Hi! I'm **Marco** Pasini

Bachelor in
Industrial Engineering
(2020)

Master in
Artificial Intelligence
(2022)

Research **Intern** at
Sony CSL Paris
(2023)

Hi! I’m **Marco** Pasini

Voice Conversion

Hi! I’m **Marco** Pasini

Voice Conversion  **Music** Style Transfer

Hi! I’m **Marco** Pasini

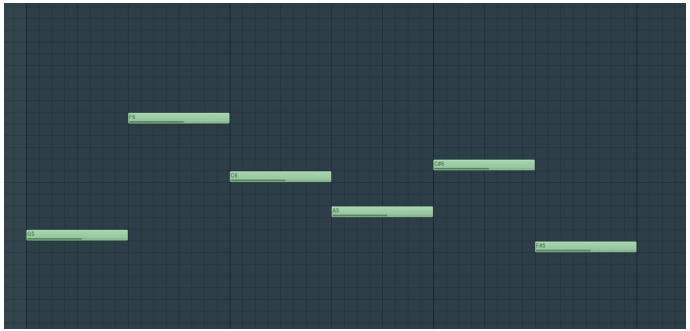
Voice Conversion → **Music** Style Transfer → **Music** Generation

Fast Infinite Waveform Music Generation

Marco Pasini, Jan Schlüter

Music Generation

Music Generation

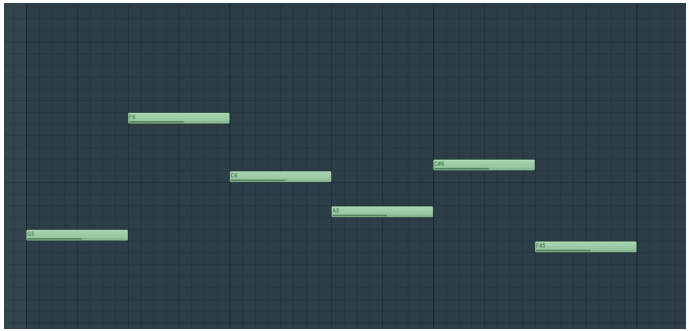


Symbolic

Constrained!

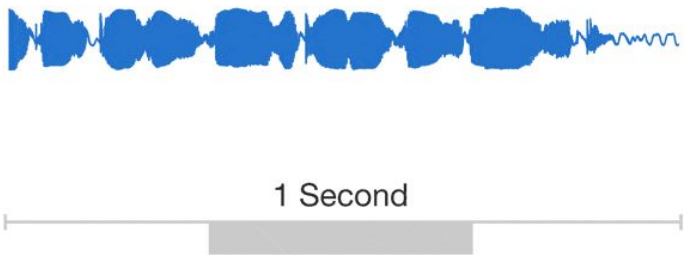
https://benanne.github.io/2020/03/24/audio-generation.html

Music Generation



Symbolic

Constrained!



Waveform

No Constraints!

