ISMIR

ISMIR Tutorial
Abu Dhabi, UAE, November 08, 2026

ISMIR ABU DHABI 2026

Introduction to Music Information Retrieval: Three Perspectives

Part 1: A Signal Processing Perspective

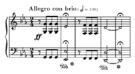
Meinard Müller
International Audio Laboratories Erlangen
Friedrich-Alexander Universität Erlangen-Nürnberg
meinard.mueller@audiolabs-erlangen.de

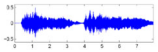
Christof Weiß
Center for Artificial Intelligence and Data Science
Julius-Maximilians-Universität Würzburg
christof.weiss@uni-wuerzburg.de

AUDIO LABS | FAU | Fraunhofer IIS | Universität Würzburg | CAIDAS | zpd

Music

Music Information Retrieval (MIR)

Sheet Music (Image)

CD / MP3 (Audio)



MusicXML (Text)

Dance / Motion (Mocap)

Music

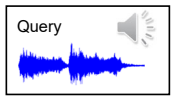
MIDI

Singing / Voice (Audio)

Music Film (Video)

Music Literature (Text)

Music Retrieval



Audio ID

Bernstein (1962)
Beethoven, Symphony No. 5

Music Retrieval

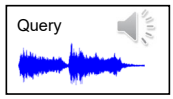
Audio ID

Bernstein (1962)
Beethoven, Symphony No. 5

Version ID

Beethoven, Symphony No. 5:
- Bernstein (1962)
- Karajan (1982)
- Gould (1992)

Music Retrieval



Audio ID

Bernstein (1962)
Beethoven, Symphony No. 5

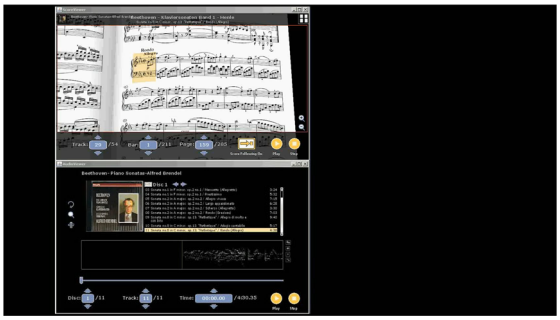
Version ID

Beethoven, Symphony No. 5:
- Bernstein (1962)
- Karajan (1982)
- Gould (1992)

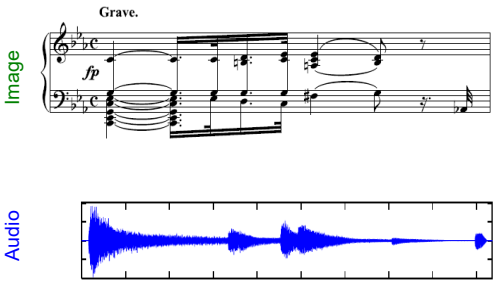
Category ID

- Beethoven, Symphony No. 9
- Beethoven, Symphony No. 3
- Haydn Symphony No. 94

Music Synchronization

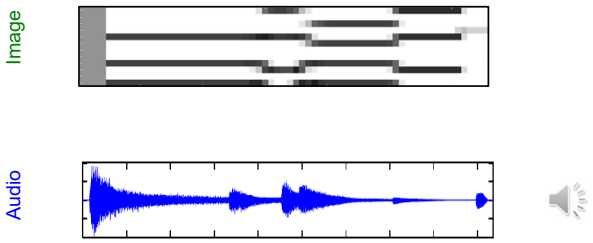


Music Synchronization: Image-Audio



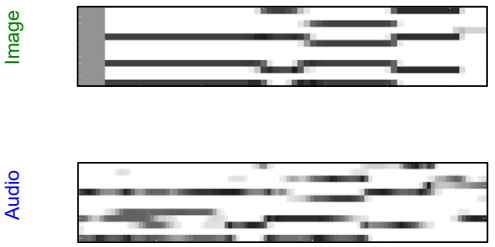
Music Synchronization: Image-Audio

Image Processing: Optical Music Recognition



Music Synchronization: Image-Audio

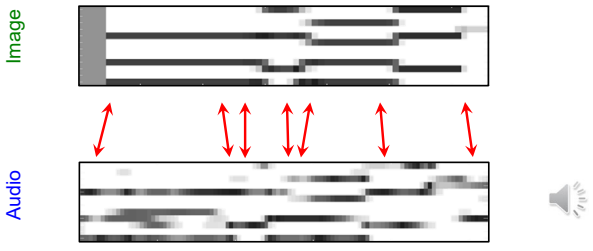
Image Processing: Optical Music Recognition



Audio Processing: Fourier Analysis

Music Synchronization: Image-Audio

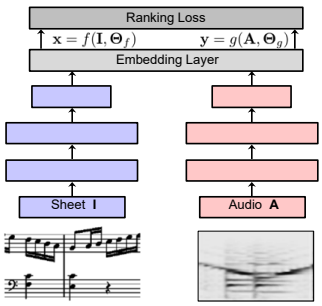
Image Processing: Optical Music Recognition



Audio Processing: Fourier Analysis

Music Synchronization: Image-Audio

Müller, Arzt, Balke, Dorfer, Widmer: **Cross-Modal Music Retrieval and Applications.** IEEE Signal Process. Mag. 36(1): 52–62, 2019



- Deep learning
- Embedding techniques
- Weak annotations
- Loss functions
- …

Part 1: A Signal Processing Perspective
Key Topics

- Soft Dynamic Time Warping
- Piano Concerto Source Separation
- Wind Instrument Synthesis
- Multi-Scale Spectral Loss
- Hierarchical Classification Loss

Part 1: A Signal Processing Perspective
Key Topics

- **Soft Dynamic Time Warping**
- Piano Concerto Source Separation
- Wind Instrument Synthesis
- Multi-Scale Spectral Loss
- Hierarchical Classification Loss

Special Thanks

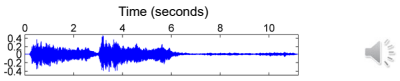
Johannes Zeitler

Literature

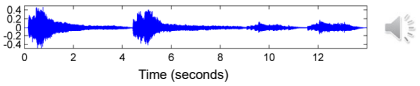
- Cuturi, Blondel: Soft-DTW: A Differentiable Loss Function for Time-Series. Proc. ICML, 2017.
- Blondel, Mensch, Vert: Differentiable Divergences Between Time Series. AISTATS, 2021.
- Krause, Weiß, Müller: Soft Dynamic Time Warping For Multi Pitch Estimation And Beyond. Proc. ICASSP, 2023.
- Zeitler, Müller: A Unified Perspective on CTC and Soft-DTW Using Differentiable DTW. IEEE TASLPRO, 2026.
- Zeitler: Differentiable Alignment Techniques for Music Processing. PhD Thesis, FAU, 2006.

Motivation: Audio-Audio Alignment
Beethoven's Fifth

Karajan (Orchester)

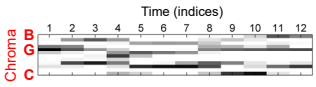


Gould (Piano)



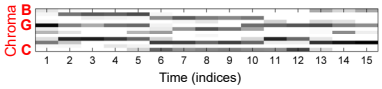
Motivation: Audio-Audio Alignment
Beethoven's Fifth

Karajan (Orchester)



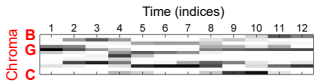
Time–chroma representations

Gould (Piano)



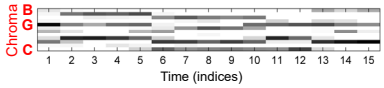
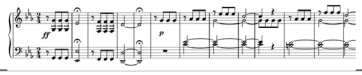
Motivation: Audio-Audio Alignment
Beethoven's Fifth

Karajan (Orchester)



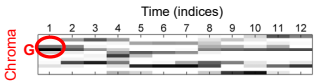
Time–chroma representations

Gould (Piano)



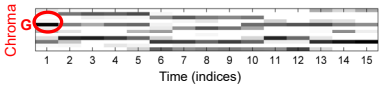
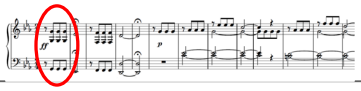
Motivation: Audio-Audio Alignment
Beethoven's Fifth

Karajan (Orchester)



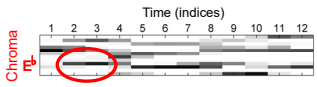
Time–chroma representations

Gould (Piano)



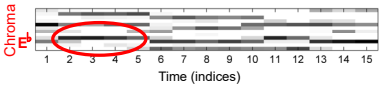
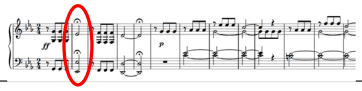
Motivation: Audio-Audio Alignment
Beethoven's Fifth

Karajan (Orchester)

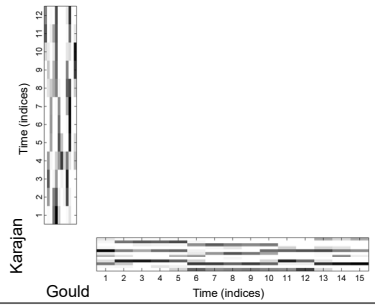


Time–chroma representations

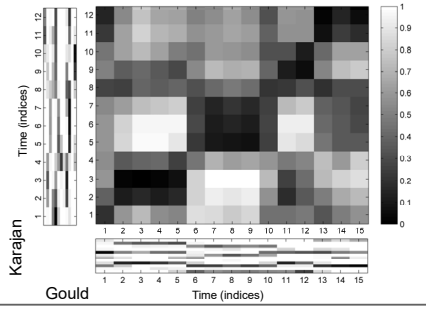
Gould (Piano)



Motivation: Audio-Audio Alignment
Beethoven's Fifth

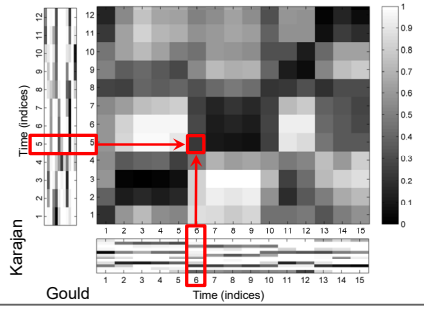


Motivation: Audio-Audio Alignment
Beethoven's Fifth



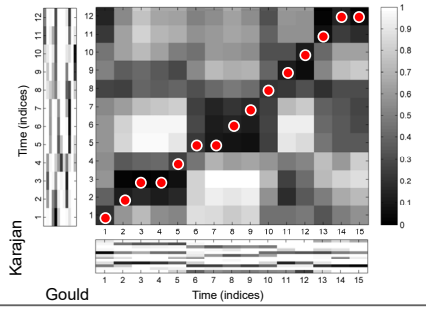
Cost matrix

Motivation: Audio-Audio Alignment
Beethoven's Fifth



Cost matrix

Motivation: Audio-Audio Alignment
Beethoven's Fifth

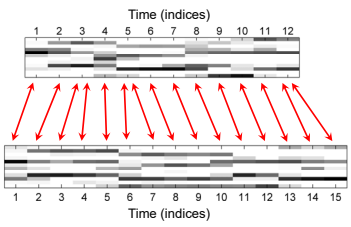


Cost-minimizing warping path

Motivation: Audio-Audio Alignment
Beethoven's Fifth

Karajan (Orchester)

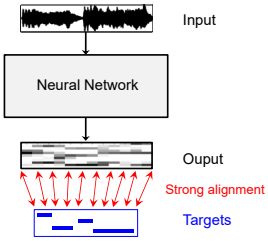
Gould (Piano)



Cost-minimizing warping path

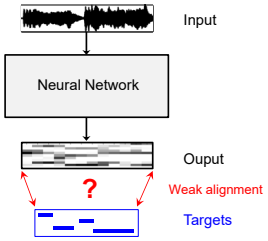
→ Strong alignment

Feature Learning



- Task: Learn audio features using a neural network
- Loss: Binary cross-entropy
  - framewise loss
  - requires strongly aligned targets
  - hard to obtain

Feature Learning



- Task: Learn audio features using a neural network
- Loss: Binary cross-entropy
  - framewise loss
  - requires strongly aligned targets
  - hard to obtain
- Alignment as part of loss function
  - requires only weakly aligned targets
  - needs to be differentiable
- Problem: DTW is not differentiable
  → Soft DTW

Dynamic Time Warping (DTW)

