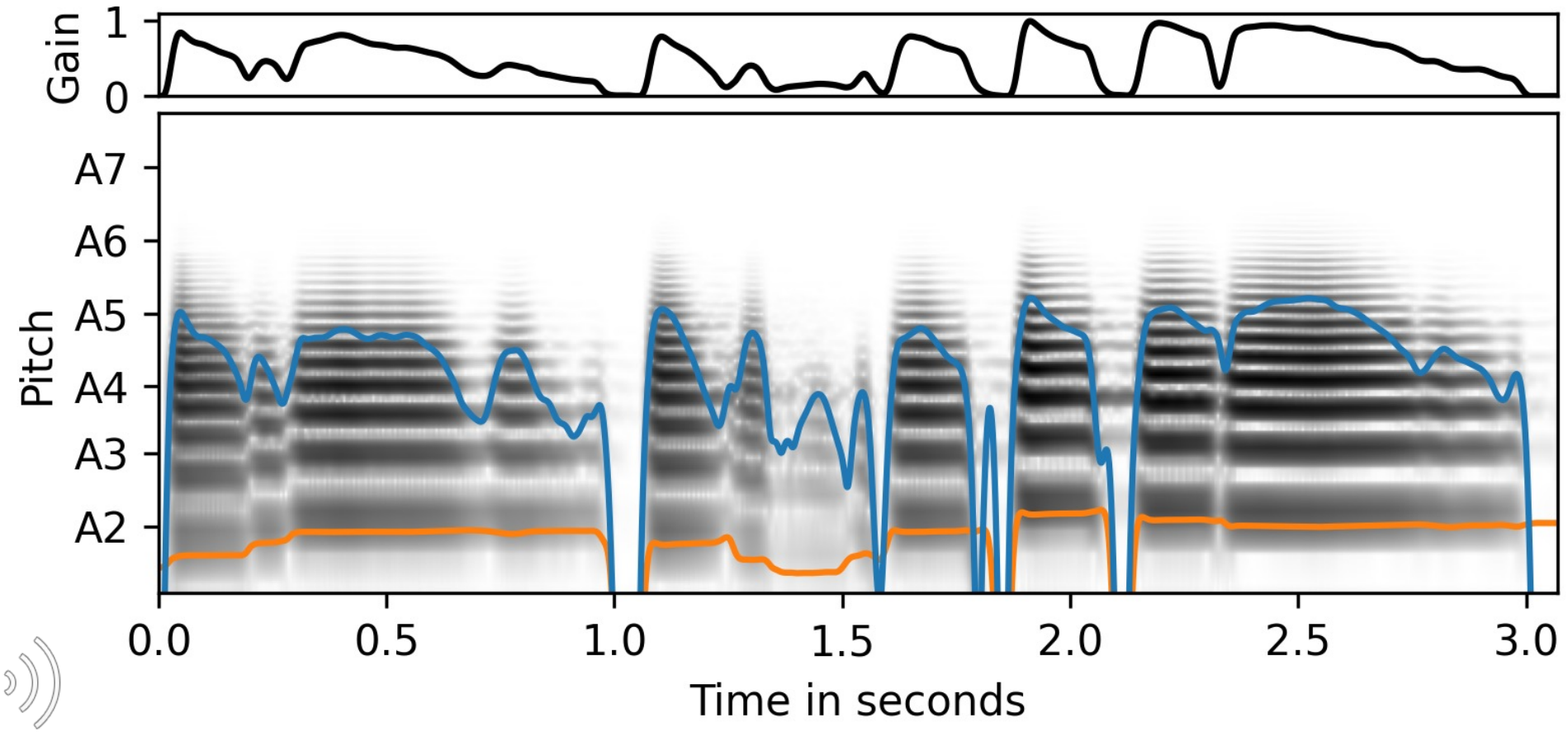
Timbre

Pitch (F0)

© AudioLabs, 2026

Wind Instrument Synthesis

Resynthesized Tuba (Two Octaves Lower)



Loudness

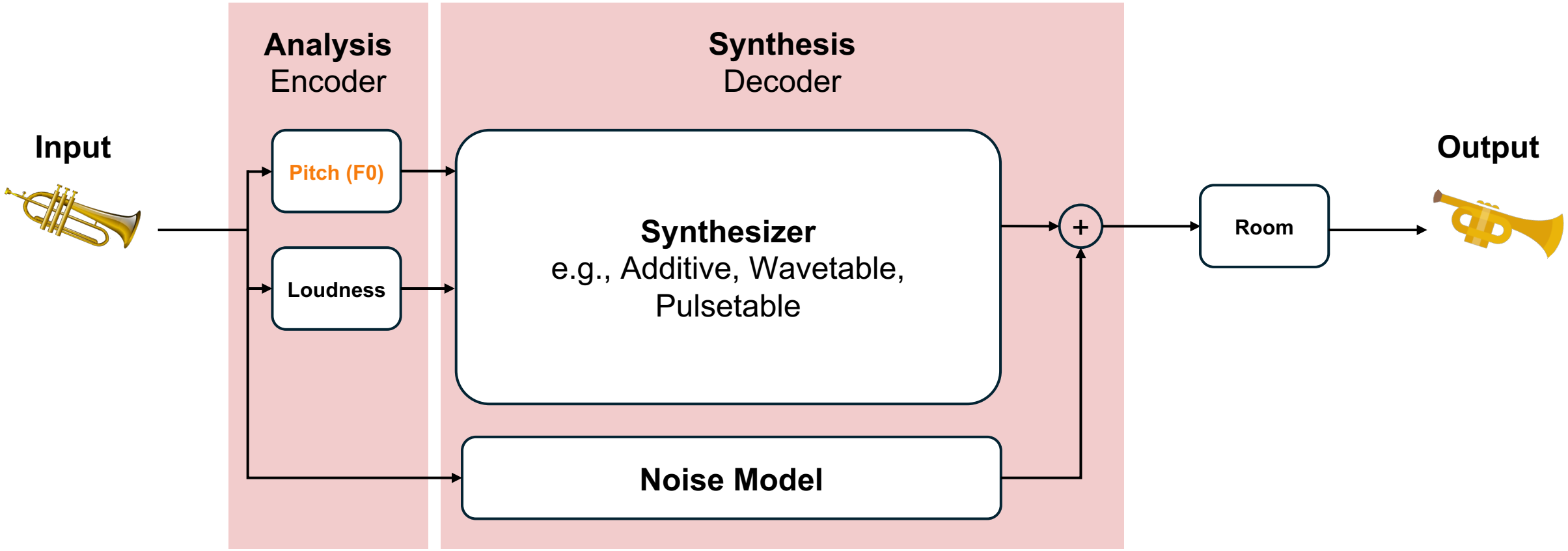
Timbre

Pitch (F0)

© AudioLabs, 2026

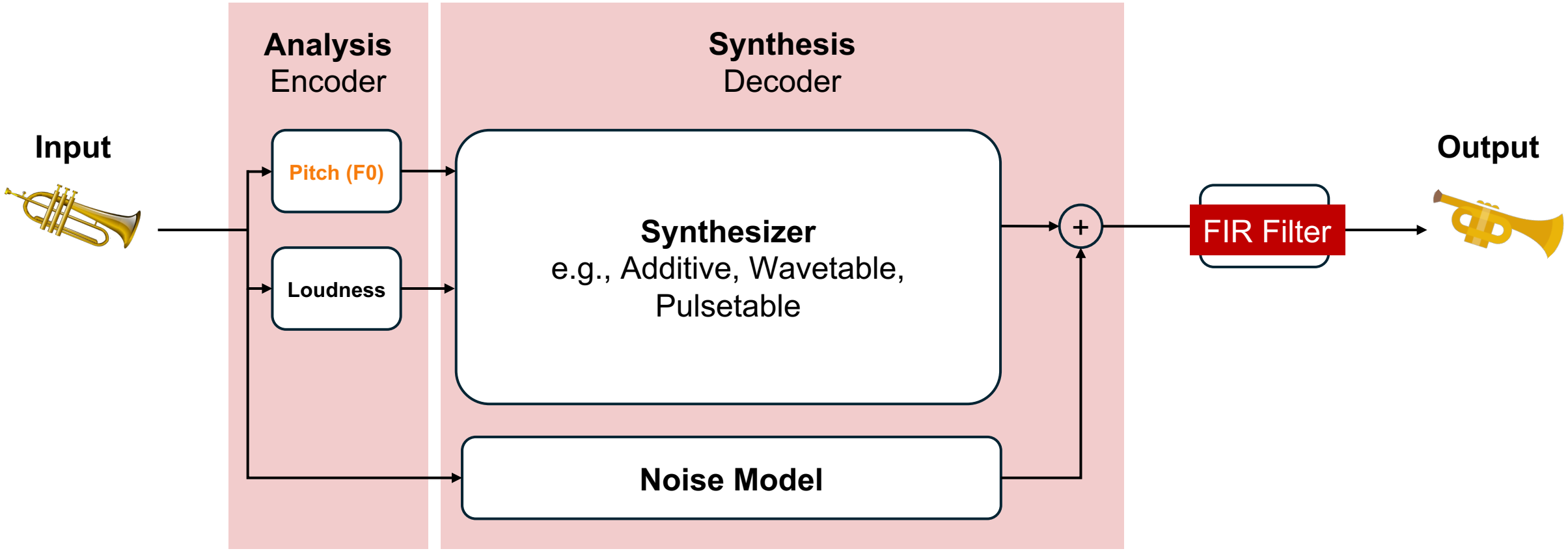
Differentiable Digital Signal Processing (DDSP)

Synthesis



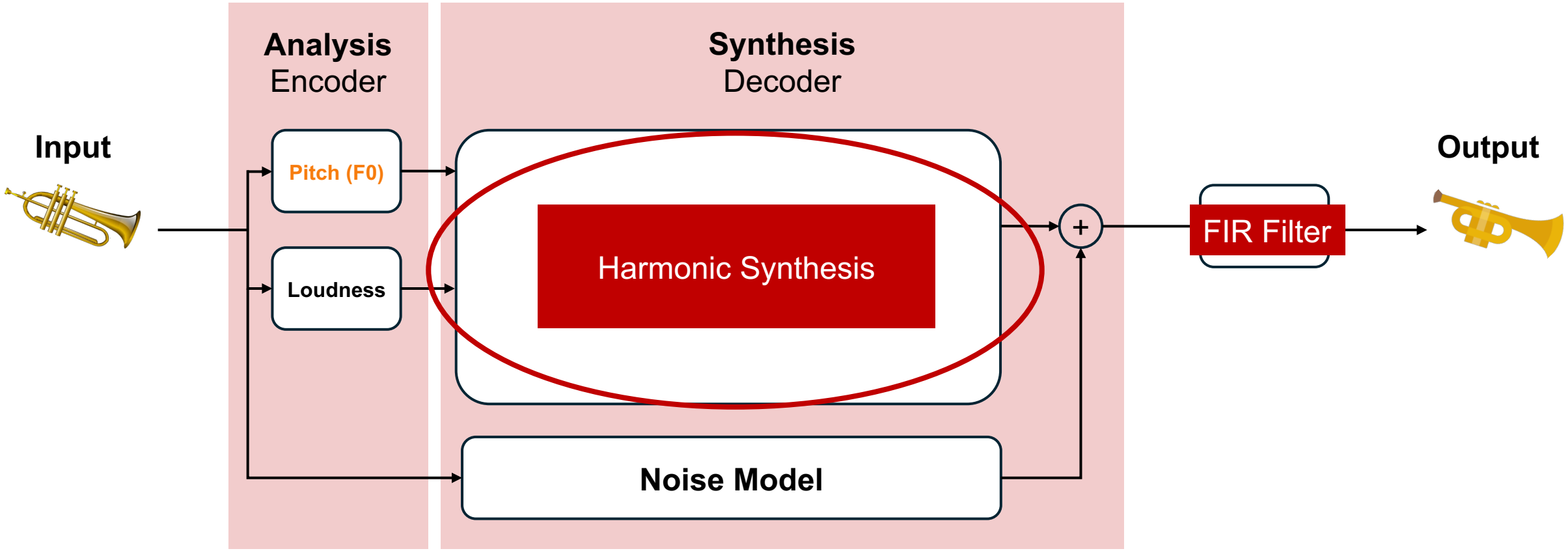
Differentiable Digital Signal Processing (DDSP)