https://benanne.github.io/2020/03/24/audio-generation.html

State-of-the-art

State-of-the-art: OpenAI **Jukebox**

https://openai.com/blog/jukebox/

State-of-the-art: OpenAI **Jukebox**

Encodes music into discrete codes

https://openai.com/blog/jukebox/

State-of-the-art: OpenAI **Jukebox**

Encodes music into discrete codes

Generates sequences with transformers

https://openai.com/blog/jukebox/

State-of-the-art: OpenAI **Jukebox**

Encodes music into discrete codes

Generates sequences with transformers

Genre/Lyrics **Conditioning**

https://openai.com/blog/jukebox/

State-of-the-art: OpenAI **Jukebox**

Encodes music into discrete codes

Generates sequences with transformers

Genre/Lyrics **Conditioning**

Slow! 8 hours for 1 minute of music

https://openai.com/blog/jukebox/

State-of-the-art: OpenAI **Jukebox**

*“**Pop**, in the style of **Beatles**”*

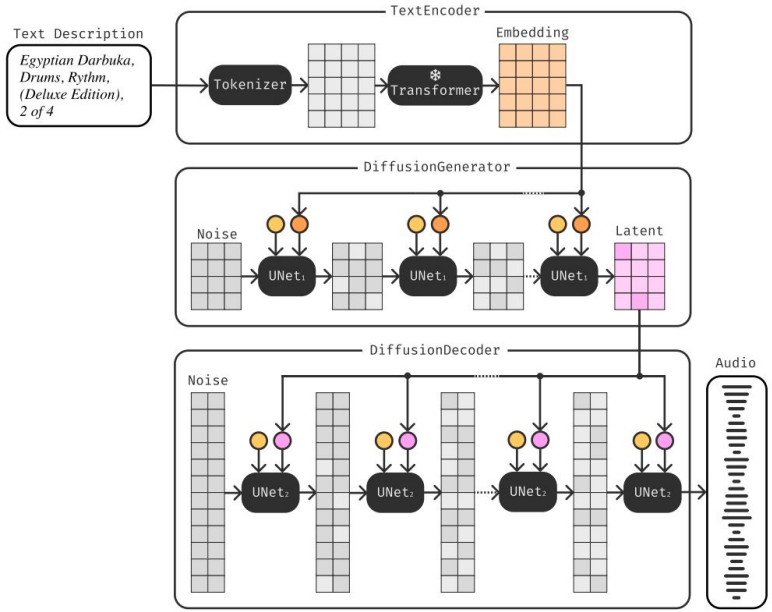
https://openai.com/blog/jukebox/

State-of-the-art: OpenAI **Jukebox**

*“**Pop**, in the style of **Beatles**”*

https://openai.com/blog/jukebox/

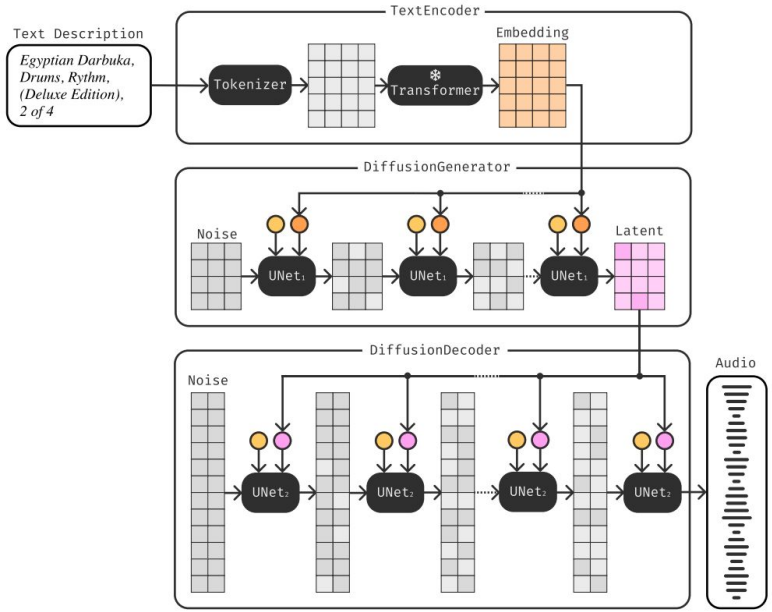
State-of-the-art: **Moûsai**



https://arxiv.org/pdf/2301.11757.pdf

State-of-the-art: **Moûsai**

Latent Diffusion approach

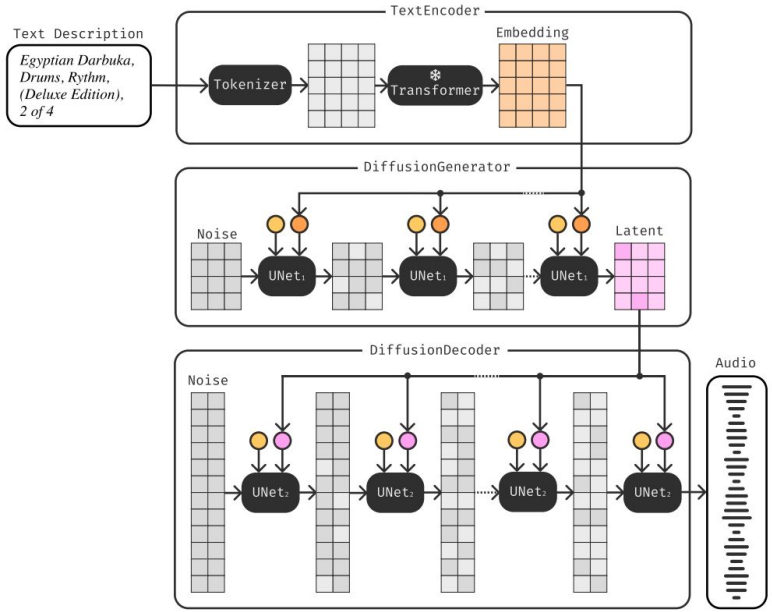