$X := (x_1, x_2, \ldots, x_N)$

$Y := (y_1, y_2, \ldots, y_M)$

$x_n, y_m \in \mathcal{F},\ n \in [1:N],\ m \in [1:M]$

$\mathcal{F}$ = Feature space

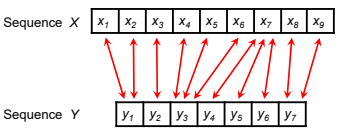
Alignment matrix

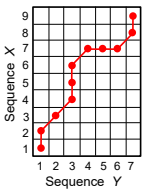
$A \in \{0,1\}^{N \times M}$

Set of all possible alignment matrices

$\mathcal{A}_{N,M} \subset \{0,1\}^{N \times M}$

Alignment





Dynamic Time Warping (DTW)

$X := (x_1, x_2, \ldots, x_N)$

$Y := (y_1, y_2, \ldots, y_M)$

$x_n, y_m \in \mathcal{F},\ n \in [1:N],\ m \in [1:M]$

$\mathcal{F}$ = Feature space

Alignment matrix

$A \in \{0,1\}^{N \times M}$

Set of all possible alignment matrices

$\mathcal{A}_{N,M} \subset \{0,1\}^{N \times M}$

Cost measure: $c : \mathcal{F} \times \mathcal{F} \to \mathbb{R}_{\geq 0}$

Cost matrix: $C \in \mathbb{R}^{N \times M}$ with $C(n,m) := c(x_n, y_m)$

Cost of alignment: $\langle A, C \rangle$

DTW cost: $\mathrm{DTW}(C) = \min\left(\{\langle A, C\rangle \mid A \in \mathcal{A}_{N,M}\}\right)$

Optimal alignment: $A^* = \operatorname{argmin}\left(\{\langle A, C\rangle \mid A \in \mathcal{A}_{N,M}\}\right)$

Dynamic Time Warping (DTW)

DTW cost: $\mathrm{DTW}(C) = \min\left(\{\langle A, C\rangle \mid A \in \mathcal{A}_{N,M}\}\right)$

- Efficient computation via Bellman's recursion in O(*NM*)

$D(n,m) = \min\{D(n-1,m), D(n,m-1), D(n,m)\} + C(n,m)$

for *n>1* and *m>1* and suitable initialization.

$\mathrm{DTW}(C) = D(N,M)$

- Problem: DTW(C) is not differentiable with regard to *C*
- Idea: Replace min-function by a smooth version

$$\min^{\gamma}(\mathcal{S}) = -\gamma \log \sum\nolimits_{s \in \mathcal{S}} \exp(-s/\gamma)$$

for set $\mathcal{S} \subset \mathbb{R}$ and temperature parameter $\gamma \in \mathbb{R}$

Soft Dynamic Time Warping (SDTW)

SDTW cost: $\mathrm{SDTW}^{\gamma}(C) = \min^{\gamma}\left(\{\langle A, C\rangle \mid A \in \mathcal{A}_{N,M}\}\right)$

- Efficient computation via Bellman's recursion in O(*NM*) still works:

$D^{\gamma}(n,m) = \min^{\gamma}\{D^{\gamma}(n-1,m), D^{\gamma}(n,m-1), D^{\gamma}(n,m)\} + C(n,m)$

for *n>1* and *m>1* and suitable initialization.

$\mathrm{SDTW}^{\gamma}(C) = D^{\gamma}(N,M)$

- Limit case: $\mathrm{SDTW}^{\gamma}(C) \xrightarrow{\gamma \to 0} \mathrm{DTW}(C)$
- SDTW(C) is differentiable with regard to *C*
- Questions:
  - How does the gradient look like?
  - Can it be computed efficiently?
  - How does SDTW generalize the alignment concept?

Soft Dynamic Time Warping (SDTW)

Soft-DTW
Cuturi, Blondel: Soft-DTW: A Differentiable Loss Function for Time-Series. ICML, 2017

SDTW cost: $\mathrm{SDTW}^{\gamma}(C) = \min^{\gamma}(\{\langle A, C\rangle \mid A \in \mathcal{A}_{N,M}\})$

- Define $p^{\gamma}(C)$ as the following "probability" distribution over $\mathcal{A}_{N,M}$:

$$p^{\gamma}(C)_A = \frac{\exp(-\langle A, C\rangle/\gamma)}{\sum_{A' \in \mathcal{A}_{N,M}} \exp(-\langle A', C\rangle/\gamma)} \quad \text{for } A \in \mathcal{A}_{N,M}$$

- The expected alignment with respect to $p^{\gamma}(C)$ is given by:

$$E^{\gamma}(C) = \sum_{A \in \mathcal{A}_{N,M}} p^{\gamma}(C)_A A \quad \in \mathbb{R}^{N \times M}$$

- The gradient is given by:

$$\nabla_C \mathrm{SDTW}^{\gamma}(C) = E^{\gamma}(C)$$

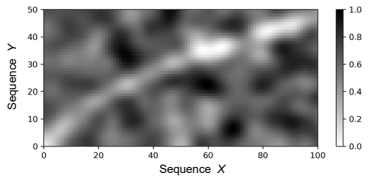
- The gradient can be computed efficiently in O(*NM*) via a recursive algorithm.

Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^{\gamma}(C) = \sum_{A \in \mathcal{A}_{N,M}} p^{\gamma}(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter γ

Cost matrix C

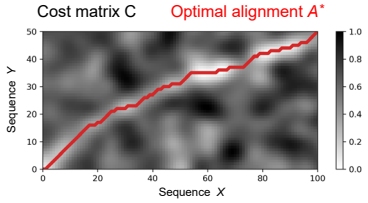


Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^{\gamma}(C) = \sum_{A \in \mathcal{A}_{N,M}} p^{\gamma}(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter γ

Cost matrix C Optimal alignment A^*

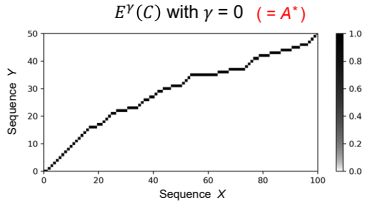


Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^{\gamma}(C) = \sum_{A \in \mathcal{A}_{N,M}} p^{\gamma}(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter γ

$E^{\gamma}(C)$ with $\gamma = 0$ ($= A^*$)

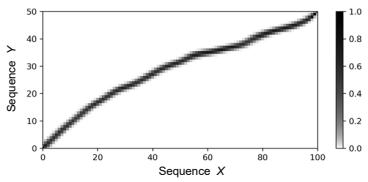


Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^{\gamma}(C) = \sum_{A \in \mathcal{A}_{N,M}} p^{\gamma}(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter γ

$E^{\gamma}(C)$ with $\gamma = 0.1$

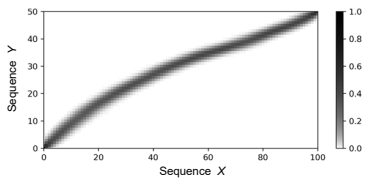


Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^{\gamma}(C) = \sum_{A \in \mathcal{A}_{N,M}} p^{\gamma}(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter γ

$E^{\gamma}(C)$ with $\gamma = 1$

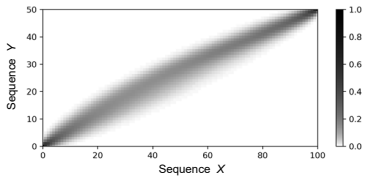


Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^{\gamma}(C) = \sum_{A \in \mathcal{A}_{N,M}} p^{\gamma}(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter γ

$E^{\gamma}(C)$ with γ = 10

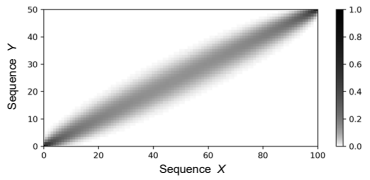


Soft Dynamic Time Warping (SDTW)

Expected alignment : $E^{\gamma}(C) = \sum_{A \in \mathcal{A}_{N,M}} p^{\gamma}(C)_A A \quad \in \mathbb{R}^{N \times M}$

- Can be interpreted as a smoothed version of an alignment
- Degree of smoothing depends on temperature parameter γ

$E^{\gamma}(C)$ with γ = 100



Implementation of Differentiable Alignment Losses

dDTW

- Covers many soft alignment methods
  - Soft Dynamic Time Warping (SDTW) loss
  - Connectionist Temporal Classification (CTC) loss
  - Graph-based generalizations
- Unified framework
  - Fast forward algorithm
  - Fast gradient computation

Zeitler, Müller: **dDTW: A Unified and Efficient Toolbox for Differentiable Sequence Alignment.** Submitted, 2026.

GitHub https://github.com/groupmm/dDTW

pypi https://pypi.org/project/ddtw/

Part 1: A Signal Processing Perspective
Key Topics

- Soft Dynamic Time Warping
- **Piano Concerto Source Separation**
- Wind Instrument Synthesis
- Multi-Scale Spectral Loss
- Hierarchical Classification Loss

Special Thanks

Yigitcan Özer

Literature

- Cano, FitzGerald, Liutkus, Plumbley, Stöter: Musical Source Separation: An Introduction. IEEE SPM, 2019.
- Stöter, Uhlich, Luitkus, Mitsufuji: Open-Unmix–A Reference Implementation for Music Source Separation. JOSS, 2019.
- Özer, Schwär, Arifi-Müller, Lawrence, Sen, Müller: Piano Concerto Dataset (PCD): A Multitrack Dataset of Piano Concertos. TISMIR, 2023.
- Özer, Müller: Source Separation of Piano Concertos Using Musically Motivated Augmentation Techniques. IEEE TASLP, 2024.
- Özer: Source Separation of Piano Music Recordings. PhD Thesis, FAU, 2024.

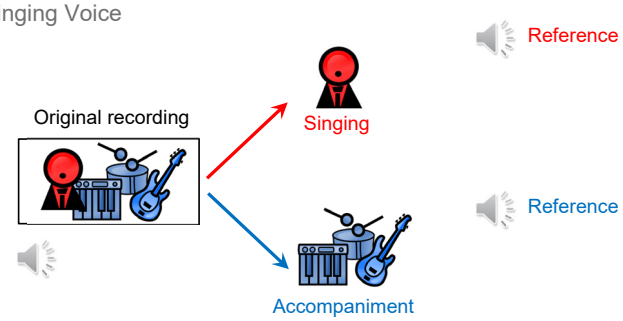
Source Separation

- Decomposition of audio stream into different sound sources
- Central task in digital signal processing
- “Cocktail party effect”

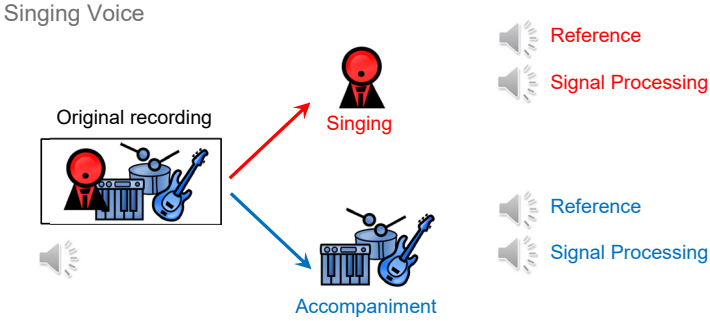
Source Separation
Singing Voice

Original recording

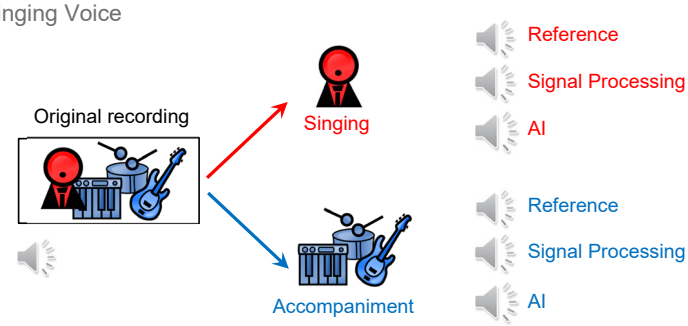
Source Separation
Singing Voice



Source Separation
Singing Voice

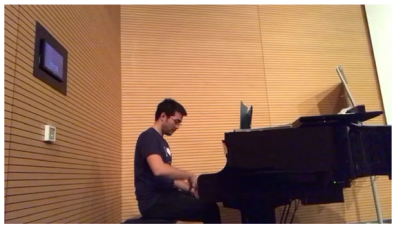


Source Separation
Singing Voice



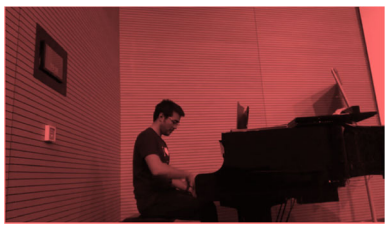
Source Separation
Piano Concerto

- Yigitcan Özer
- PhD student in engineering
- Pianist



Source Separation
Piano Concerto

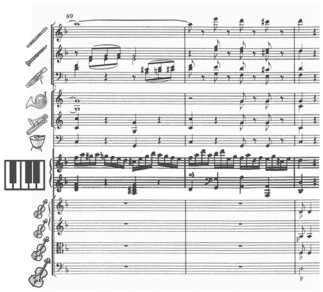
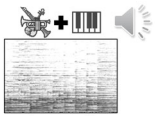
- Yigitcan Özer
- PhD student in engineering
- Pianist



Only Piano!