Synthesis



Differentiable Digital Signal Processing (DDSP)

Synthesis

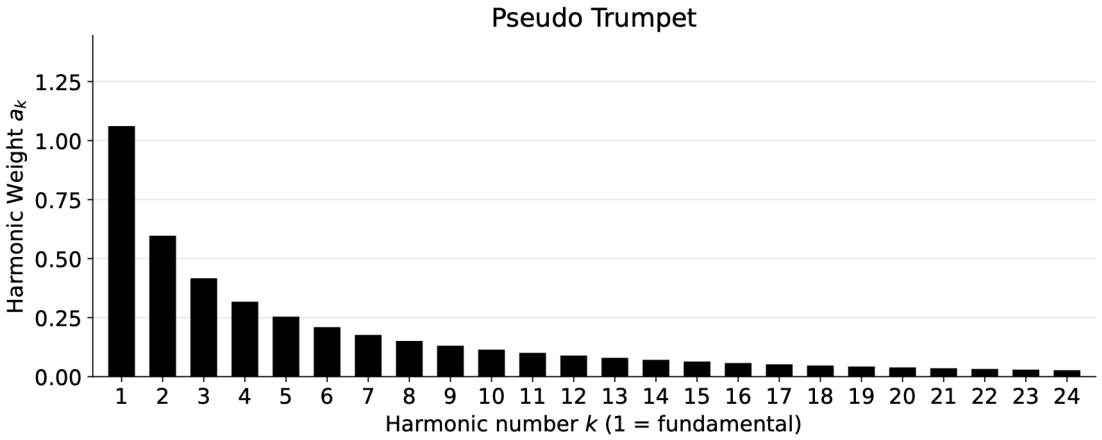


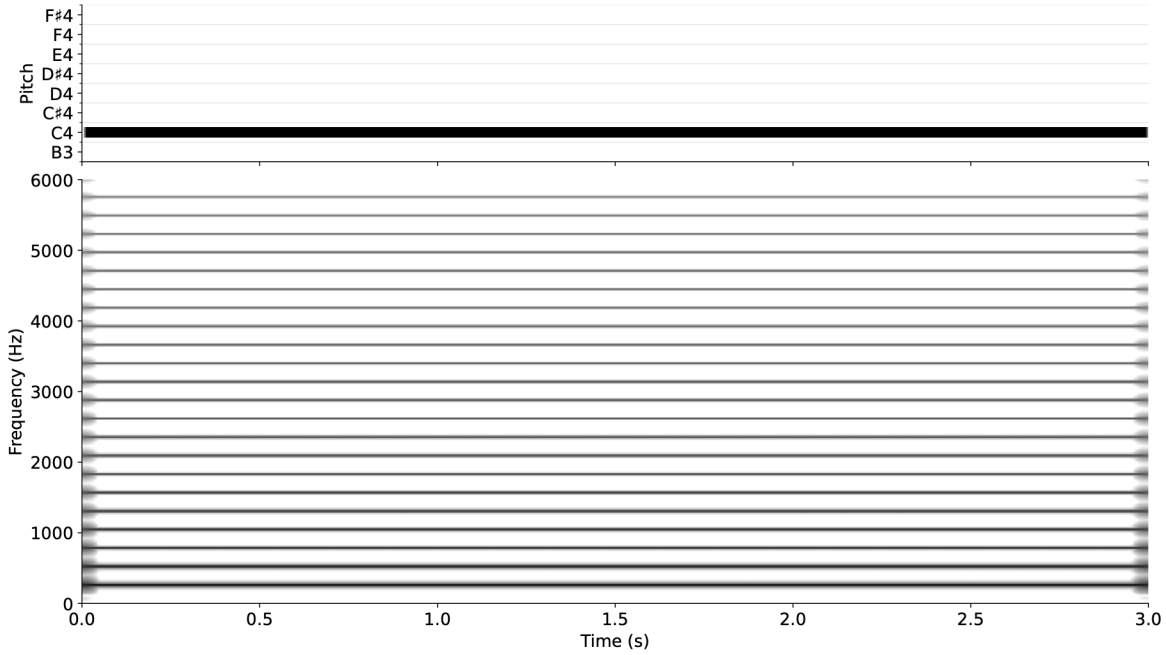
Differentiable Digital Signal Processing (DDSP)

Harmonic Synthesis

$$\hat{y}(t) = \sum_{k=1}^{K} a_k \sin\left(2\pi k f_0 \frac{t}{F_s}\right)$$

f_0: Fundamental frequency
a_k: Harmonic weights
F_s: Sampling rate





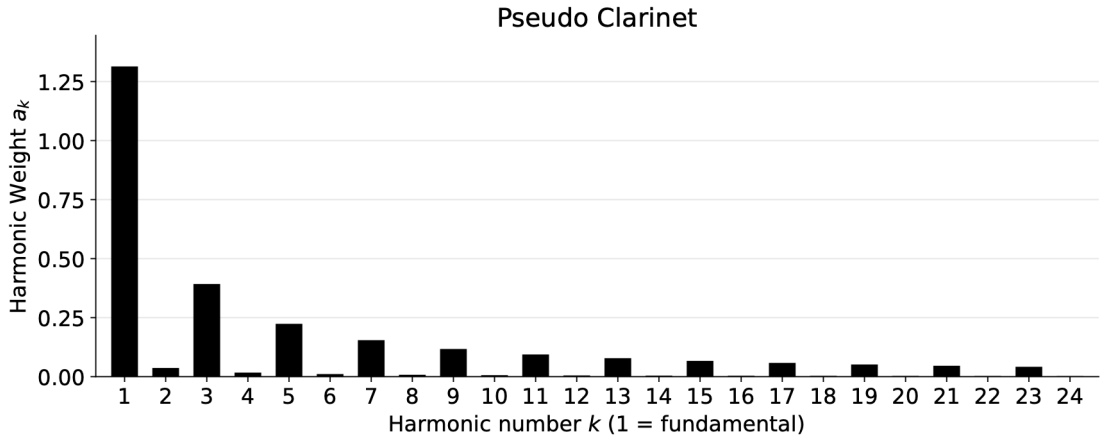
© AudioLabs, 2026

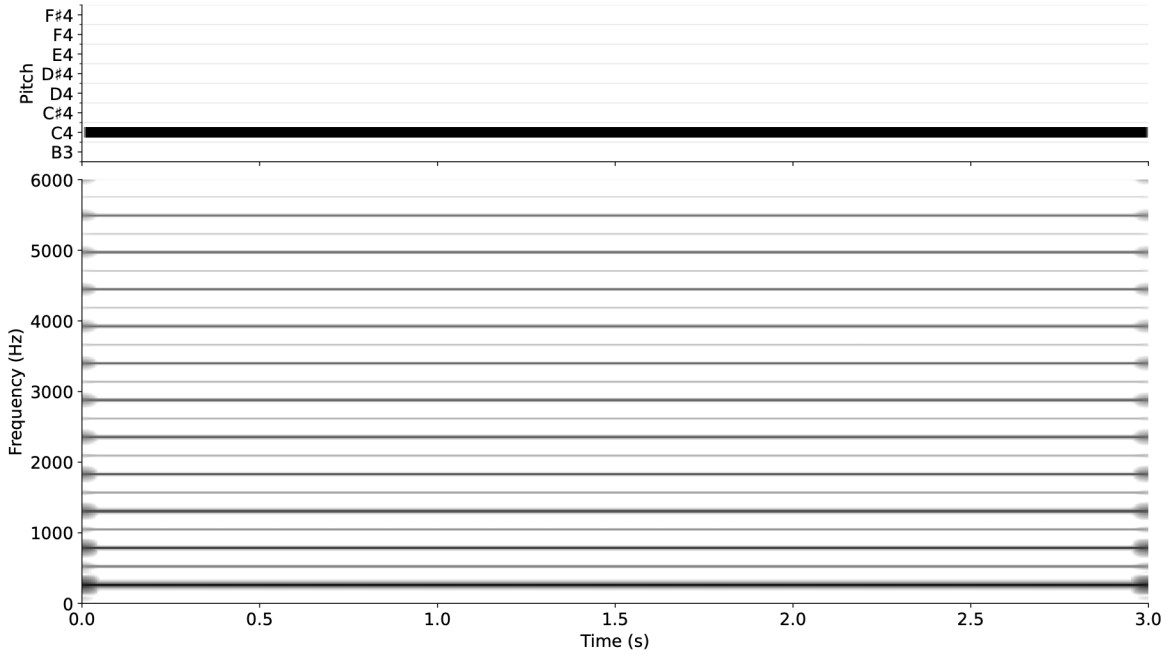
Differentiable Digital Signal Processing (DDSP)

Harmonic Synthesis

$$\hat{y}(t) = \sum_{k=1}^{K} a_k \sin\left(2\pi k f_0 \frac{t}{F_s}\right)$$

f_0: Fundamental frequency
a_k: Harmonic weights
F_s: Sampling rate





© AudioLabs, 2026

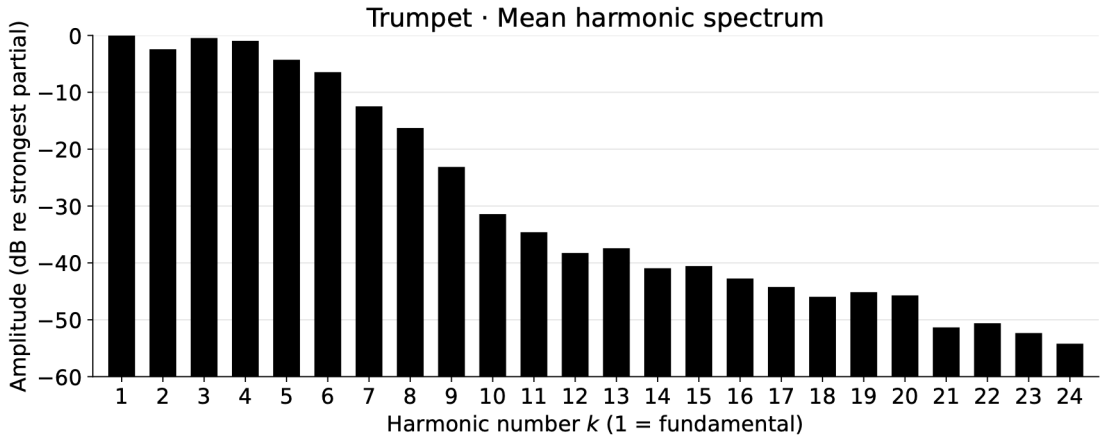
Differentiable Digital Signal Processing (DDSP)