https://arxiv.org/pdf/2301.11757.pdf

State-of-the-art: **Moûsai**

Latent Diffusion approach

Text prompts **Conditioning**



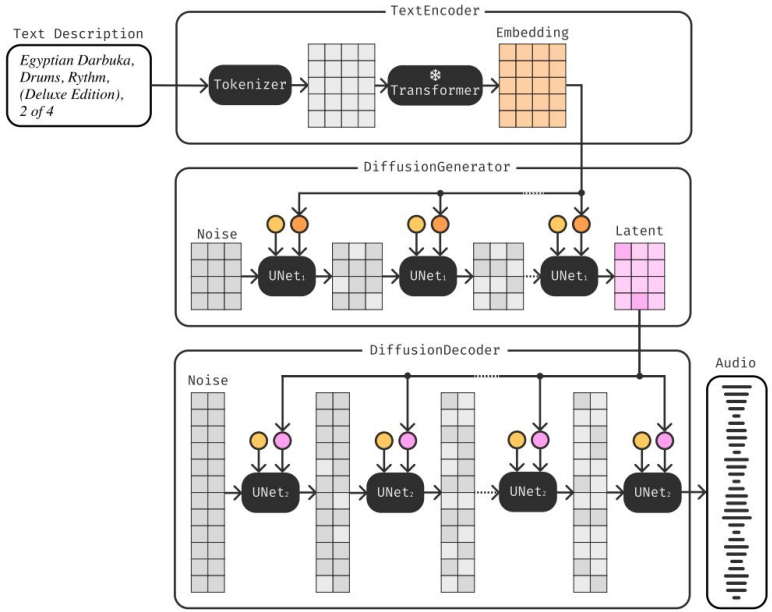
https://arxiv.org/pdf/2301.11757.pdf

State-of-the-art: **Moûsai**

Latent Diffusion approach

Text prompts **Conditioning**

48 kHz **stereo** music



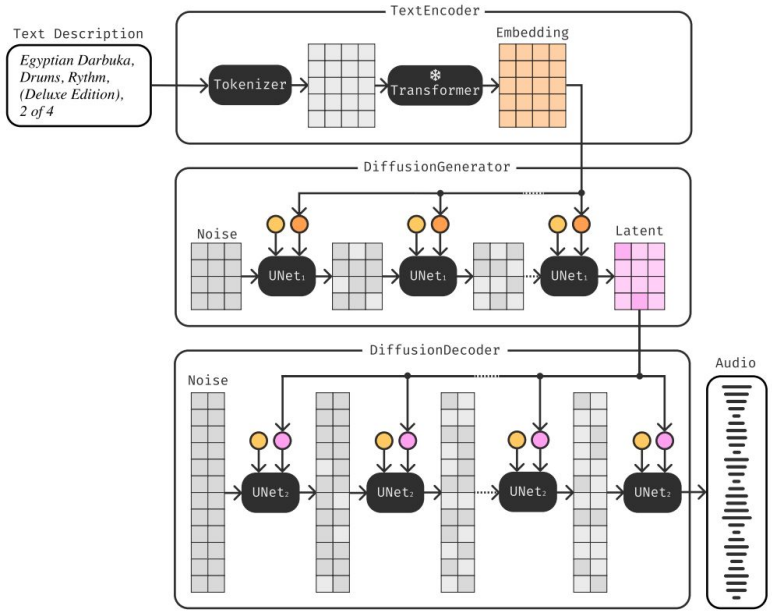
https://arxiv.org/pdf/2301.11757.pdf

State-of-the-art: **Moûsai**

Latent Diffusion approach

Text prompts **Conditioning**

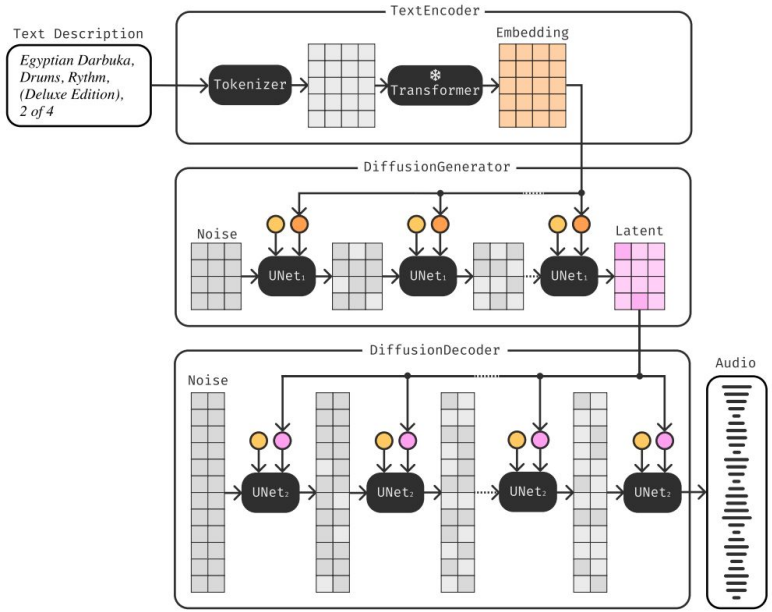
48 kHz **stereo** music



Slow/**Expensive** Training

https://arxiv.org/pdf/2301.11757.pdf