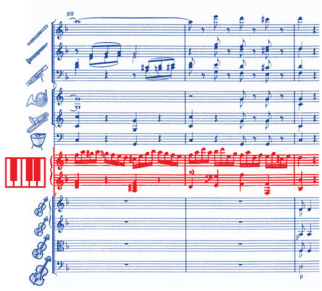
Where is the orchestra?

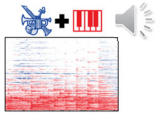
Source Separation
Piano Concerto

Özer, Müller: **Source Separation of Piano Concertos Using Musically Motivated Augmentation Techniques.** IEEE/ACM Trans. ASLP, 32: 1214–1225, 2024



Source Separation
Piano Concerto

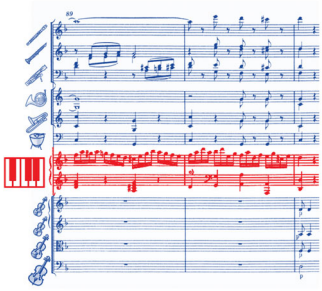
Özer, Müller: **Source Separation of Piano Concertos Using Musically Motivated Augmentation Techniques.** IEEE/ACM Trans. ASLP, 32: 1214–1225, 2024

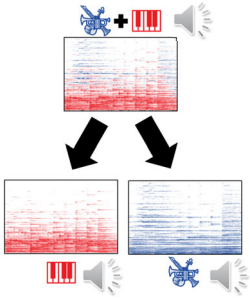




Source Separation
Piano Concerto

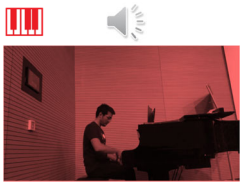
Özer, Müller: **Source Separation of Piano Concertos Using Musically Motivated Augmentation Techniques.** IEEE/ACM Trans. ASLP, 32: 1214–1225, 2024

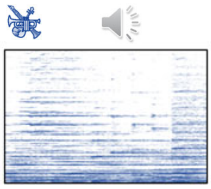




Source Separation
Piano Concerto

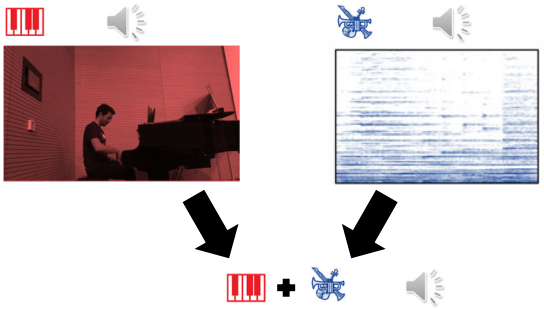
Özer, Müller: **Source Separation of Piano Concertos Using Musically Motivated Augmentation Techniques.** IEEE/ACM Trans. ASLP, 32: 1214–1225, 2024





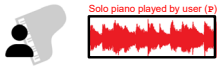
Source Separation
Piano Concerto

Özer, Müller: **Source Separation of Piano Concertos Using Musically Motivated Augmentation Techniques.** IEEE/ACM Trans. ASLP, 32: 1214–1225, 2024



Accompaniment Creation

- Lonely pianist plays solo piano part

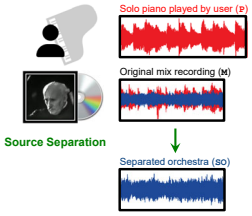


Accompaniment Creation

- Lonely pianist plays solo piano part
- Select full mix recording of concerto

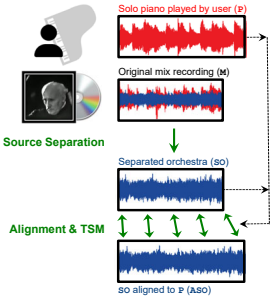
Accompaniment Creation

- Lonely pianist plays solo piano part
- Select full mix recording of concerto
- Isolate orchestra track



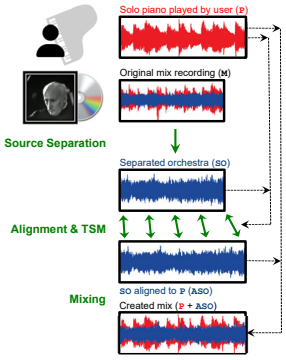
Accompaniment Creation

- Lonely pianist plays solo piano part
- Select full mix recording of concerto
- Isolate orchestra track
- Align to the solo performance



Accompaniment Creation

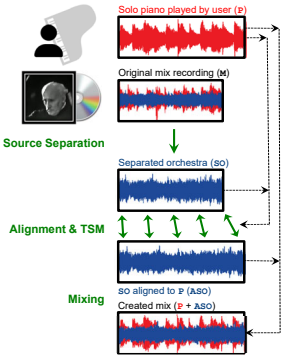
- Lonely pianist plays solo piano part
- Select full mix recording of concerto
- Isolate orchestra track
- Align to the solo performance
- Create own coherent mix



Accompaniment Creation

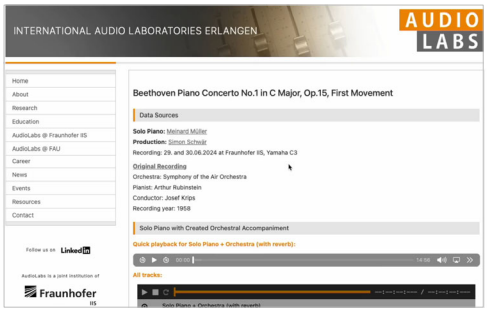
- Lonely pianist plays solo piano part
- Select full mix recording of concerto
- Isolate orchestra track
- Align to the solo performance
- Create own coherent mix

Make lonely pianist happy with "Berliner Philharmoniker"



Accompaniment Creation

Özer, Schwär, Müller: **Piano Concerto Accompaniment Creation.** LBD ISMIR, 2024



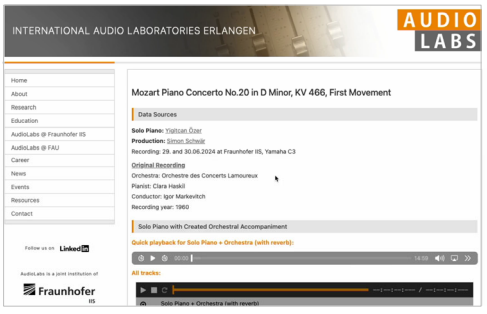
Beethoven: Piano Concerto No. 1

Lonely pianist: Meinard Müller

Recording (1958): Symphony of the Air Orchestra

Accompaniment Creation

Özer, Schwär, Müller: **Piano Concerto Accompaniment Creation.** LBD ISMIR, 2024



Mozart: Piano Concerto No. 20

Lonely pianist: Yiğitcan Özer

Recording (1960): Orchestre des Concerts Lamoureux

Part 1: A Signal Processing Perspective
Key Topics

- Soft Dynamic Time Warping
- Piano Concerto Source Separation
- **Wind Instrument Synthesis**
- Multi-Scale Spectral Loss
- Hierarchical Classification Loss

Special Thanks

Christian Dittmar

Special Thanks

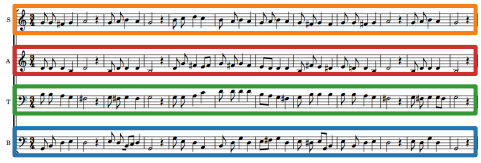
Stefan Balke

Literature

- Engel, Hantrakul, Gu, Roberts: DDSP: Differentiable digital signal processin. Proc. ICLR, 2020
- Hayes et al.: A review of differentiable digital signal processing for music and speech synthesis," Frontiers in Signal Processing, 2024
- Balke, Berndt, Müller: ChoraleBricks: A Modular Multitrack Dataset for Wind Music Research. TISMIR, 2025.
- Schwär, Dittmar, Balke, Müller: Differentiable Pulsetable Synthesis for Wind Instrument Sounds. Proc. ICASSP, 2026.
- Balke, Müller. Advancing Music Processing for Concert Band and Wind Music. DFG-Funded Project, 2025–2028

Concert Band

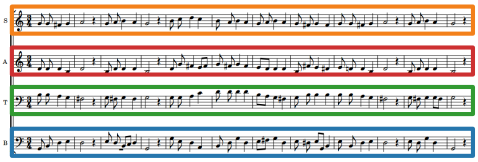
Concert Band

Four-Voice Choir (SATB)
Drese: Jesu geh voran

Soprano (S)

Alto (A)

Tenor (T)

Bass (B)

Four-Voice Choir (SATB)
Drese: Jesu geh voran

Soprano (S)

Alto (A)

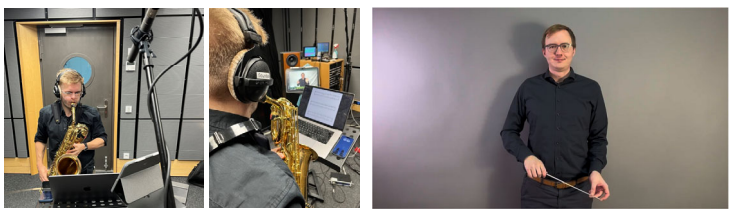
Tenor (T)

Bass (B)

ChoraleBricks
Multitrack Dataset for Wind Music Research

Balke, Berndt, Müller: **ChoraleBricks: A Modular Multitrack Dataset for Wind Music Research**. TISMIR, 8(1): 39–54, 2025

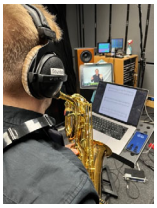
- Dataset featuring 10 SATB chorales
- 193 isolated tracks with 13 instruments
- Recorded along conducting video for syncronization



ChoraleBricks
Multitrack Dataset for Wind Music Research

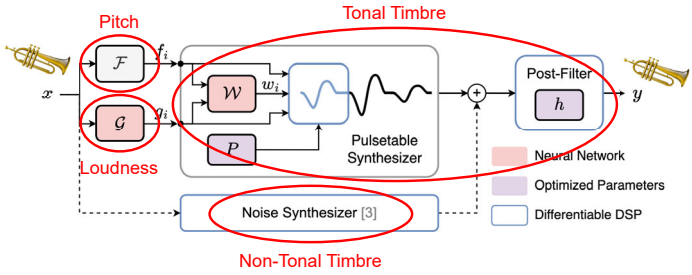
Balke, Berndt, Müller: **ChoraleBricks: A Modular Multitrack Dataset for Wind Music Research**. TISMIR, 8(1): 39–54, 2025

- Dataset featuring 10 SATB chorales
- 193 isolated tracks with 13 instruments
- Recorded along conducting video for syncronization

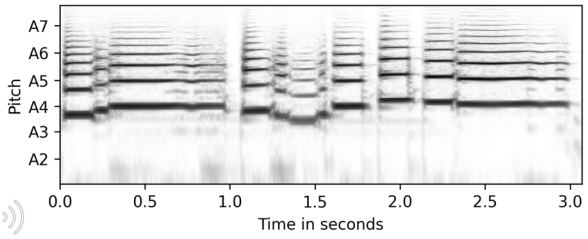


Differentiable Digital Signal Processing
Pulsetable Synthesis

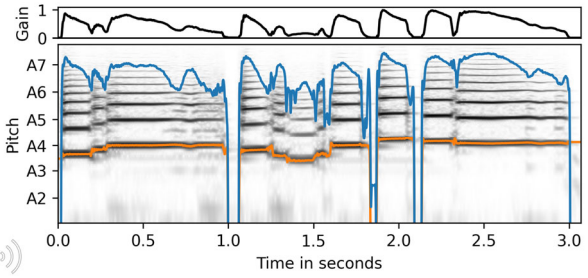
Dittmar, Zeitler, Balke, Schwär, Müller: **PULSE-IT: Lightweight and Expressive Synthesis of Wind Instrument Playing**. LBD ISMIR, 2025