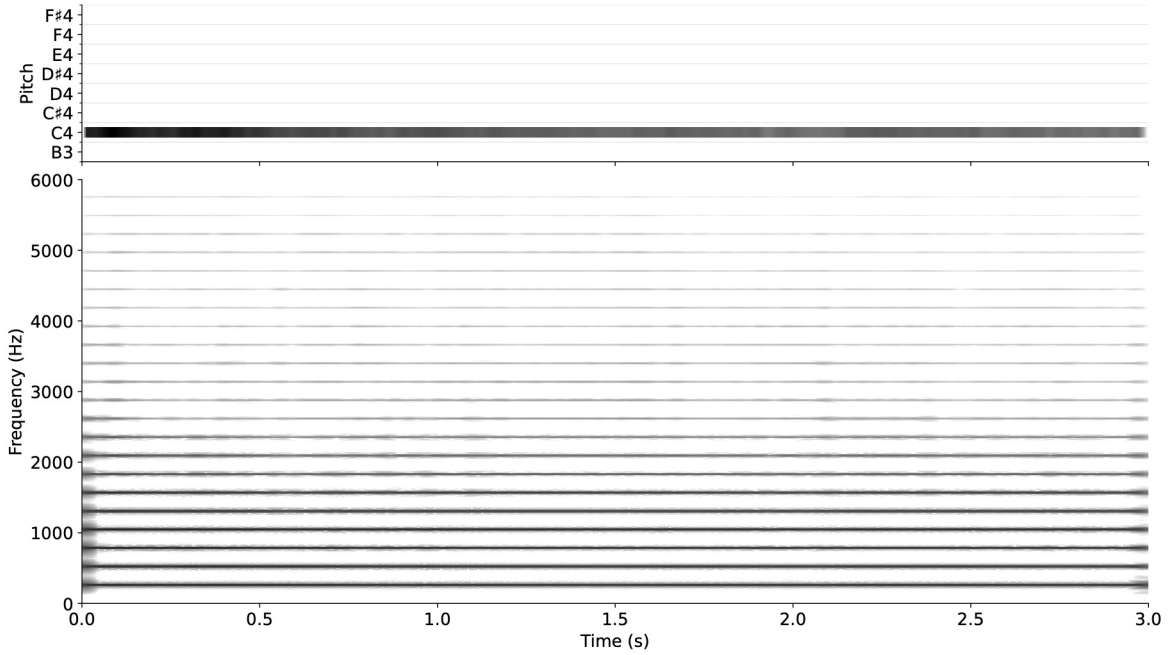
Harmonic Synthesis

Harmonic series + loudness envelope estimated from “clean” trumpet recording

$$\hat{y}(t) = \sum_{k=1}^{K} a_k \sin\left(2\pi k f_0 \frac{t}{F_s}\right)$$

f_0: Fundamental frequency
a_k: Harmonic weights
F_s: Sampling rate





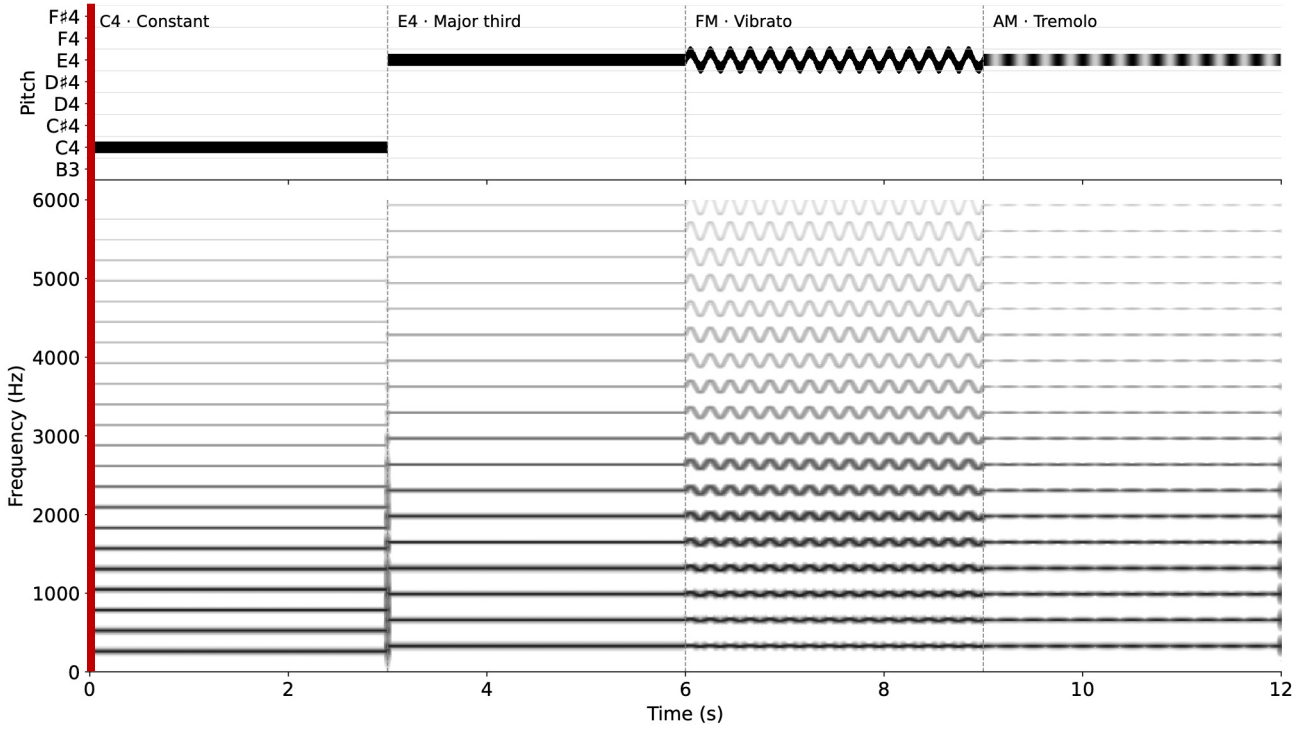
© AudioLabs, 2026

Differentiable Digital Signal Processing (DDSP)

Harmonic Synthesis

Harmonic weights and fundamental frequencies can be time-varying:

$$a_k \longrightarrow a_k(t), \qquad f_0 \longrightarrow f_0(t)$$



Differentiable Digital Signal Processing (DDSP)

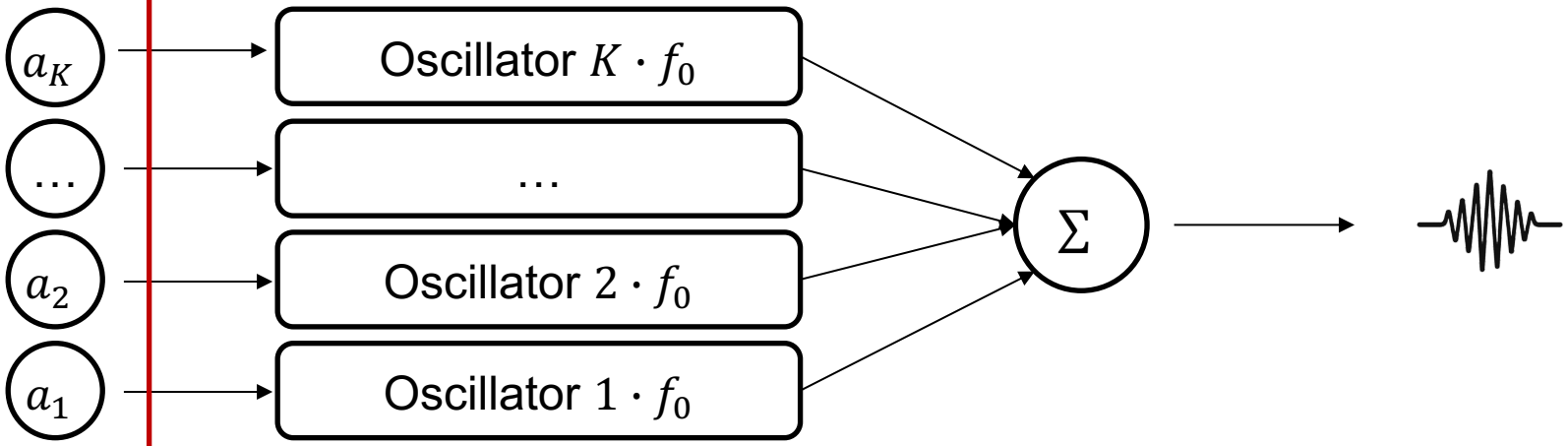
Harmonic Synthesis

$$\hat{y}(t) = \sum_{k=1}^{K} a_k \sin\left(2\pi k f_0 \frac{t}{f_s}\right)$$

f_0: Fundamental frequency
a_k: Harmonic weights
f_s: Sampling rate

Harmonic weights can be treated as “gains” for the oscillators.
For MSE Loss and constant f_0 :

$$\frac{\partial \mathcal{L}}{\partial a_{k_0}} = \frac{2}{T} \sum_{t=0}^{T-1} \left(\hat{y}(t) - y(t)\right) \sin\left(2\pi k_0 f_0 \frac{t}{f_s}\right)$$



Differentiable Digital Signal Processing (DDSP)
Harmonic Synthesis

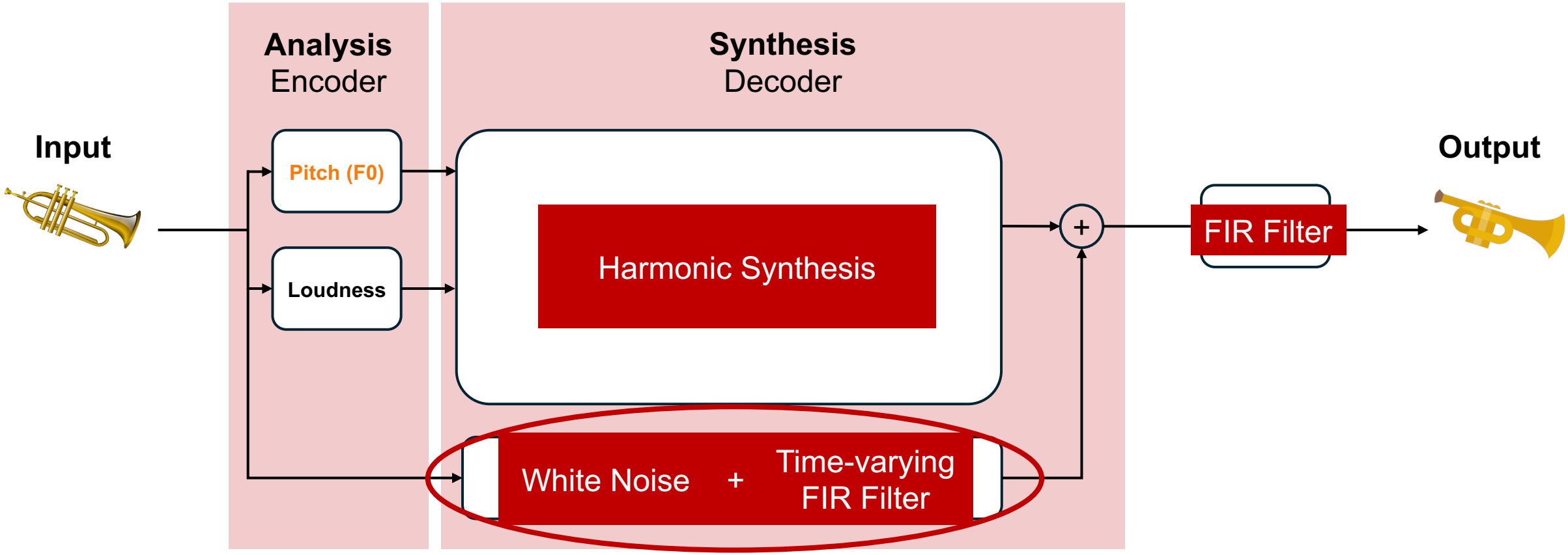
Limitations:

- One fundamental frequency per voice
- Partials constrained to harmonic frequencies
- Noise and irregular transients require additional components
- Harmonics above Nyquist must be suppressed to avoid aliasing

Alternative synthesis approaches:
FM, Wavetable, Pulsetable, granular

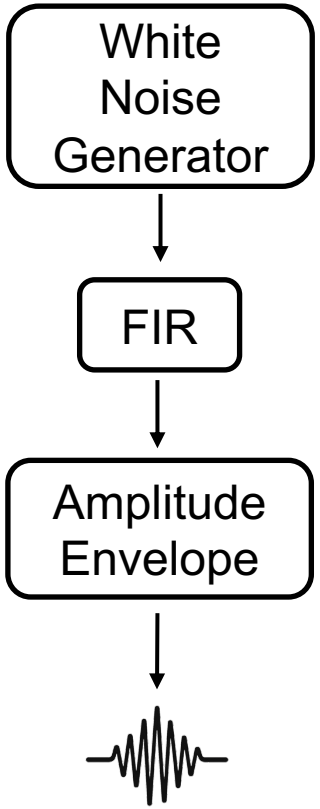
Differentiable Digital Signal Processing (DDSP)

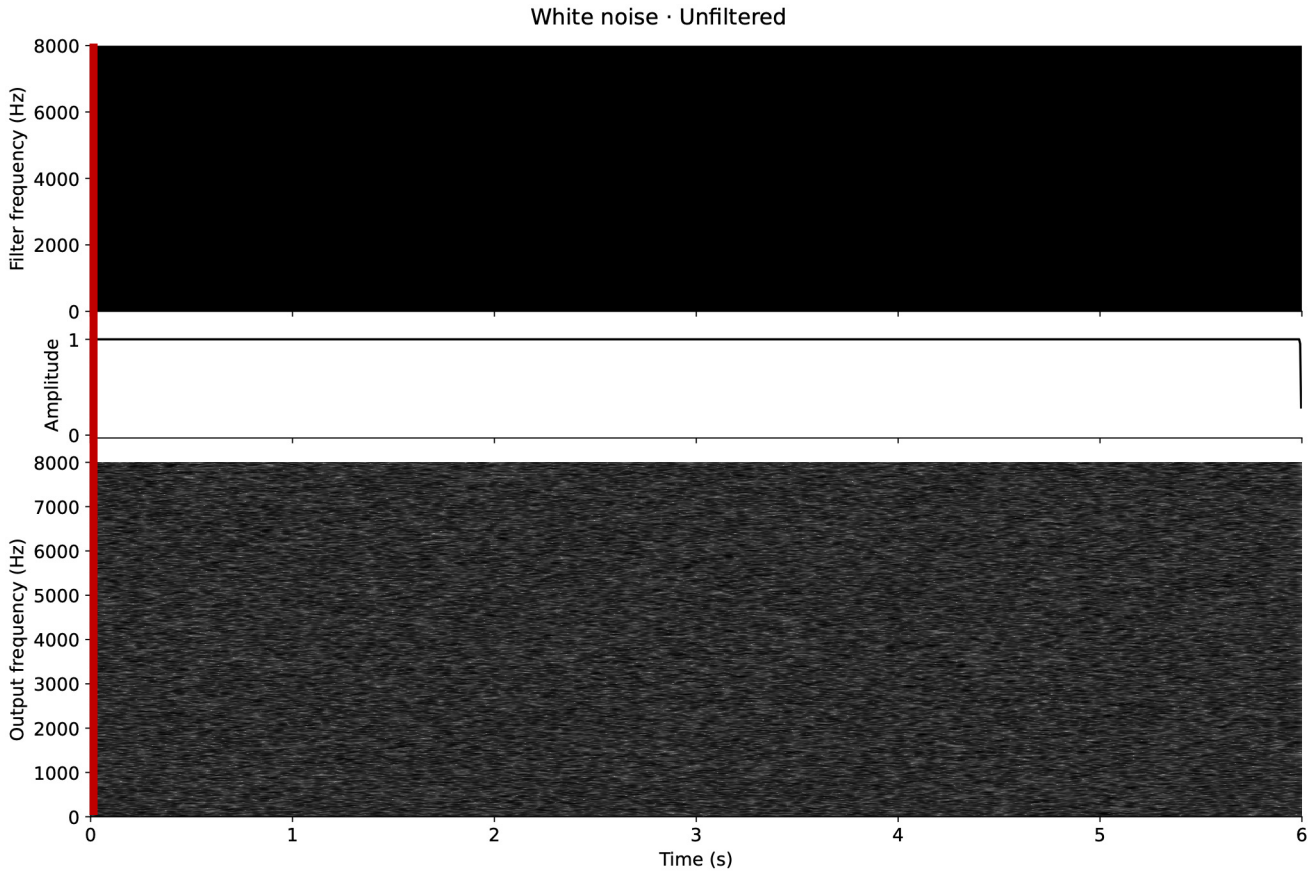
Synthesis



Differentiable Digital Signal Processing (DDSP)

Noise Model



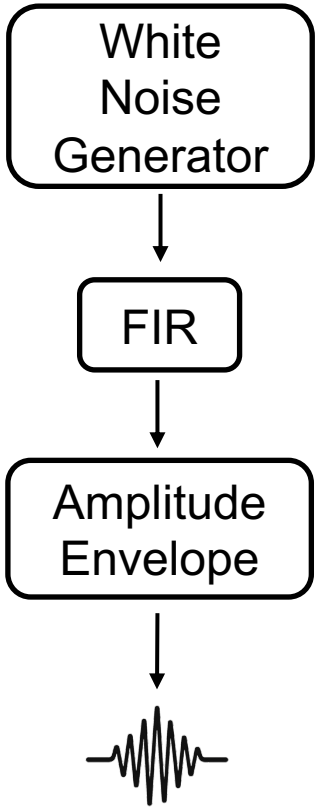


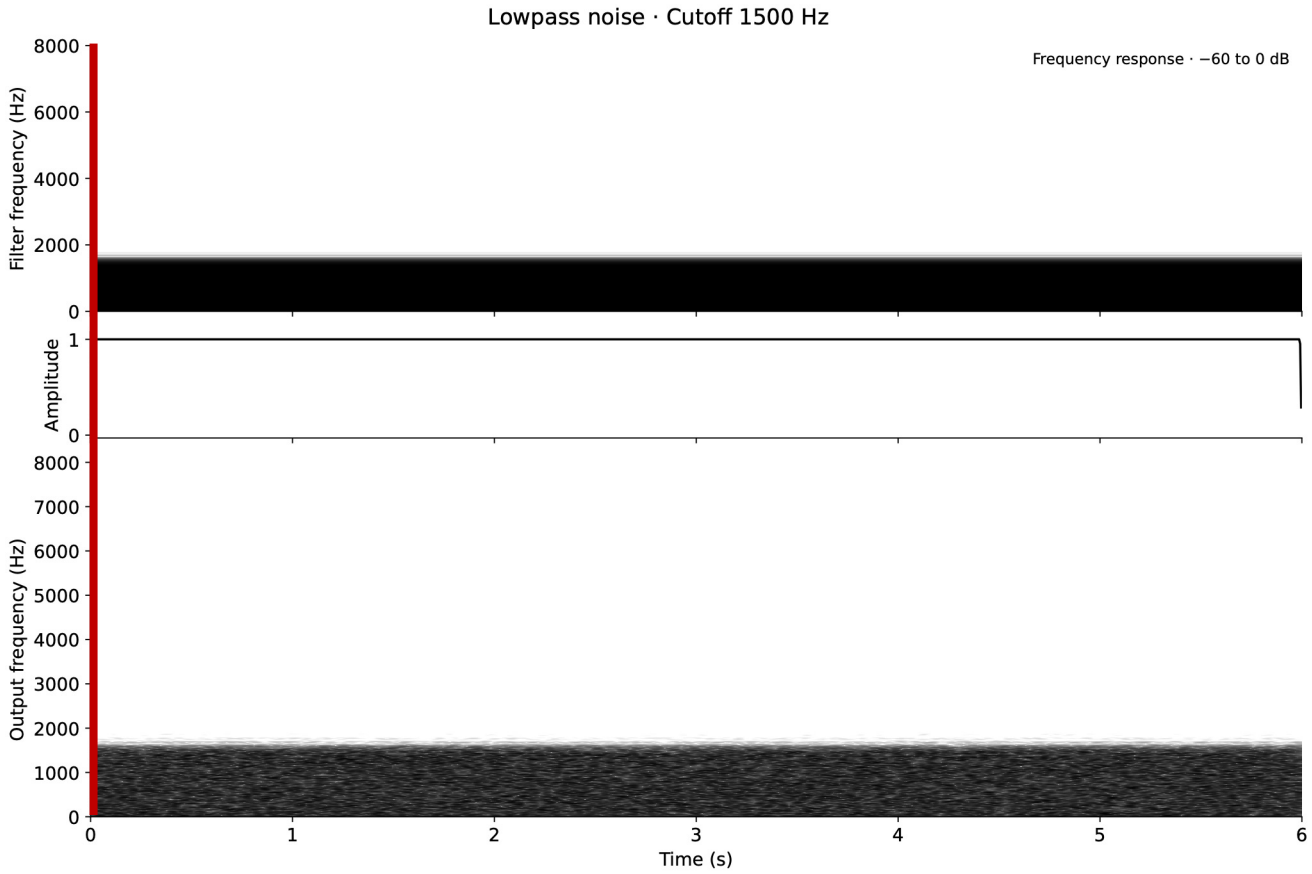
time-invariant allpass

time-invariant amplitude envelope

Differentiable Digital Signal Processing (DDSP)