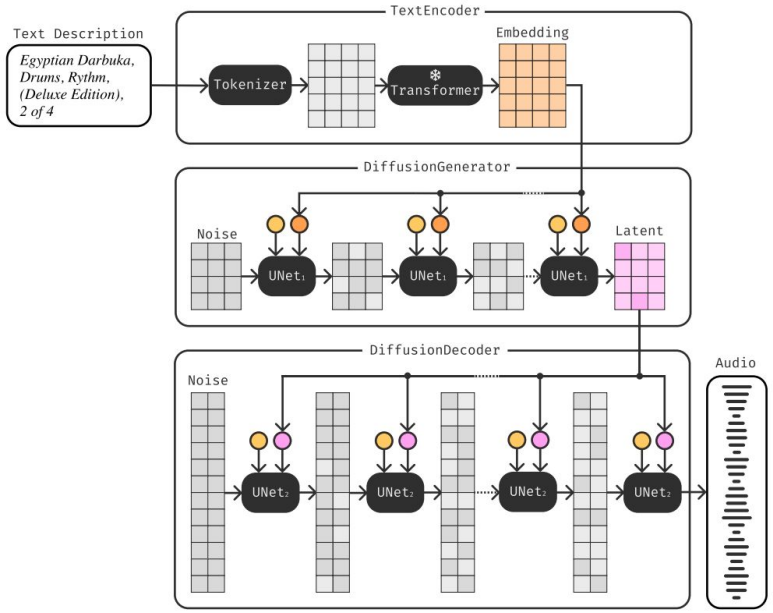
State-of-the-art: **Moûsai**



"***Electro House*** *(Remix), 2023, 3 of 4:"*

https://arxiv.org/pdf/2301.11757.pdf

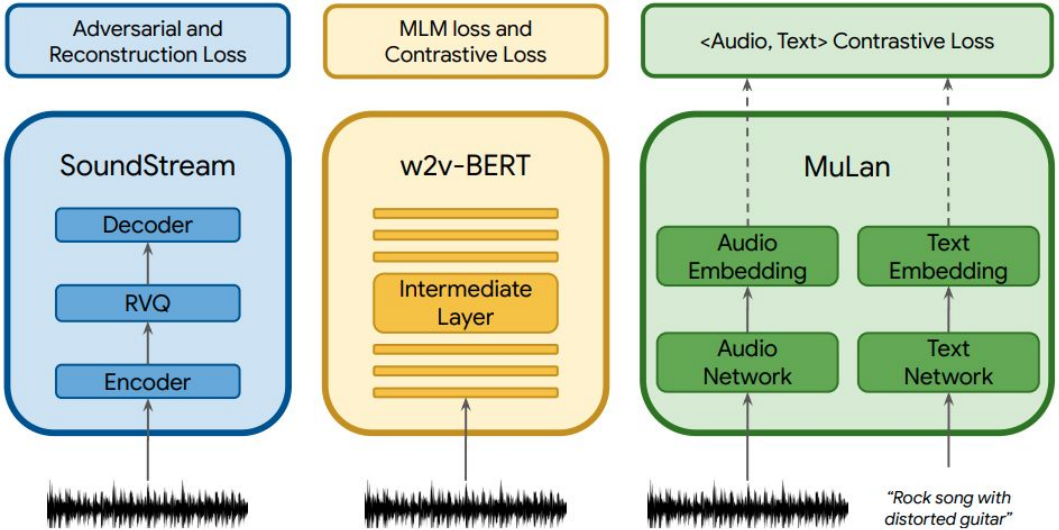
State-of-the-art: **Moûsai**



"***Electro House*** *(Remix), 2023, 3 of 4:"*

https://arxiv.org/pdf/2301.11757.pdf

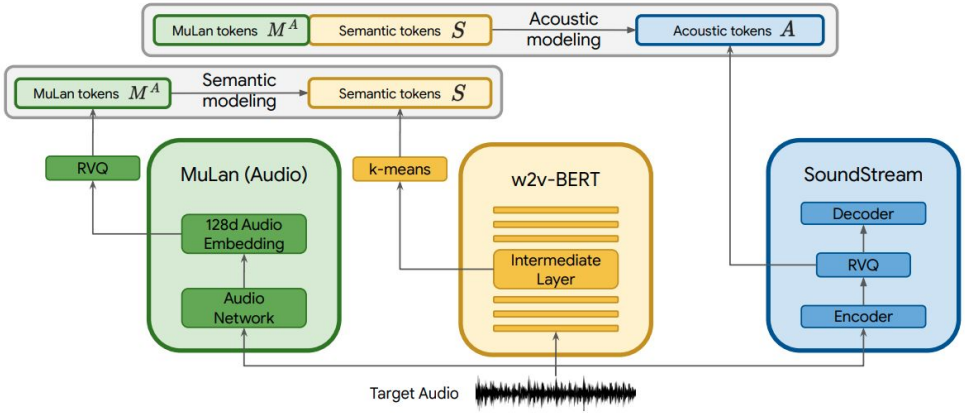
State-of-the-art: Google **MusicLM**



https://arxiv.org/pdf/2301.11325.pdf

State-of-the-art: Google **MusicLM**

Seq2Seq framework

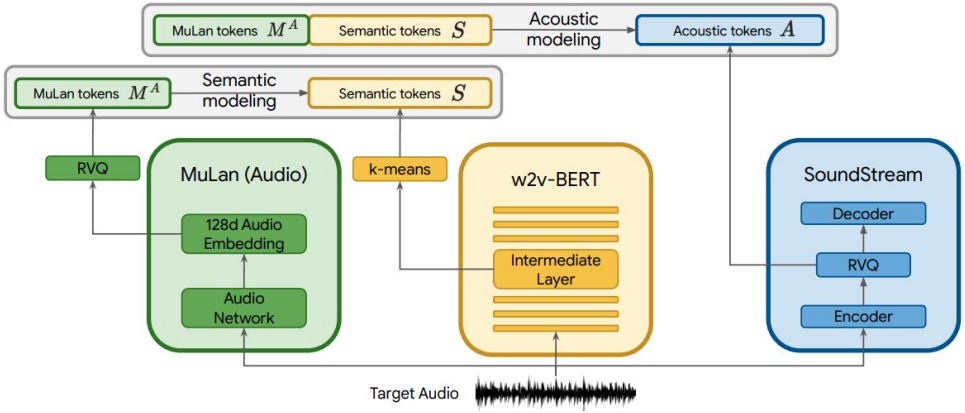


https://arxiv.org/pdf/2301.11325.pdf

State-of-the-art: Google **MusicLM**

Seq2Seq framework

Similar to **Jukebox**