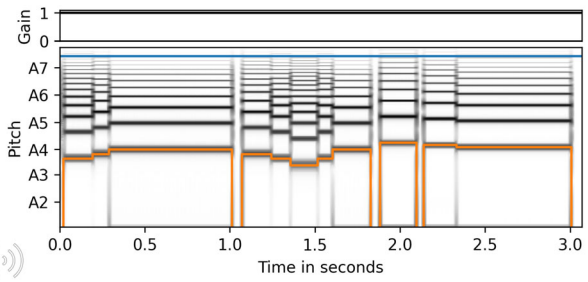
Wind Instrument Synthesis
Input Signal



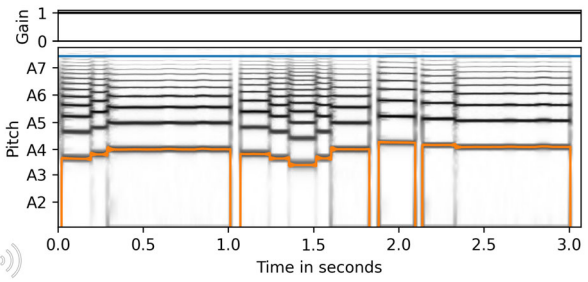
Wind Instrument Synthesis
Extract Control Signals



Wind Instrument Synthesis
Generated Control Signals

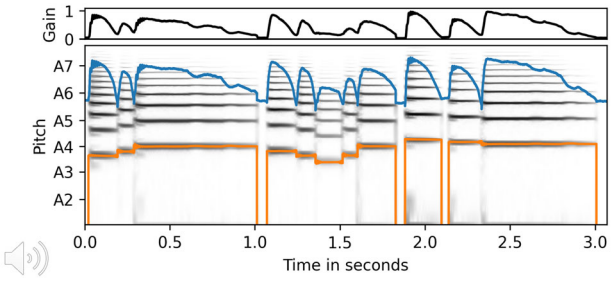


Wind Instrument Synthesis
Generated Control Signals with Pitch Fluctuations



Wind Instrument Synthesis

Generated Control Signals with Pitch & Loudness Fluctuations



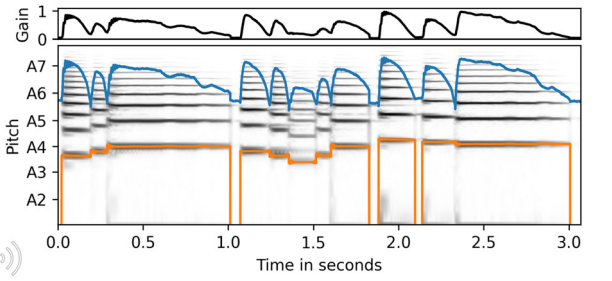
Loudness

Timbre

Pitch (F0)

Wind Instrument Synthesis

Generated Control Signals with Fluctuations and Room Acoustics



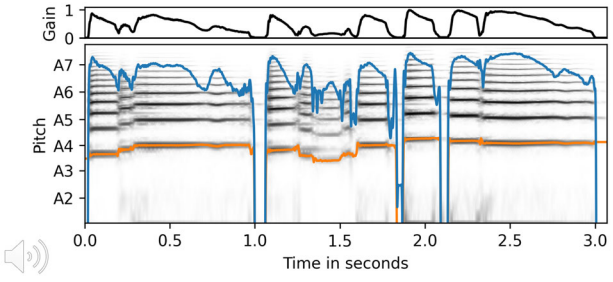
Loudness

Timbre

Pitch (F0)

Wind Instrument Synthesis

Resynthesized Trumpet



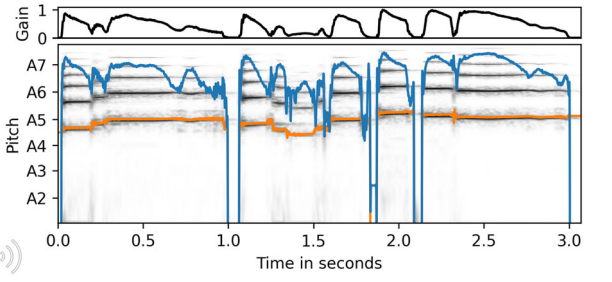
Loudness

Timbre

Pitch (F0)

Wind Instrument Synthesis

Resynthesized Trumpet (One Octave Higher)



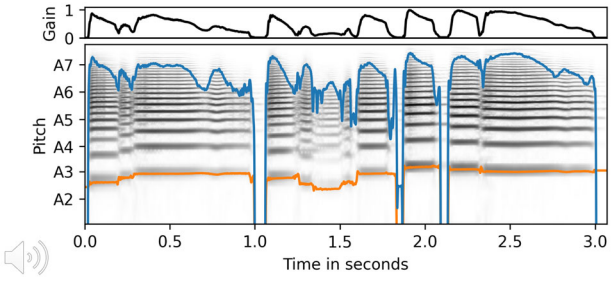
Loudness

Timbre

Pitch (F0)

Wind Instrument Synthesis

Resynthesized Trumpet (One Octave Lower)



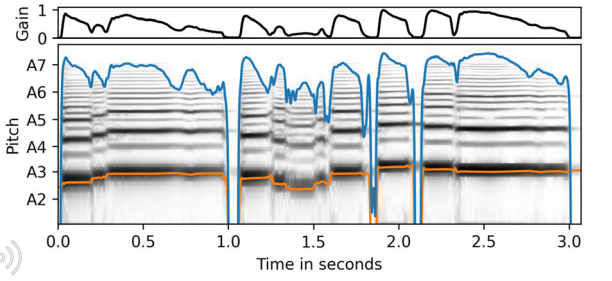
Loudness

Timbre

Pitch (F0)

Wind Instrument Synthesis

Resynthesized Clarinet (One Octave Lower)



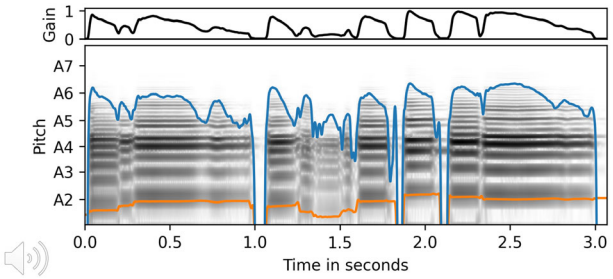
Loudness

Timbre

Pitch (F0)

Wind Instrument Synthesis

Resynthesized Bassoon (Two Octaves Lower)



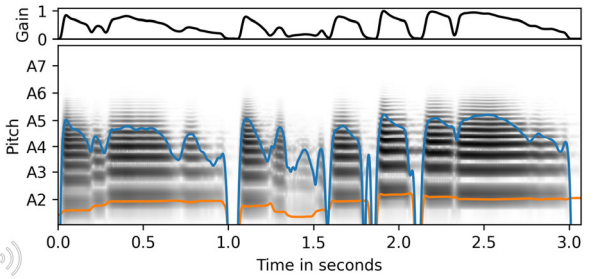
Loudness

Timbre

Pitch (F0)

Wind Instrument Synthesis

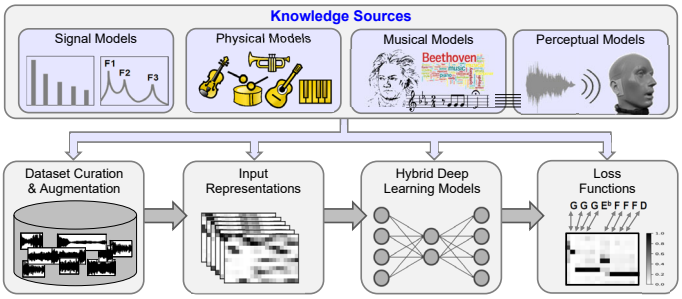
Resynthesized Tuba (Two Octaves Lower)



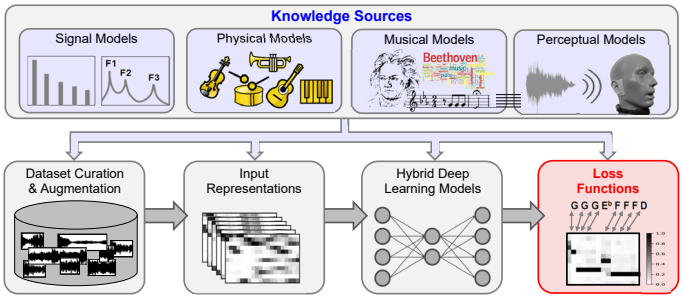
Loudness

Timbre

Pitch (F0)



Richard, Lostanlen, Yang, Müller: **Model-Based Deep Learning for Music Information Research**: Leveraging Diverse Knowledge Sources to Enhance Explainability, Controllability, and Resource Efficiency. IEEE Signal Processing Magazine, 41(6): 51–59, 2024



Richard, Lostanlen, Yang, Müller: **Model-Based Deep Learning for Music Information Research**: Leveraging Diverse Knowledge Sources to Enhance Explainability, Controllability, and Resource Efficiency. IEEE Signal Processing Magazine, 41(6): 51–59, 2024

Part 1: A Signal Processing Perspective

Key Topics

- Soft Dynamic Time Warping
- Piano Concerto Source Separation
- Wind Instrument Synthesis
- **Multi-Scale Spectral Loss**
- Hierarchical Classification Loss

Special Thanks

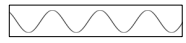
Simon Schwär

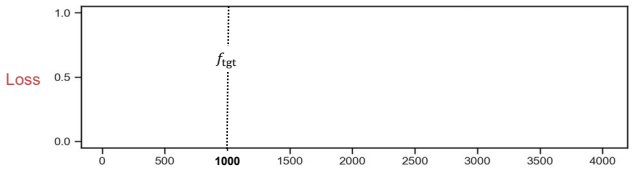
Literature

- Turian, Henry: I'm sorry for your loss: Spectrally-based audio distances are bad at pitch. Proc. Adv. Neural Inf. Process. Syst., 2020.
- Hayes, Saitis, Fazekas: Sinusoidal frequency estimation by gradient descent. Proc. ICASSP, 2023.
- Torres, Peeters, Richard: Unsupervised Harmonic Parameter Estimation Using DDSP and Spectral Optimal Transport. Proc. ICASSP, 2024
- Schwär, Müller: Multi-Scale Spectral Loss Revisited. IEEE Signal Processing Letters, 2023.
- Schwär: Differentiable Models for Music Signal Reconstruction, Adaptation, and Synthesis. PhD Thesis, FAU, 2026.

Example Scenario: Sinusoidal Frequency Estimation

Sinusoid with target frequency: $f_{\text{tgt}} = 1000$ Hz

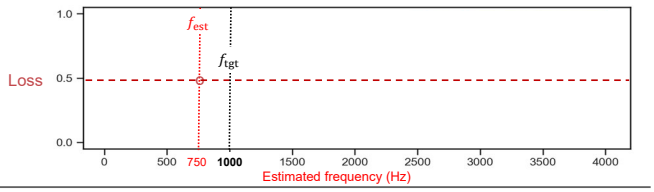




Example Scenario: Sinusoidal Frequency Estimation

Sinusoid with target frequency: $f_{\text{tgt}} = 1000$ Hz

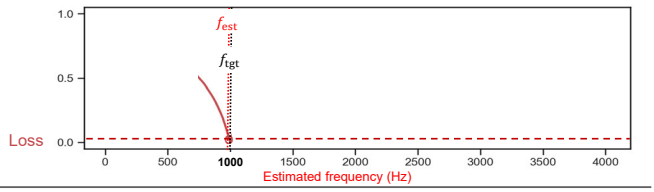
Sinusoid with estimated frequency: $f_{\text{est}} = 750$ Hz



Example Scenario: Sinusoidal Frequency Estimation

Sinusoid with target frequency: $f_{\text{tgt}} = 1000$ Hz

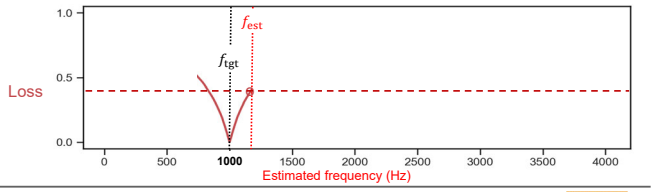
Sinusoid with estimated frequency: $f_{\text{est}} = 972$ Hz