Noise Model



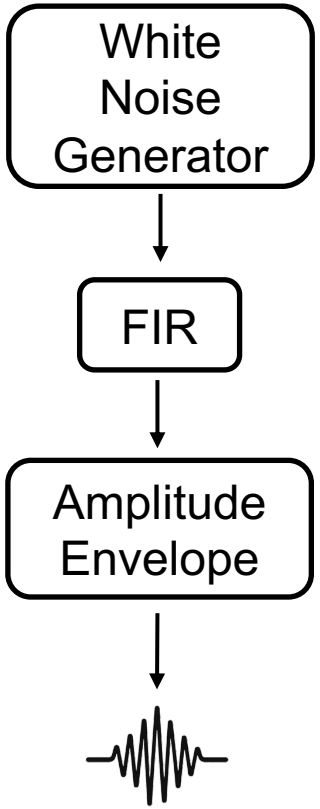


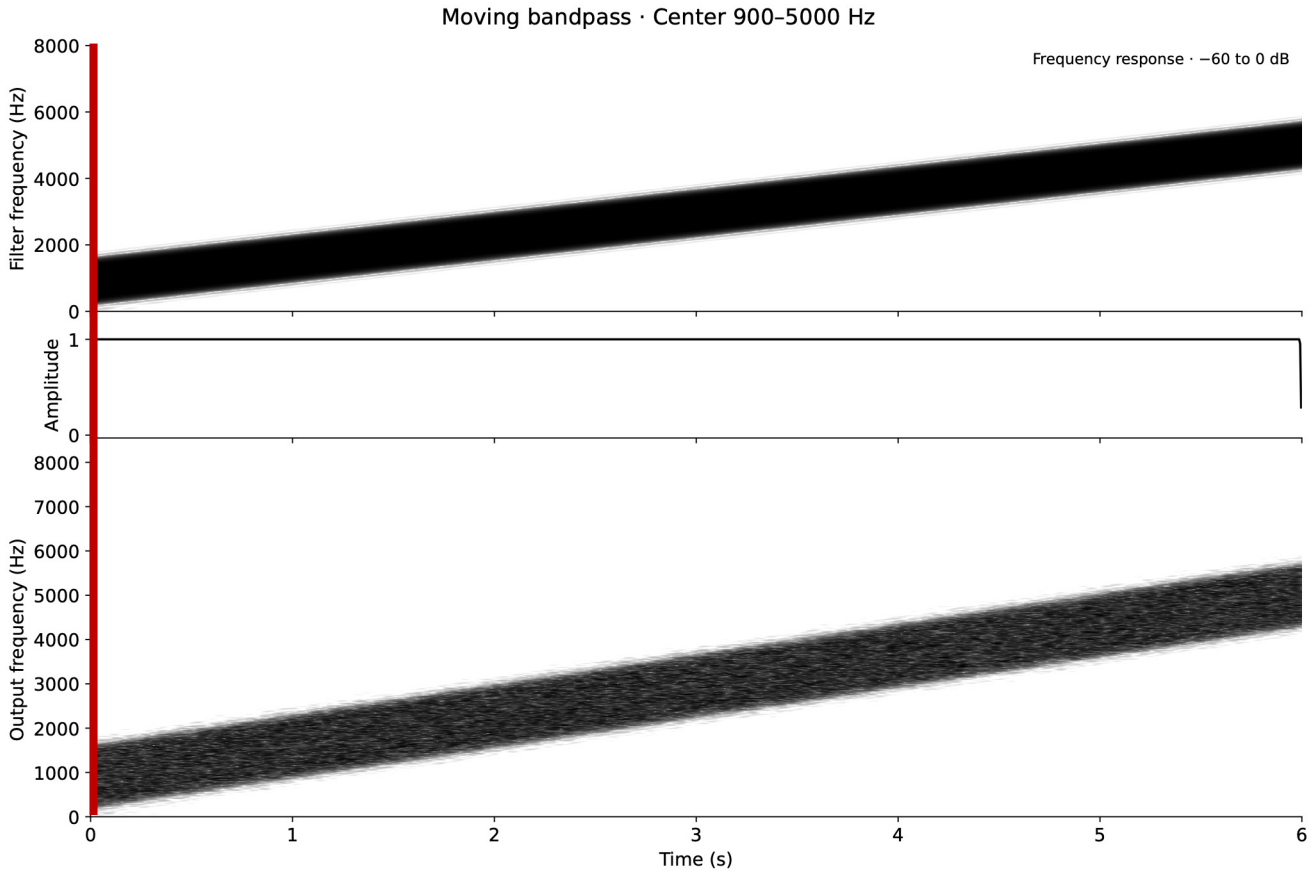
time-invariant lowpass

time-invariant amplitude envelope

Differentiable Digital Signal Processing (DDSP)

Noise Model



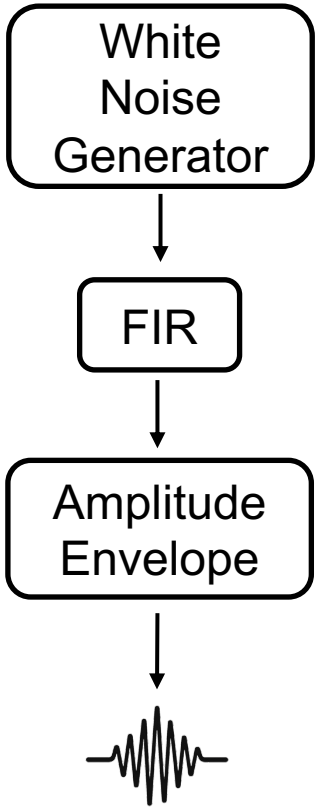


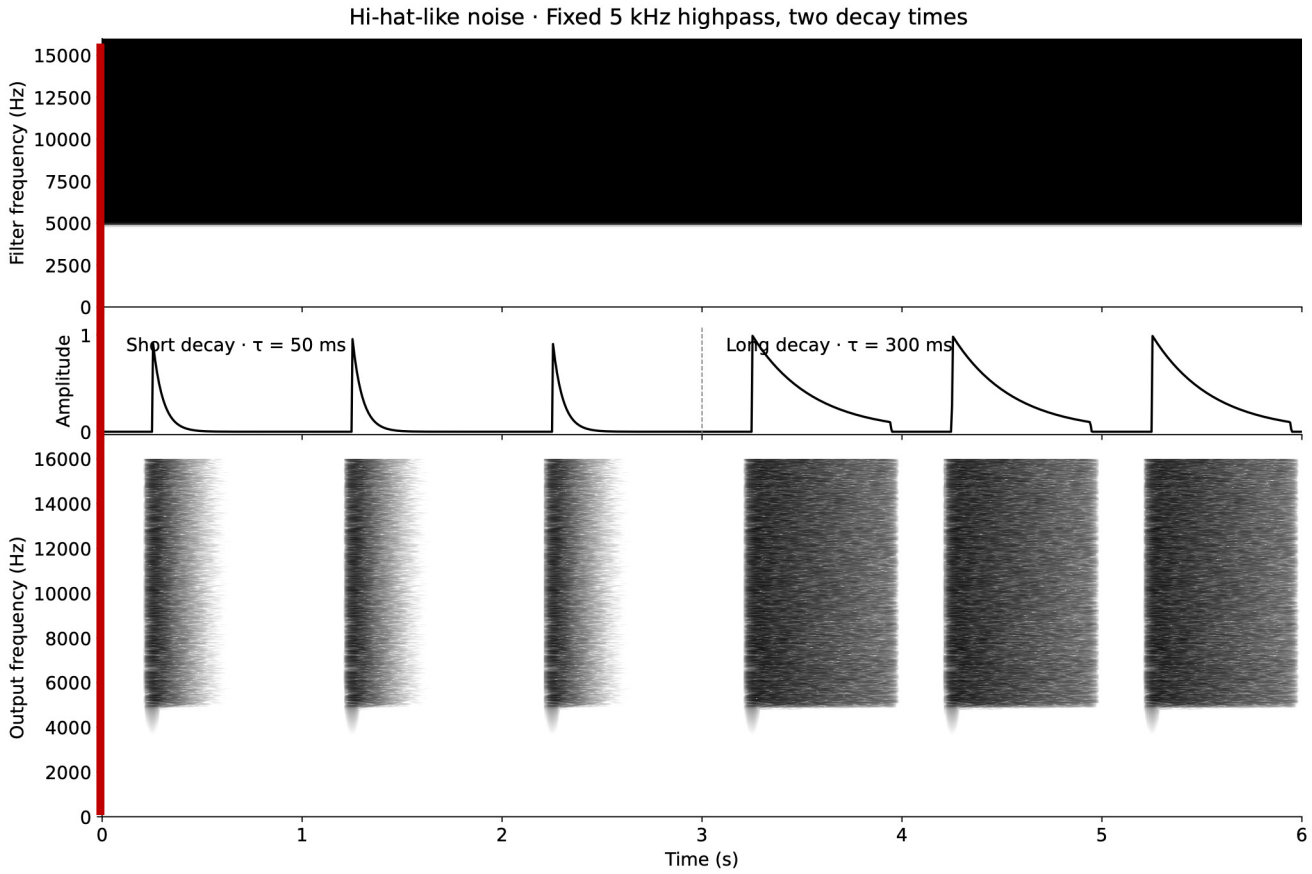
time-variant bandpass

time-invariant amplitude envelope

Differentiable Digital Signal Processing (DDSP)

Noise Model





time-invariant highpass

time-varying amplitude envelope

Differentiable Digital Signal Processing (DDSP)
Noise Model

What it can model

- Breath noise and other stochastic sound components.
- Changing noise color and amplitude envelopes.
- Noisy attacks and decays.

Limitations

- Models a spectral envelope rather than an exact noise waveform.
- Stochastic excitation does not provide precise phase control
- Spectral detail is limited by FIR filter length

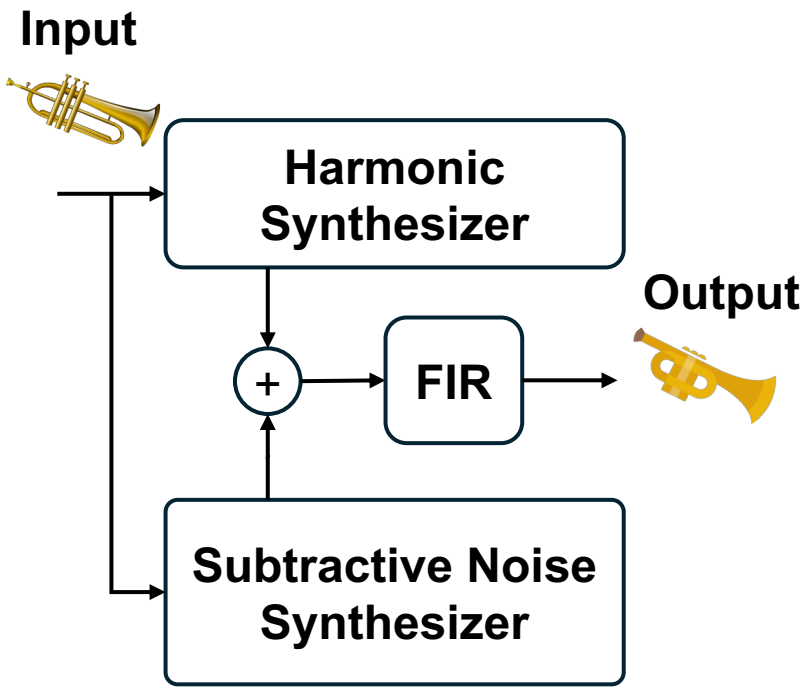
Differentiable Digital Signal Processing (DDSP)

What are the benefits?