Example Scenario: Sinusoidal Frequency Estimation

Sinusoid with target frequency: $f_{\text{tgt}} = 1000$ Hz

Sinusoid with estimated frequency: $f_{\text{est}} = 1100$ Hz

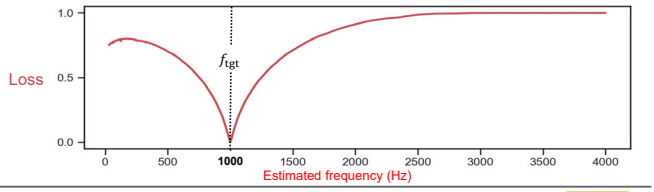


Example Scenario: Sinusoidal Frequency Estimation

Sinusoid with target frequency: $f_{\text{tgt}} = 1000$ Hz

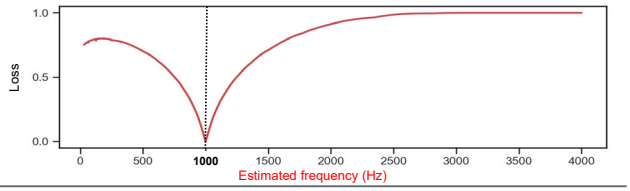
Sinusoidal sweep of estimated frequencies f_{est}

Loss landscape over **estimates** for a given **target**



Example Scenario: Sinusoidal Frequency Estimation

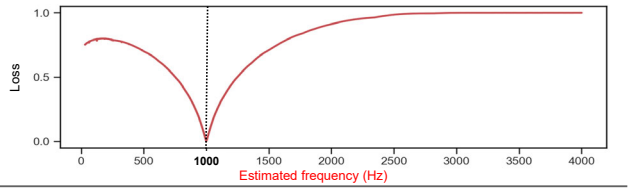
Loss landscape depends a lot on the chosen loss function to compare **estimated** and **target** signal



Example Scenario: Sinusoidal Frequency Estimation

Loss landscape depends a lot on the chosen loss function to compare **estimated** and **target** signal

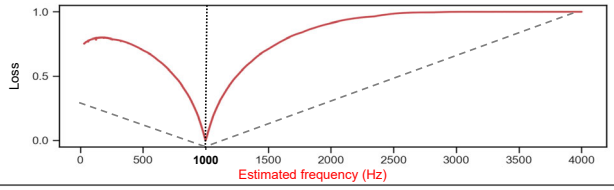
- Loss function discussed later



Example Scenario: Sinusoidal Frequency Estimation

Loss landscape depends a lot on the chosen loss function to compare **estimated** and **target** signal

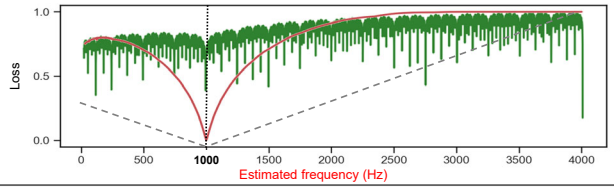
- Loss function discussed later
- Ideal convex loss



Example Scenario: Sinusoidal Frequency Estimation

Loss landscape depends a lot on the chosen loss function to compare **estimated** and **target** signal

- Loss function discussed later
- Ideal convex loss
- Multi-Scale Spectral (MSS) loss with standard settings

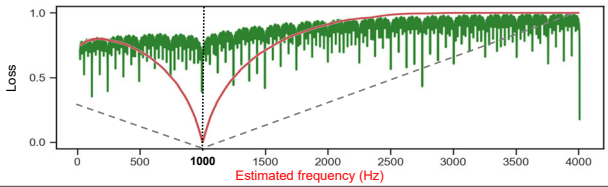


Example Scenario: Sinusoidal Frequency Estimation

Loss landscape depends a lot on the chosen loss function to compare **estimated** and **target** signal

- Loss function discussed later
- Ideal convex loss
- Multi-Scale Spectral (MSS) loss with standard settings

The MSS loss is what we widely use in audio processing (e.g., DDSP)



Multi-Scale Spectral Loss

- x input signal
- N window size
- H hop size
- w window function
- p compression function
- d distance function
- $\mathcal{N}$ set of window sizes
- $\mathcal{P}$ set of compression function

Configuration	Value	Description
Window Type	WR	Rectangular window
	WH	Hann window
	WF	Flat Top window
Window Size(s)	S1	$\mathcal{N} = \{64\}$
	S2	$\mathcal{N} = \{512\}$
	S3	$\mathcal{N} = \{2048\}$
	S4	$\mathcal{N} = \{64, 128, 256, 512, 1024, 2048\}$
	S5	$\mathcal{N} = \{67, 127, 257, 509, 1021, 2053\}$
Magnitude Compression	C0	$\mathcal{P} = \{x\}$
	C1	$\mathcal{P} = \{\log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C2	$\mathcal{P} = \{\log(1+\gamma x)\},\ \gamma = 1$
	C3	$\mathcal{P} = \{20\log_{10}(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C4	$\mathcal{P} = \{x,\ \log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
Matrix Distance	D1	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_1$
	D2	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_2^2$

Spectrum $$\mathcal{Y}_{w,N,p}(m,k) = p\left(\left|\sum_{n=0}^{N-1} x[n+mH]w[n]\exp\left(\frac{-i2\pi kn}{N}\right)\right|\right)$$

MSS loss $$\mathcal{L}_{\mathrm{MSS}}(x,\hat{x}) := \sum_{N\in\mathcal{N}}\sum_{p\in\mathcal{P}} d(\mathcal{Y}_{w,N,p}, \hat{\mathcal{Y}}_{w,N,p})$$

Multi-Scale Spectral Loss

- x input signal
- N window size
- H hop size
- w window function
- p compression function
- d distance function
- $\mathcal{N}$ set of window sizes
- $\mathcal{P}$ set of compression function

Configuration	Value	Description
Window Type	WR	Rectangular window
	WH	Hann window
	WF	Flat Top window
Window Size(s)	S1	$\mathcal{N} = \{64\}$
	S2	$\mathcal{N} = \{512\}$
	S3	$\mathcal{N} = \{2048\}$
	S4	$\mathcal{N} = \{64, 128, 256, 512, 1024, 2048\}$
	S5	$\mathcal{N} = \{67, 127, 257, 509, 1021, 2053\}$
Magnitude Compression	C0	$\mathcal{P} = \{x\}$
	C1	$\mathcal{P} = \{\log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C2	$\mathcal{P} = \{\log(1+\gamma x)\},\ \gamma = 1$
	C3	$\mathcal{P} = \{20\log_{10}(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C4	$\mathcal{P} = \{x,\ \log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
Matrix Distance	D1	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_1$
	D2	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_2^2$

Spectrum $$\mathcal{Y}_{w,N,p}(m,k) = p\left(\left|\sum_{n=0}^{N-1} x[n+mH]w[n]\exp\left(\frac{-i2\pi kn}{N}\right)\right|\right)$$

MSS loss $$\mathcal{L}_{\mathrm{MSS}}(x,\hat{x}) := \sum_{N\in\mathcal{N}}\sum_{p\in\mathcal{P}} d(\mathcal{Y}_{w,N,p}, \hat{\mathcal{Y}}_{w,N,p})$$

Multi-Scale Spectral Loss

- x input signal
- N window size
- H hop size
- w window function
- p compression function
- d distance function
- $\mathcal{N}$ set of window sizes
- $\mathcal{P}$ set of compression function

Configuration	Value	Description
Window Type	WR	Rectangular window
	WH	Hann window
	WF	Flat Top window
Window Size(s)	S1	$\mathcal{N} = \{64\}$
	S2	$\mathcal{N} = \{512\}$
	S3	$\mathcal{N} = \{2048\}$
	S4	$\mathcal{N} = \{64, 128, 256, 512, 1024, 2048\}$
	S5	$\mathcal{N} = \{67, 127, 257, 509, 1021, 2053\}$
Magnitude Compression	C0	$\mathcal{P} = \{x\}$
	C1	$\mathcal{P} = \{\log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C2	$\mathcal{P} = \{\log(1+\gamma x)\},\ \gamma = 1$
	C3	$\mathcal{P} = \{20\log_{10}(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C4	$\mathcal{P} = \{x,\ \log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
Matrix Distance	D1	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_1$
	D2	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_2^2$

Spectrum $$\mathcal{Y}_{w,N,p}(m,k) = p\left(\left|\sum_{n=0}^{N-1} x[n+mH]w[n]\exp\left(\frac{-i2\pi kn}{N}\right)\right|\right)$$

MSS loss $$\mathcal{L}_{\mathrm{MSS}}(x,\hat{x}) := \sum_{N\in\mathcal{N}}\sum_{p\in\mathcal{P}} d(\mathcal{Y}_{w,N,p}, \hat{\mathcal{Y}}_{w,N,p})$$

Multi-Scale Spectral Loss

- x input signal
- N window size
- H hop size
- w window function
- p compression function
- d distance function
- $\mathcal{N}$ set of window sizes
- $\mathcal{P}$ set of compression function

Configuration	Value	Description
Window Type	WR	Rectangular window
	WH	Hann window
	WF	Flat Top window
Window Size(s)	S1	$\mathcal{N} = \{64\}$
	S2	$\mathcal{N} = \{512\}$
	S3	$\mathcal{N} = \{2048\}$
	S4	$\mathcal{N} = \{64, 128, 256, 512, 1024, 2048\}$
	S5	$\mathcal{N} = \{67, 127, 257, 509, 1021, 2053\}$
Magnitude Compression	C0	$\mathcal{P} = \{x\}$
	C1	$\mathcal{P} = \{\log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C2	$\mathcal{P} = \{\log(1+\gamma x)\},\ \gamma = 1$
	C3	$\mathcal{P} = \{20\log_{10}(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C4	$\mathcal{P} = \{x,\ \log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
Matrix Distance	D1	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_1$
	D2	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_2^2$

Spectrum $$\mathcal{Y}_{w,N,p}(m,k) = p\left(\left|\sum_{n=0}^{N-1} x[n+mH]w[n]\exp\left(\frac{-i2\pi kn}{N}\right)\right|\right)$$

MSS loss $$\mathcal{L}_{\text{MSS}}(x,\hat{x}) := \sum_{N\in\mathcal{N}}\sum_{p\in\mathcal{P}} d(\mathcal{Y}_{w,N,p}, \hat{\mathcal{Y}}_{w,N,p})$$

Multi-Scale Spectral Loss

- x input signal
- N window size
- H hop size
- w window function
- p compression function
- d distance function
- $\mathcal{N}$ set of window sizes
- $\mathcal{P}$ set of compression function

Configuration	Value	Description
Window Type	WR	Rectangular window
	WH	Hann window
	WF	Flat Top window
Window Size(s)	S1	$\mathcal{N} = \{64\}$
	S2	$\mathcal{N} = \{512\}$
	S3	$\mathcal{N} = \{2048\}$
	S4	$\mathcal{N} = \{64, 128, 256, 512, 1024, 2048\}$
	S5	$\mathcal{N} = \{67, 127, 257, 509, 1021, 2053\}$
Magnitude Compression	C0	$\mathcal{P} = \{x\}$
	C1	$\mathcal{P} = \{\log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C2	$\mathcal{P} = \{\log(1+\gamma x)\},\ \gamma = 1$
	C3	$\mathcal{P} = \{20\log_{10}(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C4	$\mathcal{P} = \{x,\ \log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
Matrix Distance	D1	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_1$
	D2	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_2^2$

Spectrum $$\mathcal{Y}_{w,N,p}(m,k) = p\left(\left|\sum_{n=0}^{N-1} x[n+mH]w[n]\exp\left(\frac{-i2\pi kn}{N}\right)\right|\right)$$

MSS loss $$\mathcal{L}_{\text{MSS}}(x,\hat{x}) := \sum_{N\in\mathcal{N}}\sum_{p\in\mathcal{P}} d(\mathcal{Y}_{w,N,p}, \hat{\mathcal{Y}}_{w,N,p})$$

Multi-Scale Spectral Loss

- x input signal
- N window size
- H hop size
- w window function
- p compression function
- d distance function
- $\mathcal{N}$ set of window sizes
- $\mathcal{P}$ set of compression function

Configuration	Value	Description
Window Type	WR	Rectangular window
	WH	Hann window
	WF	Flat Top window
Window Size(s)	S1	$\mathcal{N} = \{64\}$
	S2	$\mathcal{N} = \{512\}$
	S3	$\mathcal{N} = \{2048\}$
	S4	$\mathcal{N} = \{64, 128, 256, 512, 1024, 2048\}$
	S5	$\mathcal{N} = \{67, 127, 257, 509, 1021, 2053\}$
Magnitude Compression	C0	$\mathcal{P} = \{x\}$
	C1	$\mathcal{P} = \{\log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C2	$\mathcal{P} = \{\log(1+\gamma x)\},\ \gamma = 1$
	C3	$\mathcal{P} = \{20\log_{10}(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C4	$\mathcal{P} = \{x,\ \log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
Matrix Distance	D1	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_1$
	D2	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_2^2$

Spectrum $$\mathcal{Y}_{w,N,p}(m,k) = p\left(\left|\sum_{n=0}^{N-1} x[n+mH]w[n]\exp\left(\frac{-i2\pi kn}{N}\right)\right|\right)$$

MSS loss $$\mathcal{L}_{\text{MSS}}(x,\hat{x}) := \sum_{N\in\mathcal{N}}\sum_{p\in\mathcal{P}} d(\mathcal{Y}_{w,N,p}, \hat{\mathcal{Y}}_{w,N,p})$$

Multi-Scale Spectral Loss

- x input signal
- N window size
- H hop size
- w window function
- p compression function
- d distance function
- $\mathcal{N}$ set of window sizes
- $\mathcal{P}$ set of compression function

Configuration	Value	Description
Window Type	WR	Rectangular window
	WH	Hann window
	WF	Flat Top window
Window Size(s)	S1	$\mathcal{N} = \{64\}$
	S2	$\mathcal{N} = \{512\}$
	S3	$\mathcal{N} = \{2048\}$
	S4	$\mathcal{N} = \{64, 128, 256, 512, 1024, 2048\}$
	S5	$\mathcal{N} = \{67, 127, 257, 509, 1021, 2053\}$
Magnitude Compression	C0	$\mathcal{P} = \{x\}$
	C1	$\mathcal{P} = \{\log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C2	$\mathcal{P} = \{\log(1+\gamma x)\},\ \gamma = 1$
	C3	$\mathcal{P} = \{20\log_{10}(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C4	$\mathcal{P} = \{x,\ \log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
Matrix Distance	D1	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_1$
	D2	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_2^2$

Spectrum $$\mathcal{Y}_{w,N,p}(m,k) = p\left(\left|\sum_{n=0}^{N-1} x[n+mH]w[n]\exp\left(\frac{-i2\pi kn}{N}\right)\right|\right)$$

MSS loss $$\mathcal{L}_{\text{MSS}}(x,\hat{x}) := \sum_{N\in\mathcal{N}}\sum_{p\in\mathcal{P}} d(\mathcal{Y}_{w,N,p}, \hat{\mathcal{Y}}_{w,N,p})$$

Multi-Scale Spectral Loss

- x input signal
- N window size
- H hop size
- w window function
- p compression function
- d distance function
- $\mathcal{N}$ set of window sizes
- $\mathcal{P}$ set of compression function

Configuration	Value	Description
Window Type	WR	Rectangular window
	WH	Hann window
	WF	Flat Top window
Window Size(s)	S1	$\mathcal{N} = \{64\}$
	S2	$\mathcal{N} = \{512\}$
	S3	$\mathcal{N} = \{2048\}$
	S4	$\mathcal{N} = \{64, 128, 256, 512, 1024, 2048\}$
	S5	$\mathcal{N} = \{67, 127, 257, 509, 1021, 2053\}$
Magnitude Compression	C0	$\mathcal{P} = \{x\}$
	C1	$\mathcal{P} = \{\log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C2	$\mathcal{P} = \{\log(1+\gamma x)\},\ \gamma = 1$
	C3	$\mathcal{P} = \{20\log_{10}(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C4	$\mathcal{P} = \{x,\ \log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
Matrix Distance	D1	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_1$
	D2	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_2^2$

Spectrum $$\mathcal{Y}_{w,N,p}(m,k) = p\left(\left|\sum_{n=0}^{N-1} x[n+mH]w[n]\exp\left(\frac{-i2\pi kn}{N}\right)\right|\right)$$

MSS loss $$\mathcal{L}_{\text{MSS}}(x,\hat{x}) := \sum_{N\in\mathcal{N}}\sum_{p\in\mathcal{P}} d(\mathcal{Y}_{w,N,p}, \hat{\mathcal{Y}}_{w,N,p})$$

Multi-Scale Spectral Loss

- x input signal
- N window size
- H hop size
- w window function
- p compression function
- d distance function
- $\mathcal{N}$ set of window sizes
- $\mathcal{P}$ set of compression function

MSS loss with standard settings: (WH, S4, C4, D1)

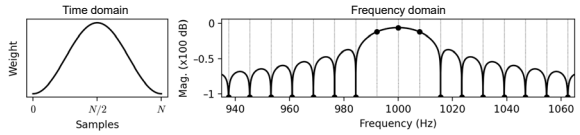
Configuration	Value	Description
Window Type	WR	Rectangular window
	WH	Hann window
	WF	Flat Top window
Window Size(s)	S1	$\mathcal{N} = \{64\}$
	S2	$\mathcal{N} = \{512\}$
	S3	$\mathcal{N} = \{2048\}$
	S4	$\mathcal{N} = \{64, 128, 256, 512, 1024, 2048\}$
	S5	$\mathcal{N} = \{67, 127, 257, 509, 1021, 2053\}$
Magnitude Compression	C0	$\mathcal{P} = \{x\}$
	C1	$\mathcal{P} = \{\log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C2	$\mathcal{P} = \{\log(1+\gamma x)\},\ \gamma = 1$
	C3	$\mathcal{P} = \{20\log_{10}(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
	C4	$\mathcal{P} = \{x,\ \log(x+\varepsilon)\},\ \varepsilon = 10^{-7}$
Matrix Distance	D1	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_1$
	D2	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_2^2$

Spectrum $$\mathcal{Y}_{w,N,p}(m,k) = p\left(\left|\sum_{n=0}^{N-1} x[n+mH]w[n]\exp\left(\frac{-i2\pi kn}{N}\right)\right|\right)$$

MSS loss $$\mathcal{L}_{\text{MSS}}(x,\hat{x}) := \sum_{N\in\mathcal{N}}\sum_{p\in\mathcal{P}} d(\mathcal{Y}_{w,N,p}, \hat{\mathcal{Y}}_{w,N,p})$$

Spectrum-Based Distance

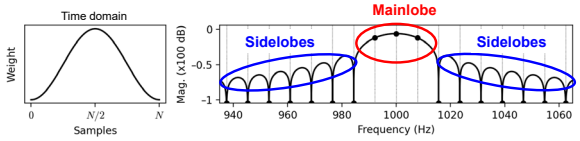
Hann window



- Input signal: Sinusoid with frequency $f = 1000$ Hz

Spectrum-Based Distance

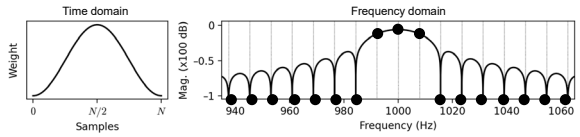
Hann window



- Input signal: Sinusoid with frequency $f = 1000$ Hz
- STFT → Spectral leakage due to windowing

Spectrum-Based Distance

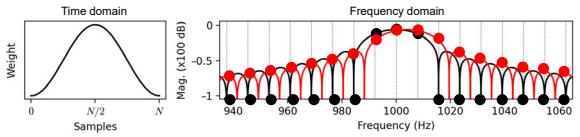
Hann window



- Input signal: Sinusoid with frequency $f = 1000$ Hz
- STFT → Spectral leakage due to windowing
- Discrete STFT → **Frequency grid**

Spectrum-Based Distance

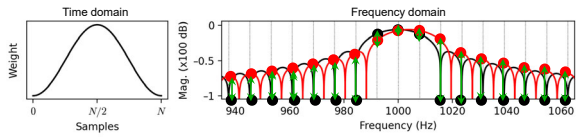
Hann window



- Input signal: Sinusoid with frequency $f = 1000$ Hz
- STFT → Spectral leakage due to windowing
- Discrete STFT → **Frequency grid**
- Second signal: Sinusoid with frequency $f = 1003.9$ Hz

Spectrum-Based Distance

Hann window

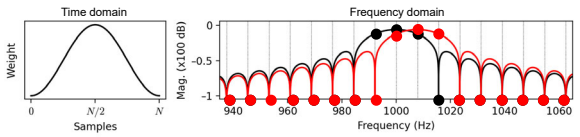


- Input signal: Sinusoid with frequency $f = 1000$ Hz
- STFT → Spectral leakage due to windowing
- Discrete STFT → **Frequency grid**
- Second signal: Sinusoid with frequency $f = 1003.9$ Hz

Distance depends on
- Grid sampling
- Mainlobe & sidelobes
- Window type
- STFT parameters

Spectrum-Based Distance

Hann window

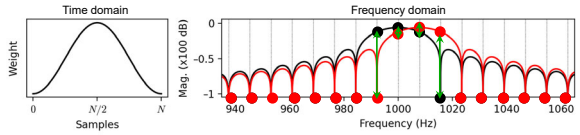


- Input signal: Sinusoid with frequency $f = 1000$ Hz
- STFT → Spectral leakage due to windowing
- Discrete STFT → **Frequency grid**
- Second signal: Sinusoid with frequency $f = 1007.8$ Hz

Distance depends on
- Grid sampling
- Mainlobe & sidelobes
- Window type
- STFT parameters

Spectrum-Based Distance

Hann window

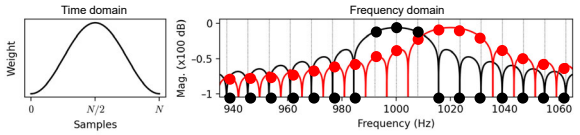


- Input signal: Sinusoid with frequency $f = 1000$ Hz
- STFT → Spectral leakage due to windowing
- Discrete STFT → **Frequency grid**
- Second signal: Sinusoid with frequency $f = 1007.8$ Hz

Distance depends on
- Grid sampling
- Mainlobe & sidelobes
- Window type
- STFT parameters

Spectrum-Based Distance

Hann window

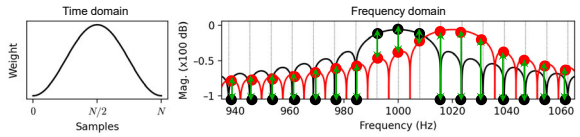


- Input signal: Sinusoid with frequency $f = 1000$ Hz
- STFT → Spectral leakage due to windowing
- Discrete STFT → **Frequency grid**
- Second signal: Sinusoid with frequency $f = 1020$ Hz

Distance depends on
- Grid sampling
- Mainlobe & sidelobes
- Window type
- STFT parameters

Spectrum-Based Distance

Hann window

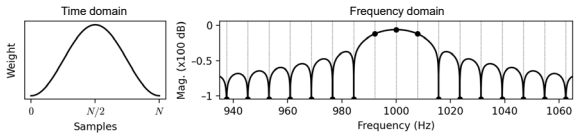


- Input signal: Sinusoid with frequency $f = 1000$ Hz
- STFT → Spectral leakage due to windowing
- Discrete STFT → **Frequency grid**
- Second signal: Sinusoid with frequency $f = 1020$ Hz

Distance depends on
- Grid sampling
- Mainlobe & sidelobes
- Window type
- STFT parameters