- **Controllability:** Control signals can have a semantic meaning
- **Interpretability:** Used components are designed for a clear task (e.g., filtering)
- **Transfer:** We can link DSP knowledge with recent technology (ML)
- **Efficiency:** Less training data, smaller models, less compute

DDSP
Interpretability

J. Engel et al., “DDSP: Differentiable Digital Signal Processing,” in Proc. ICLR, 2020.
S. Schwär et al., “Differentiable Pulsetable Synthesis for Wind Instrument Modeling,” ICASSP, 2026



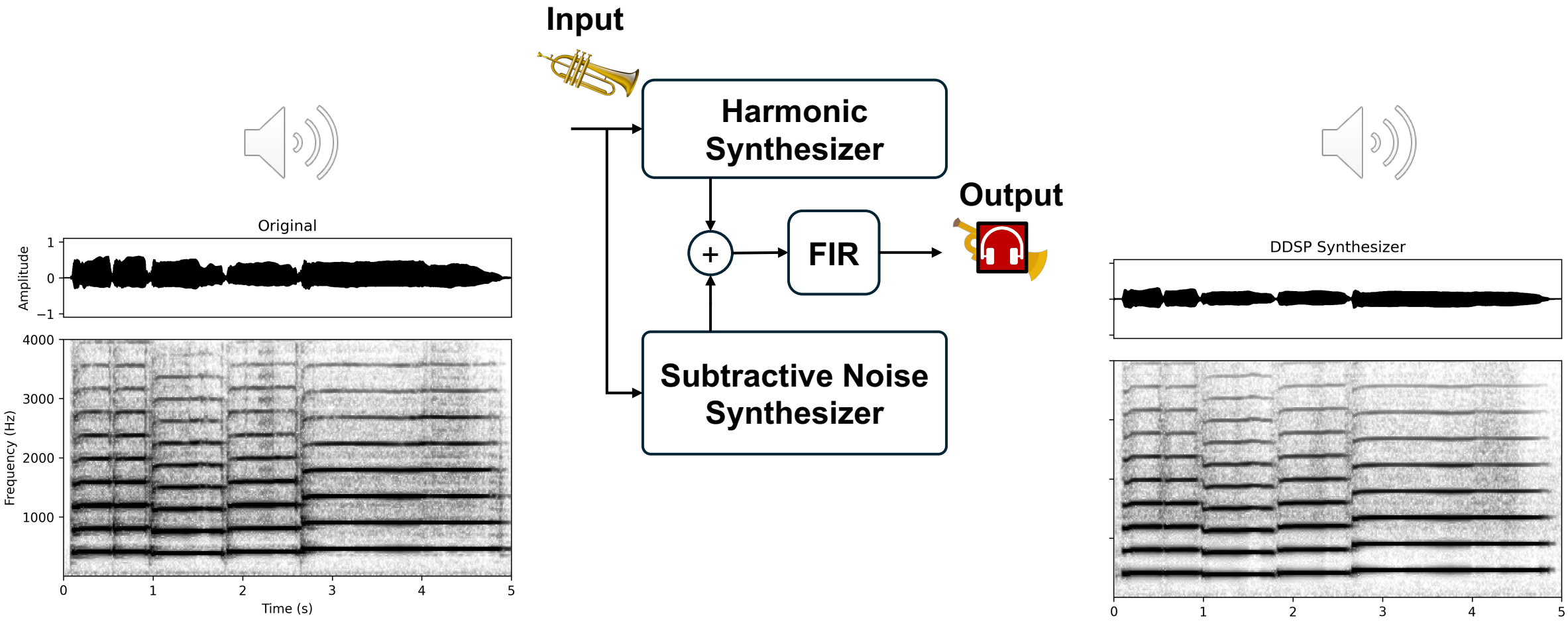
Our expectations:

- The harmonic synthesizer learns the tonal component
- Noise Synthesizer learns fricatives, onset deviations etc.
- FIR filter models the room acoustics

Experiment: Are these expectations met?

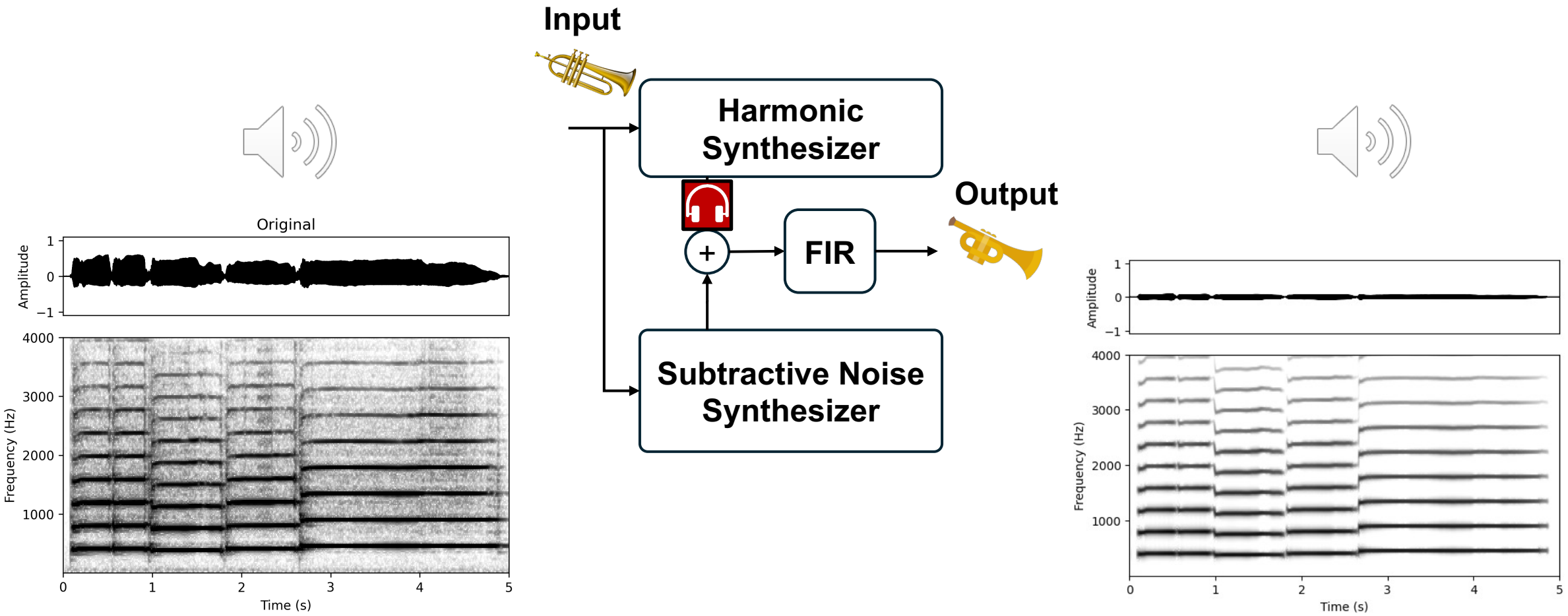
DDSP
Interpretability

Slides based on Manuel Peters' presentation in Barcelona, UPF Workshop, 2026.



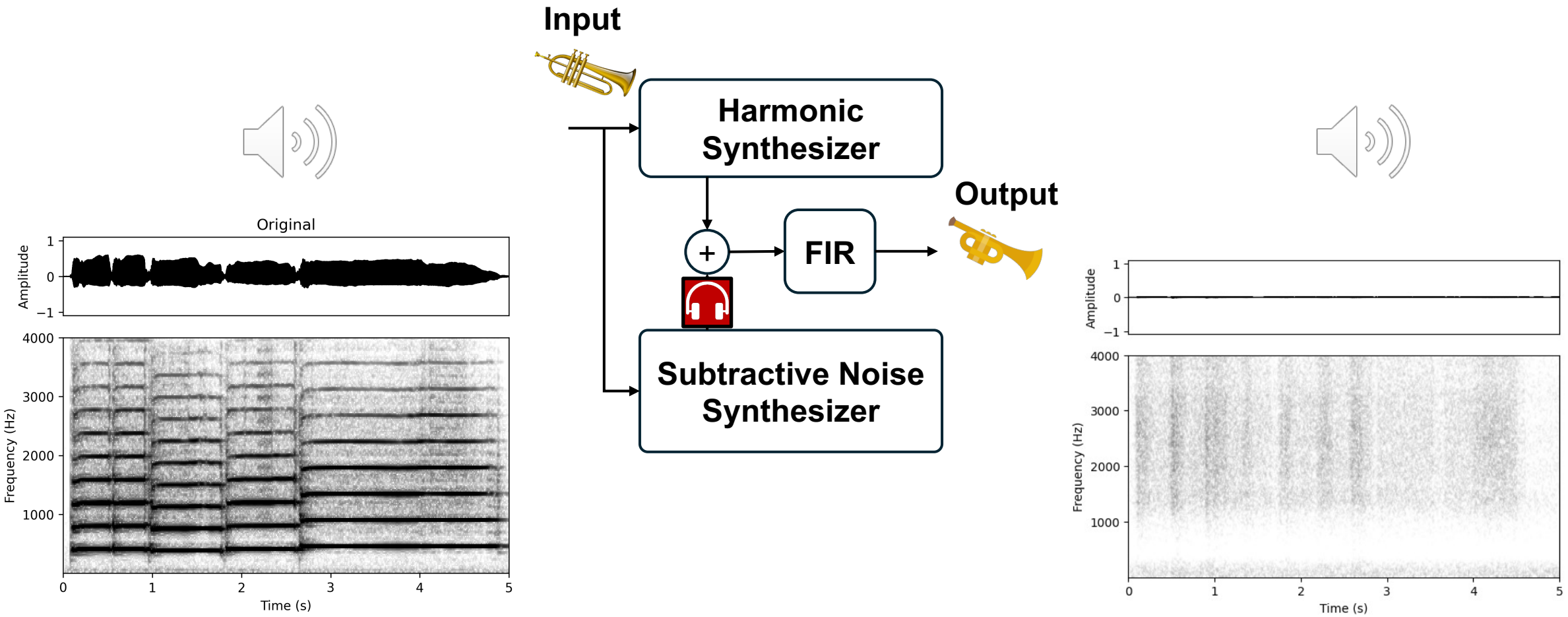
DDSP
Interpretability

Slides based on Manuel Peters' presentation in Barcelona, UPF Workshop, 2026.



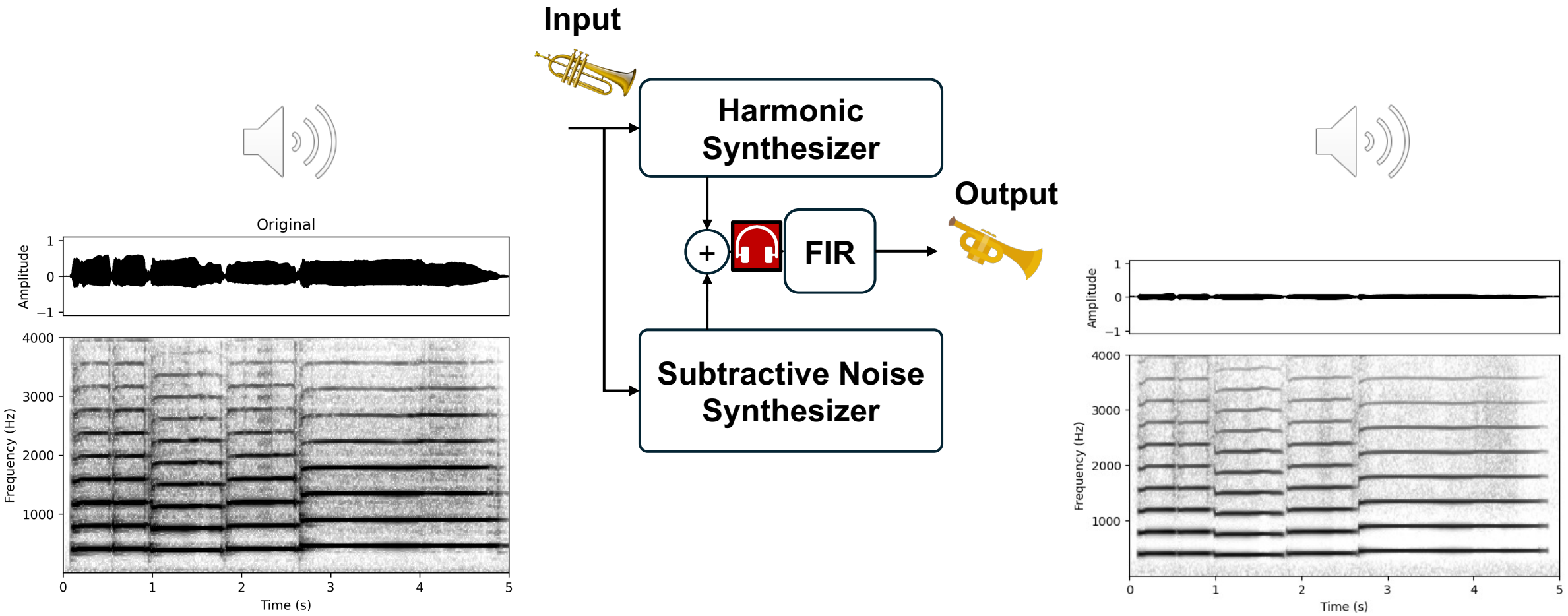
DDSP
Interpretability

Slides based on Manuel Peters' presentation in Barcelona, UPF Workshop, 2026.