Dependency: Window Type

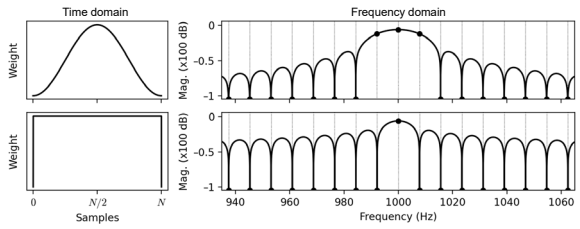
Hann window



Dependency: Window Type

Hann window

Rectangular window

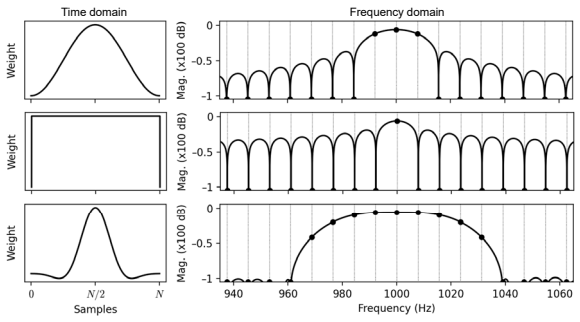


Dependency: Window Type

Hann window

Rectangular window

Flattop window

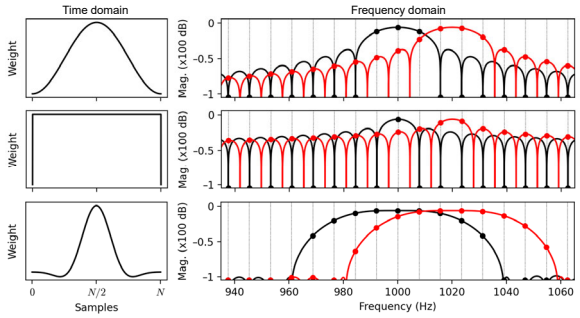


Dependency: Window Type

Hann window

Rectangular window

Flattop window



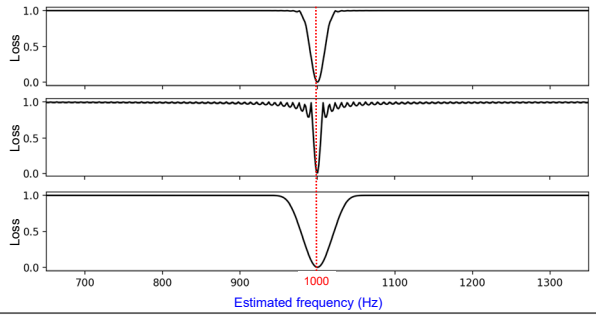
Dependency: Window Type

Loss landscape over estimates for a given target

Hann window

Rectangular window

Flattop window

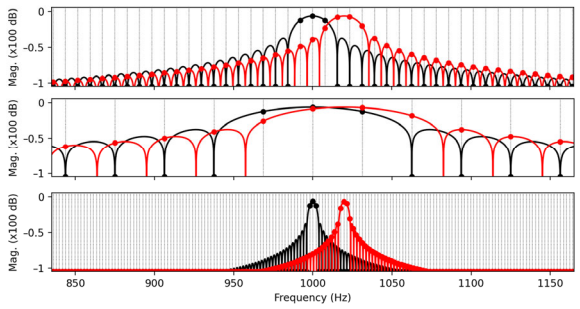


Dependency: Window Size

$N = 2048$

$N = 512$

$N = 8192$



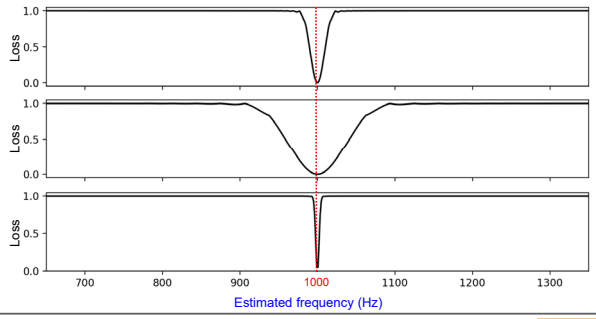
Dependency: Window Size

Loss landscape over estimates for a given target

$N = 2048$

$N = 512$

$N = 8192$

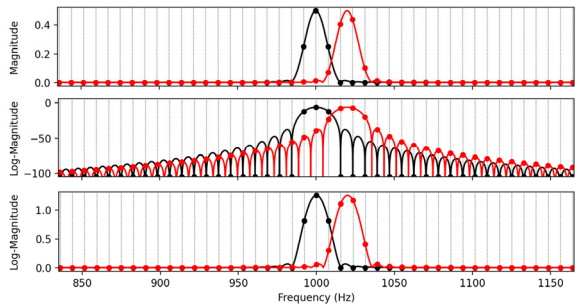


Dependency: Magnitude Compression

None

Decibels

Log(1+value)



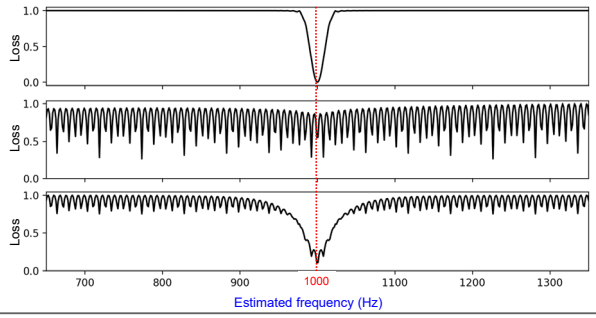
Dependency: Magnitude Compression

Loss landscape over estimates for a given target

None

Decibels

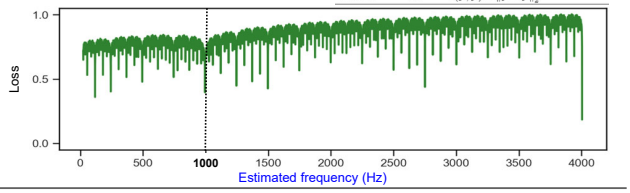
Log(1+value)



Experiments

- MSS loss with standard settings (WH, S4, C4, D1)

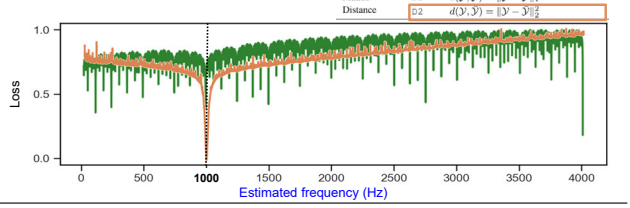
Configuration	Value	Description
Window Type	WR	Rectangular window
	WH	Hann window
	WF	Flat Top window
Window Size(s)	S1	$\mathcal{N} = \{64\}$
	S2	$\mathcal{N} = \{512\}$
	S3	$\mathcal{N} = \{2048\}$
	S4	$\mathcal{N} = \{64, 128, 256, 512, 1024, 2048\}$
	S5	$\mathcal{N} = \{67, 127, 257, 509, 1021, 2053\}$
Magnitude Compression	C0	$\mathcal{P} = \{x\}$
	C1	$\mathcal{P} = \{\log(x+\varepsilon)\}, \varepsilon = 10^{-7}$
	C2	$\mathcal{P} = \{\log(1+\gamma x)\}, \gamma = 1$
	C3	$\mathcal{P} = \{20\log_{10}(x+\varepsilon)\}, \varepsilon = 10^{-7}$
	C4	$\mathcal{P} = \{x, \log(x+\varepsilon)\}, \varepsilon = 10^{-7}$
Matrix Distance	D1	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_1$
	D2	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_2^2$



Experiments

- MSS loss with standard settings (WH, S4, C4, D1)
- Modified Hann MSS (WH, S5, C4, D2)

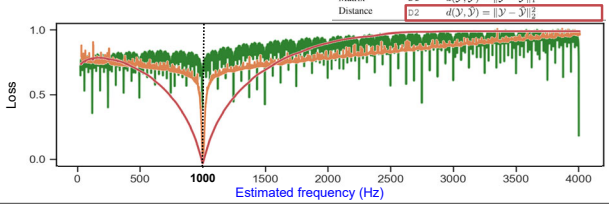
Configuration	Value	Description
Window Type	WR	Rectangular window
	WH	Hann window
	WF	Flat Top window
Window Size(s)	S1	$\mathcal{N} = \{64\}$
	S2	$\mathcal{N} = \{512\}$
	S3	$\mathcal{N} = \{2048\}$
	S4	$\mathcal{N} = \{64, 128, 256, 512, 1024, 2048\}$
	S5	$\mathcal{N} = \{67, 127, 257, 509, 1021, 2053\}$
Magnitude Compression	C0	$\mathcal{P} = \{x\}$
	C1	$\mathcal{P} = \{\log(x+\varepsilon)\}, \varepsilon = 10^{-7}$
	C2	$\mathcal{P} = \{\log(1+\gamma x)\}, \gamma = 1$
	C3	$\mathcal{P} = \{20\log_{10}(x+\varepsilon)\}, \varepsilon = 10^{-7}$
	C4	$\mathcal{P} = \{x, \log(x+\varepsilon)\}, \varepsilon = 10^{-7}$
Matrix Distance	D1	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_1$
	D2	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_2^2$



Experiments

- MSS loss with standard settings (WH, S4, C4, D1)
- Modified Hann MSS (WH, S5, C4, D2)
- Smooth MSS (WF, S5, C2, D2)

Configuration	Value	Description
Window Type	WR	Rectangular window
	WH	Hann window
	WF	Flat Top window
Window Size(s)	S1	$\mathcal{N} = \{64\}$
	S2	$\mathcal{N} = \{512\}$
	S3	$\mathcal{N} = \{2048\}$
	S4	$\mathcal{N} = \{64, 128, 256, 512, 1024, 2048\}$
	S5	$\mathcal{N} = \{67, 127, 257, 509, 1021, 2053\}$
Magnitude Compression	C0	$\mathcal{P} = \{x\}$
	C1	$\mathcal{P} = \{\log(x+\varepsilon)\}, \varepsilon = 10^{-7}$
	C2	$\mathcal{P} = \{\log(1+\gamma x)\}, \gamma = 1$
	C3	$\mathcal{P} = \{20\log_{10}(x+\varepsilon)\}, \varepsilon = 10^{-7}$
	C4	$\mathcal{P} = \{x, \log(x+\varepsilon)\}, \varepsilon = 10^{-7}$
Matrix Distance	D1	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_1$
	D2	$d(\mathcal{Y}, \hat{\mathcal{Y}}) = \|\mathcal{Y} - \hat{\mathcal{Y}}\|_2^2$

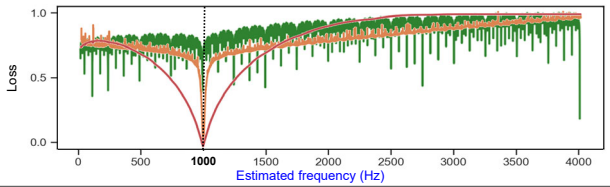


Experiments

GRA (Gradient-Sign Ranking Accuracy)

- Measures how often the loss gradient points in the correct direction.
- **Step size** distinguishes local gradient behavior from global trend.

Configuration	GRA			
Step Size	0.3 ct.	3 ct.	30 ct.	300 ct.
Standard MSS	0.523	0.529	0.573	0.775
Modified Hann MSS	0.613	0.635	0.708	0.923
Smooth MSS	0.999	0.993	0.952	0.860



Part 1: A Signal Processing Perspective
Key Topics

- Soft Dynamic Time Warping
- Piano Concerto Source Separation
- Wind Instrument Synthesis
- Multi-Scale Spectral Loss
- **Hierarchical Classification Loss**

Special Thanks

Michael Krause

Literature

- Silla, Freitas: A survey of hierarchical classification across different application domains. Data Mining and Knowledge Discovery, 2011.
- Wehrmann, Cerri, Barros: Hierarchical multi-label classification networks. Proc. ICML, 2018.
- Weiß et al.: Wagner Ring Dataset: A Complex Opera Scenario for Music Processing and Computational Musicology. TISMIR, 2023.
- Krause, Müller: Hierarchical Classification for Instrument Activity Detection in Orchestral Music Recordings. IEEE/ACM TASLP 2023.
- Krause: Activity Detection for Sound Events in Orchestral Music Recordings. PhD Thesis, FAU, 2023.

Wagner Ring Dataset

- Tetralogy (four operas)

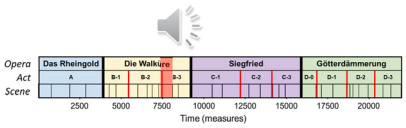
Opera	Das Rheingold	Die Walküre	Siegfried	Götterdämmerung

Wagner Ring Dataset

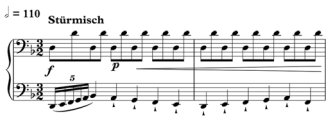
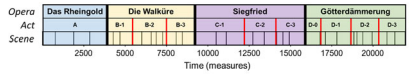
- Tetralogy (four operas)
- 11 Acts

Wagner Ring Dataset

- Tetralogy (four operas)
- 11 Acts
- 21,939 measures

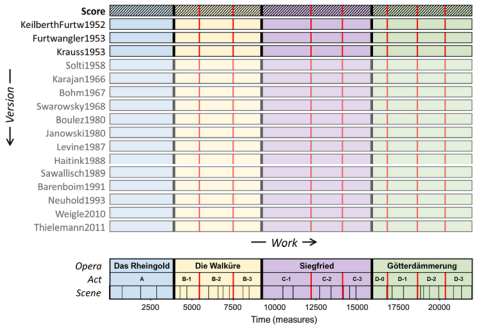


Wagner Ring Dataset
Raw Data

- Symbolic score:
  - Piano reduction
  - 822 pages



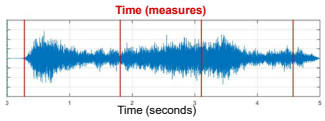
Wagner Ring Dataset
Raw Data

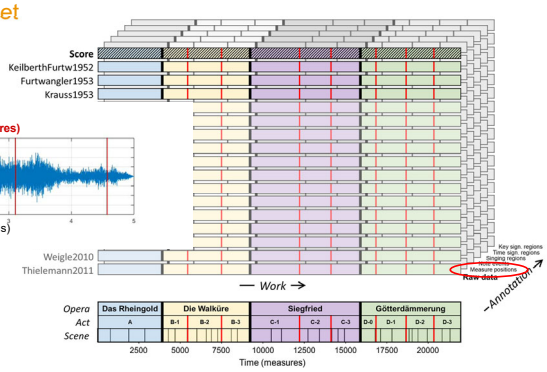
- Symbolic score:
  - Piano reduction
  - 822 pages
- Audio recordings:
  - 16 performances
  - 232 hours
  - 3 performances in Public Domain (EU)