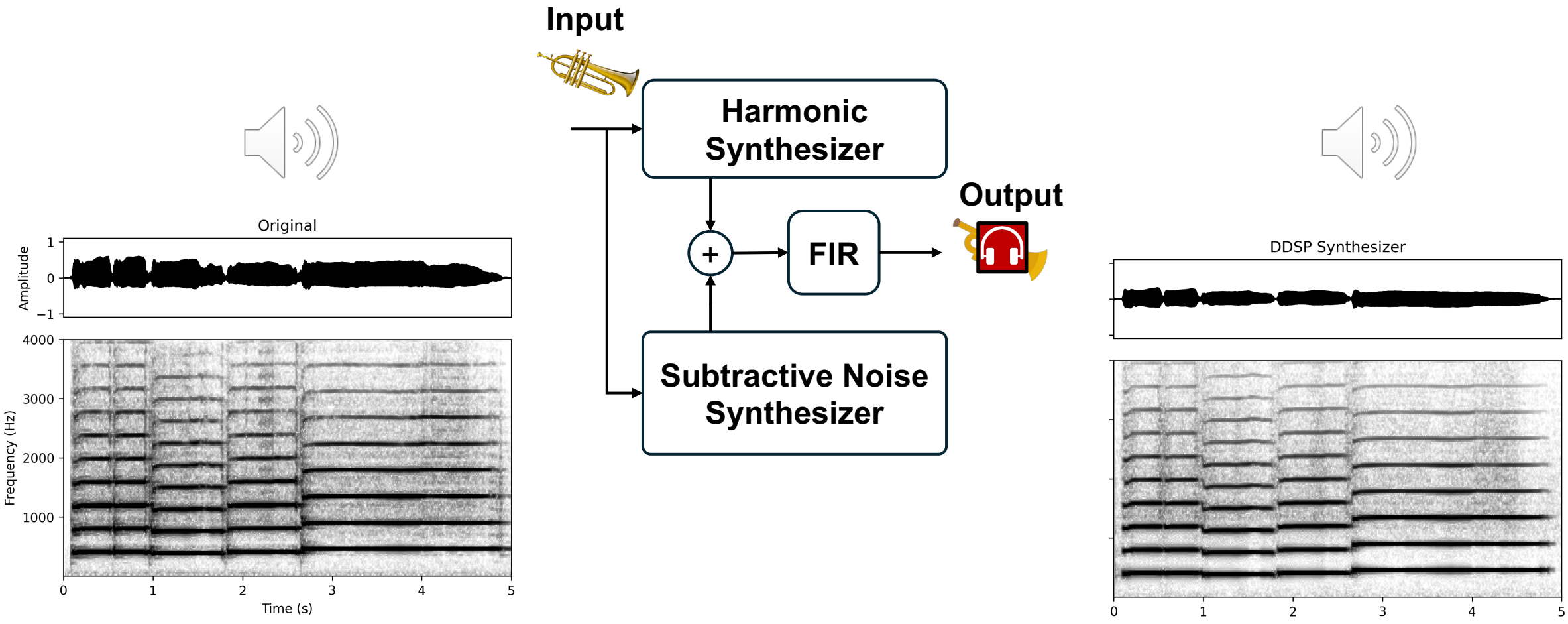
DDSP
Interpretability

Slides based on Manuel Peters' presentation in Barcelona, UPF Workshop, 2026.



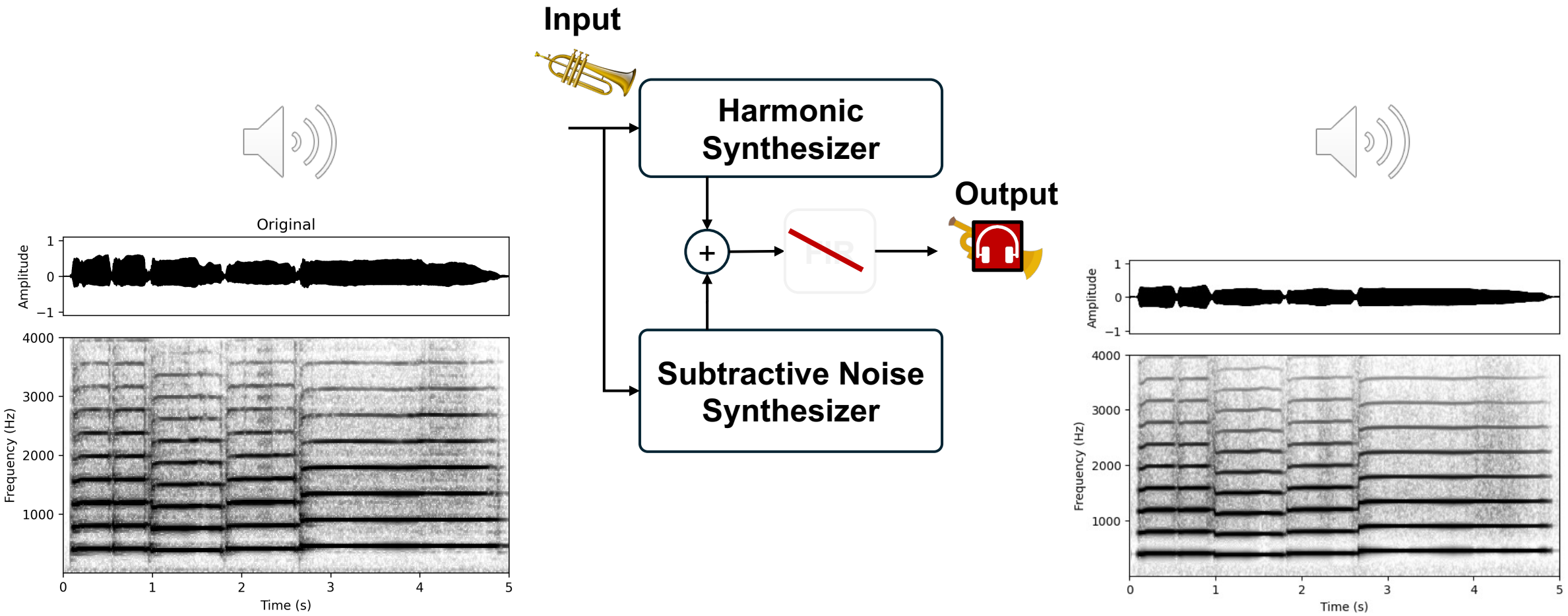
DDSP
Interpretability

Slides based on Manuel Peters' presentation in Barcelona, UPF Workshop, 2026.



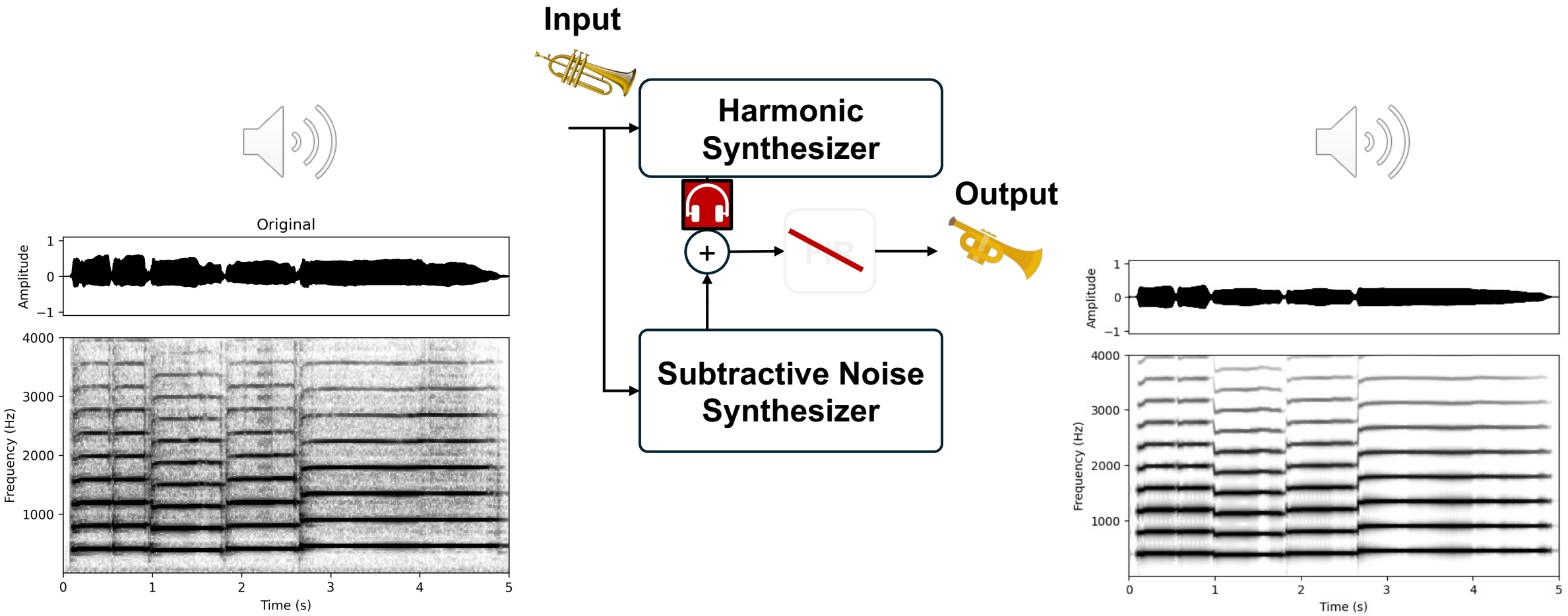
DDSP
Interpretability

Slides based on Manuel Peters' presentation in Barcelona, UPF Workshop, 2026.



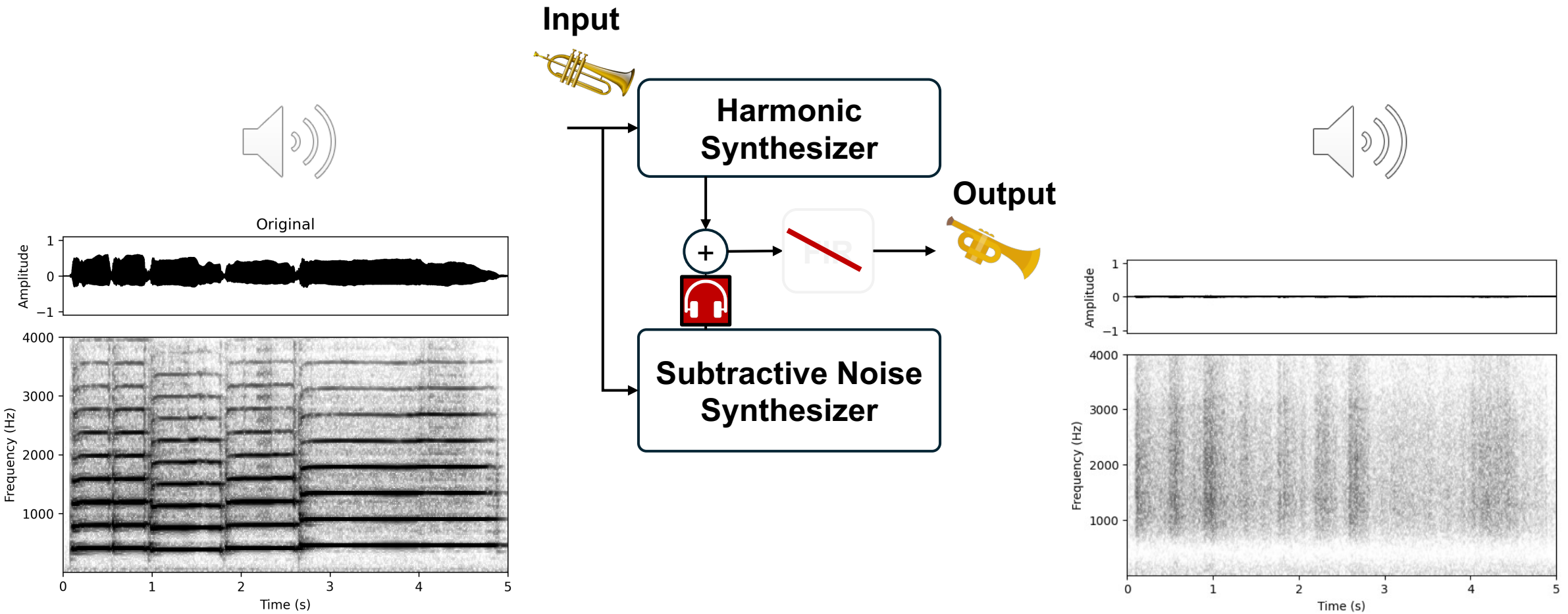
DDSP
Interpretability

Slides based on Manuel Peters' presentation in Barcelona, UPF Workshop, 2026.



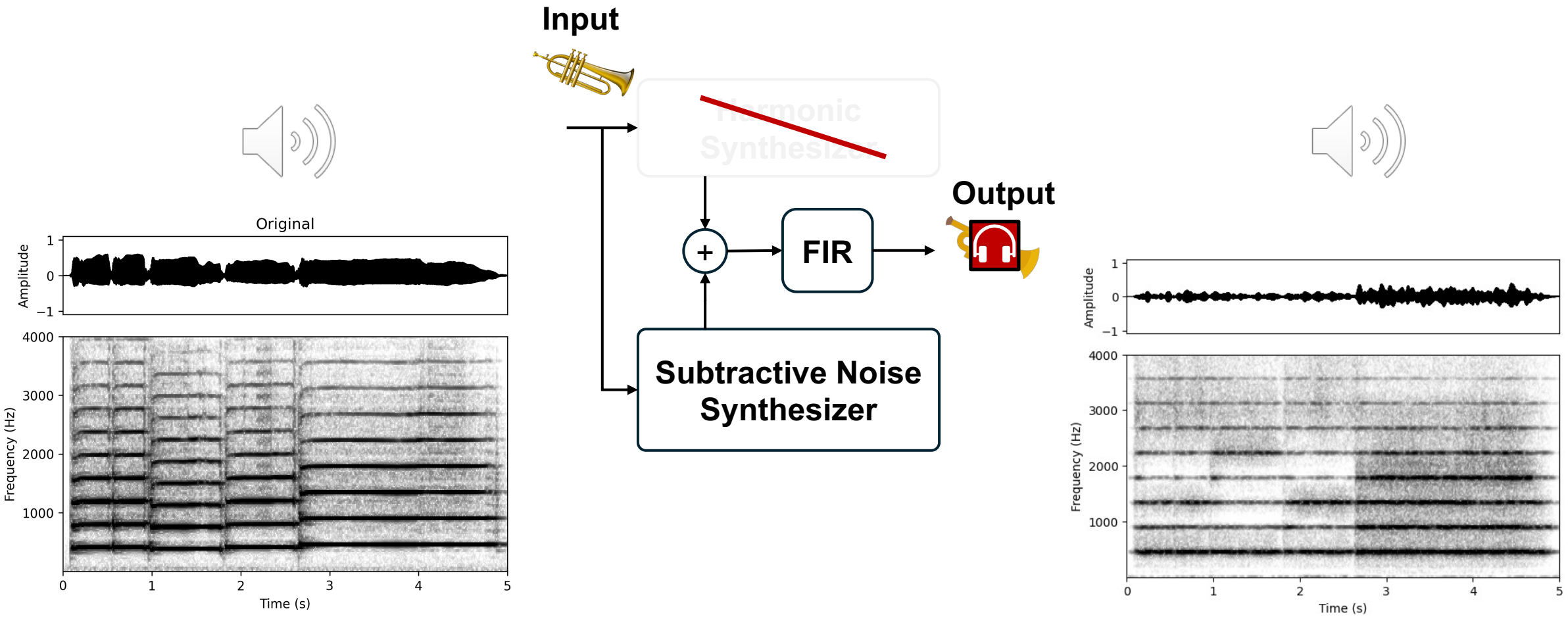
DDSP
Interpretability

Slides based on Manuel Peters' presentation in Barcelona, UPF Workshop, 2026.



DDSP
Interpretability

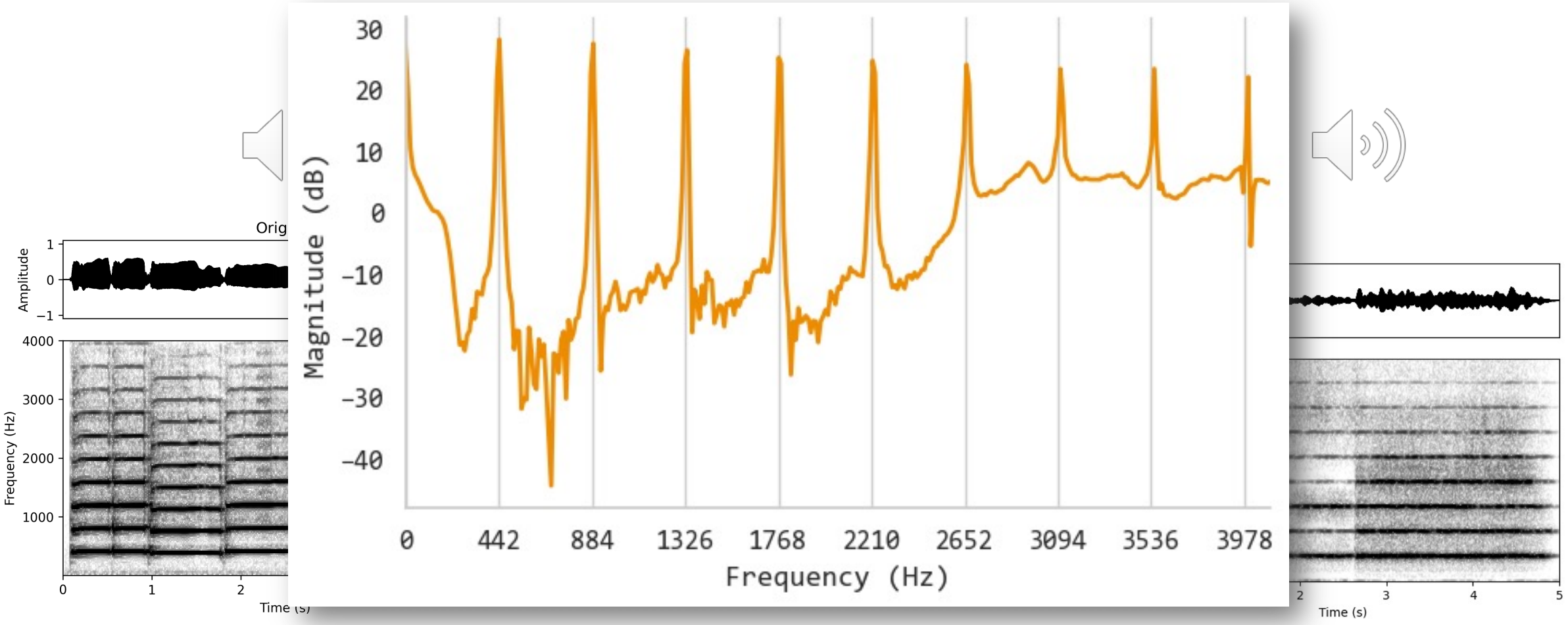
Slides based on Manuel Peters' presentation in Barcelona, UPF Workshop, 2026.



DDSP

Interpretability

Slides based on Manuel Peters' presentation in Barcelona, UPF Workshop, 2026.



DDSP
Interpretability

- DDSP Synthesizers
  - Are often musically motivated
  - But the design must be carefully translated into real DSP components

  → Otherwise, interpretability and controllability are lost

- A generic FIR filter is probably too powerful for learning reverb
  - How can we induce a stronger bias?
  - Investigate e.g. DDSP with Schroeder Reverberators, Feedback Delay Networks (FDNs), …

Differentiable Digital Signal Processing (DDSP)
What to take home?

- **Build on DSP knowledge**
  Known signal-processing structures provide useful assumptions about sound.
- **Learn through the signal path**
  Gradients optimize DSP parameters directly or train networks that predict them.
- **Create controllable sound**
  Synthesizers connect learned parameters to pitch, amplitude and spectral shape.
- **Check the interpretation**
  A good reconstruction does not guarantee that each component performs its intended role.

References (DDSP)

- Jesse Engel, Lamtharn Hantrakul, Chenjie Gu, Adam Roberts: DDSP: Differentiable Digital Signal Processing. Proceedings of ICLR. 2020. https://arxiv.org/pdf/2001.04643
- Ben Hayes, Jordie Shier, György Fazekas, Andrew McPherson and Charalampos Saitis: A review of differentiable digital signal processing for music and speech synthesis. Frontiers in Signal Processing. 2024. https://doi.org/10.3389/frsip.2023.1284100
- Xavier Serra and Julius Smith, III: Spectral Modeling Synthesis: A Sound Analysis/Synthesis System Based on a Deterministic Plus Stochastic Decomposition. Computer Music Journal. 1990. https://doi.org/10.2307/3680788
- Bernard Widrow, Marcian E. Hoff: Adaptive Switching Circuits. IRE WESCON Convention Record. 1960. https://isl.stanford.edu/~widrow/papers/c1960adaptiveswitching.pdf
- Simon Schwär, Christian Dittmar, Stefan Balke, and Meinard Müller: Differentiable Pulsetable Synthesis for Wind Instrument Modeling. Proceedings of ICASSP. 2026. https://dx.doi.org/10.1109/ICASSP55912.2026.11462505
- Ben Hayes, Jordie Shier, Chin-Yun Yu, David Südholt, and Rodrigo Diaz: Introduction to DDSP for Audio Synthesis. ISMIR tutorial notes. 2023 https://intro2ddsp.github.io/