Wagner Ring Dataset
Annotations

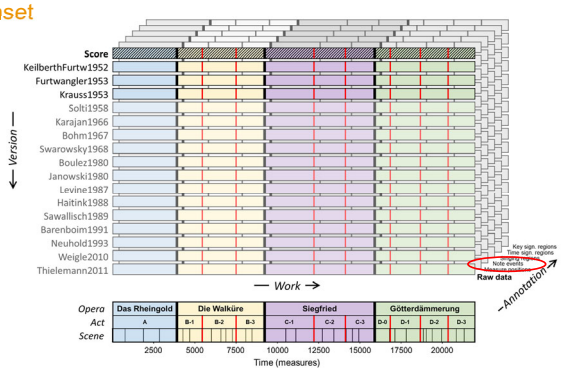
- Measure positions





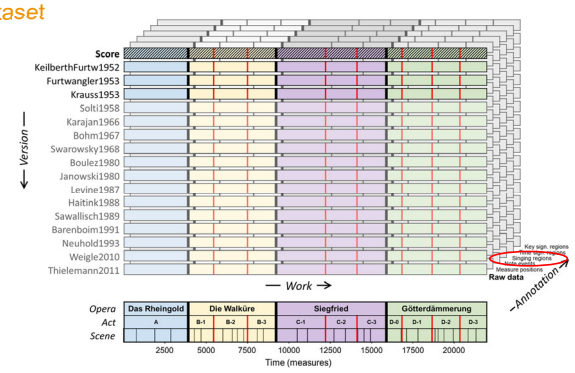
Wagner Ring Dataset
Annotations

- Measure positions
- Note events



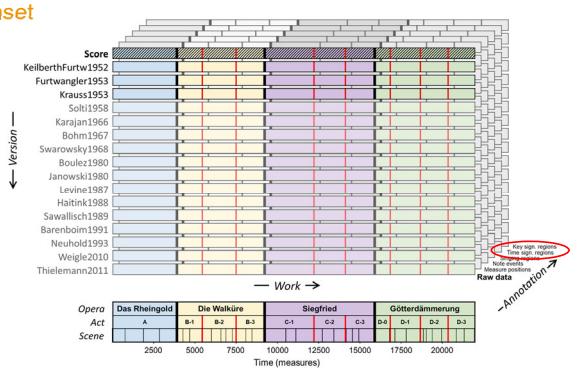
Wagner Ring Dataset
Annotations

- Measure positions
- Note events
- Singing regions



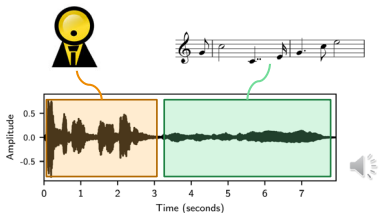
Wagner Ring Dataset
Annotations

- Measure positions
- Note events
- Singing regions
- Time signatures
- Key signatures



PhD Thesis by Michael Krause (2023)
Activity Detection for Sound Events in Orchestral Music Recordings

Singing Voice Detection | Leitmotif Detection | Instrument Detection



Hierarchical Classification
Singing Voice Detection

	Levels
Singing activity	Activity

Hierarchical Classification
Singing Voice Detection

	Levels
Singing activity	Activity
Male, Female	Gender

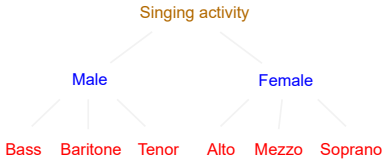
Hierarchical Classification
Singing Voice Detection

	Levels
Singing activity	Activity
Male, Female	Gender
Bass, Baritone, Tenor, Alto, Mezzo, Soprano	Voice type

Hierarchical Strategies for Activity Detection

- Strategy A: Independent Decisions
- Strategy B: Bottom-Up Aggregation
- Strategy C: Top-Down Divide-and-Conquer
- Strategy D: Joint Classification
- Strategy $D^{\alpha,\beta}$: Joint Classification with Consistency Losses

Hierarchical Strategies for Activity Detection
Strategy A: Independent Decisions

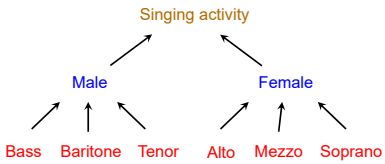


- Train and evaluate separate models for each hierarchy level
  - Activity classifier
  - Gender classifier
  - Voice type classifier

Hierarchical Strategies for Activity Detection
Strategy A: Independent Decisions

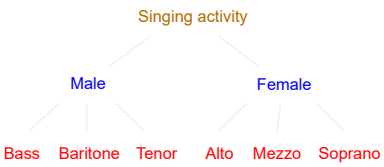
- Train and evaluate separate models for each hierarchy level
  - Activity classifier
  - Gender classifier
  - Voice type classifier
- Outputs may be inconsistent

Hierarchical Strategies for Activity Detection
Strategy B: Bottom-Up Aggregation



- Train and evaluate a single model for the lowest hierarchy level
  - Voice type classifier
- Aggregate results from lower levels
- Consistency is trivially fulfilled
- May cause poor predictions on upper levels due to error propagation

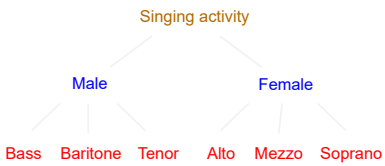
Hierarchical Strategies for Activity Detection
Strategy D: Joint Classification



- Train and evaluate a single model for all classes

  → **Multi-task model**
- Need additional loss terms to promote consistent predictions

Hierarchical Strategies for Activity Detection
Strategy $D^{\alpha,\beta}$: Joint Classification with Consistency Losses

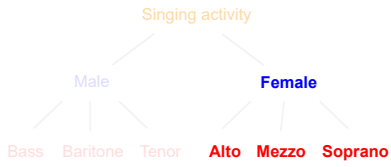


Hierarchical Strategies for Activity Detection
Strategy $D^{\alpha,\beta}$: Joint Classification with Consistency Losses

- Notation:
  - c: a class
  - p_c: probability of c

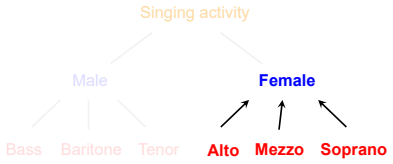
Hierarchical Strategies for Activity Detection
Strategy $D^{\alpha,\beta}$: Joint Classification with Consistency Losses

- Notation:
  - c: a class
  - p_c: probability of c
  - $c\downarrow$: child classes of c



Hierarchical Strategies for Activity Detection
Strategy $D^{\alpha,\beta}$: Joint Classification with Consistency Losses

- Notation:
  - c: a class
  - p_c: probability of c
  - $c\downarrow$: child classes of c



- For **bottom-up** consistency, minimize

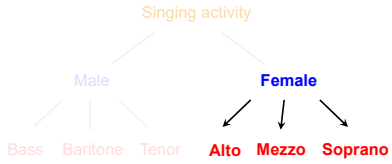
$$\sum_{c' \in c\downarrow} \max\{0, p_{c'} - p_c\}^2$$

p_c should be at least as high as any $p_{c'}$

→ penalty for every $p_{c'} > p_c$

Hierarchical Strategies for Activity Detection
Strategy $D^{\alpha,\beta}$: Joint Classification with Consistency Losses

- Notation:
  - c: a class
  - p_c: probability of c
  - $c\downarrow$: child classes of c



- For **top-down** consistency, minimize

$$\max\{0, p_c - \max_{c' \in c\downarrow} p_{c'}\}^2$$

p_c should not be above largest $p_{c'}$

Hierarchical Strategies for Activity Detection
Strategy $D^{\alpha,\beta}$: Joint Classification with Consistency Losses

Bottom-up loss term:

$$\mathcal{L}_{\uparrow} = \frac{1}{|\mathbf{C} \setminus \mathbf{C}^H|} \sum_{h=2}^{H} \sum_{c \in \mathbf{C}^h} \sum_{c' \in c\downarrow} \max\{0, p_{c'} - p_c\}^2$$

Top-down loss term:

$$\mathcal{L}_{\downarrow} = \frac{1}{|\mathbf{C} \setminus \mathbf{C}^1|} \sum_{h=2}^{H} \sum_{c \in \mathbf{C}^h} \max\{0, p_c - \max_{c' \in c\downarrow} p_{c'}\}^2$$

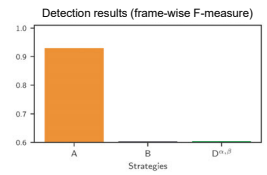
Joint loss term:

$$\mathcal{L} = \mathcal{L}_{\text{BCE}} + \alpha\mathcal{L}_{\downarrow} + \beta\mathcal{L}_{\uparrow}$$

Notation

$\mathbf{C}$	All classes
$\mathbf{C}^h$	Classes at level h
H	Number of levels
$c\downarrow$	Children of c
p_c	Probability for c

Results: Female Singing



Consistency

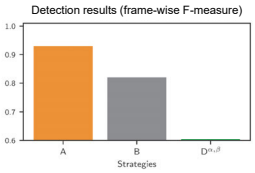
$\mathcal{I}_c^{\text{Est}}$ Frames predicted as c

$\mathcal{I}_{c\downarrow}^{\text{Est}}$ Frames predicted as child of c

$$\gamma_c = \frac{|\mathcal{I}_c^{\text{Est}} \cap \mathcal{I}_{c\downarrow}^{\text{Est}}|}{|\mathcal{I}_c^{\text{Est}} \cup \mathcal{I}_{c\downarrow}^{\text{Est}}|}$$

- Strategy A (Independent Decisions) yields good but inconsistent results

Results: Female Singing



Consistency

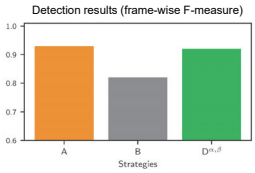
$\mathcal{I}_c^{\mathrm{Est}}$ Frames predicted as c

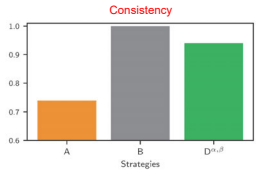
$\mathcal{I}_{c\downarrow}^{\mathrm{Est}}$ Frames predicted as child of c

$$\gamma_c = \frac{|\mathcal{I}_c^{\mathrm{Est}} \cap \mathcal{I}_{c\downarrow}^{\mathrm{Est}}|}{|\mathcal{I}_c^{\mathrm{Est}} \cup \mathcal{I}_{c\downarrow}^{\mathrm{Est}}|}$$

- Strategy A (Independent Decisions) yields good but inconsistent results
- Strategy B (Bottom-Up Aggregation) gives worse but consistent results

Results: Female Singing





Consistency

$\mathcal{I}_c^{\mathrm{Est}}$ Frames predicted as c

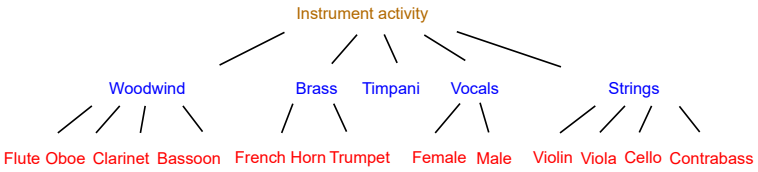
$\mathcal{I}_{c\downarrow}^{\mathrm{Est}}$ Frames predicted as child of c

$$\gamma_c = \frac{|\mathcal{I}_c^{\mathrm{Est}} \cap \mathcal{I}_{c\downarrow}^{\mathrm{Est}}|}{|\mathcal{I}_c^{\mathrm{Est}} \cup \mathcal{I}_{c\downarrow}^{\mathrm{Est}}|}$$

- Strategy A (Independent Decisions) yields good but inconsistent results
- Strategy B (Bottom-Up Aggregation) gives worse but consistent results
- Strategy $D^{\alpha,\beta}$ (Joint with Consistency Losses) provides good trade-off

Scenario: Hierarchical Instrument Classification

- Musical instruments can naturally be arranged into hierarchies



- Instrument-level annotations hard to obtain

Part 1: A Signal Processing Perspective
Conclusions

- Signal processing matters, even in deep learning
- LLMs may do the work, understanding is still your job!
- We've explored the signals … now let's explore the music

Part 2: A Computational Musicology Perspective