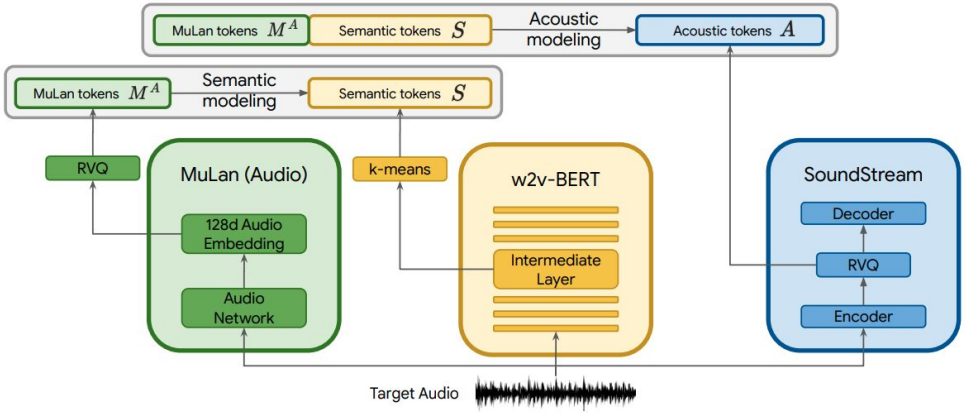
https://arxiv.org/pdf/2301.11325.pdf

State-of-the-art: Google **MusicLM**

Seq2Seq framework

Similar to **Jukebox**

Text/Melody **Conditioning**



https://arxiv.org/pdf/2301.11325.pdf

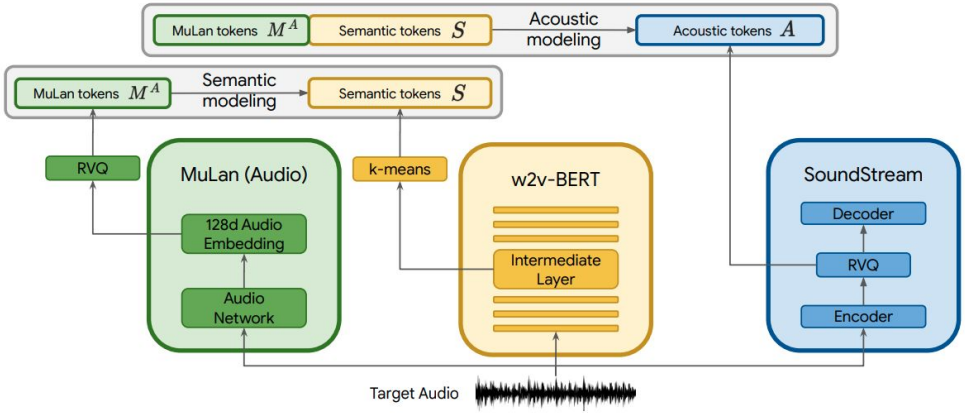
State-of-the-art: Google **MusicLM**

Seq2Seq framework

Similar to **Jukebox**

Text/Melody **Conditioning**

No need for **music-text** pairs!



https://arxiv.org/pdf/2301.11325.pdf

State-of-the-art: Google **MusicLM**

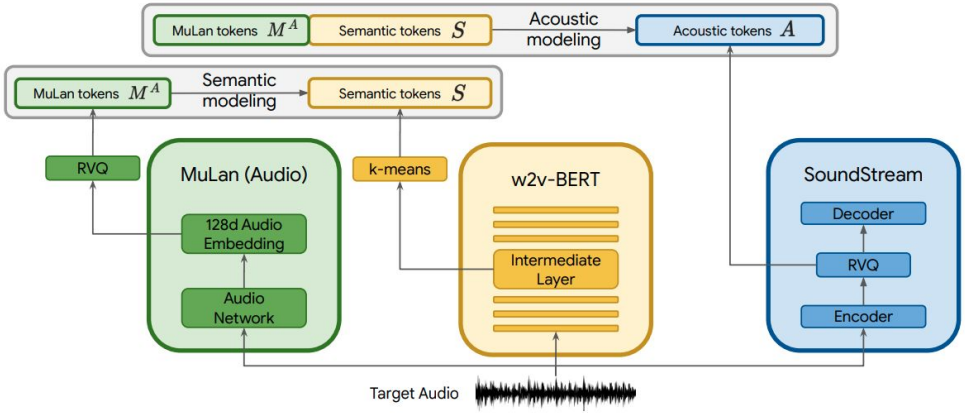
Seq2Seq framework

Similar to **Jukebox**

Text/Melody **Conditioning**

No need for **music-text** pairs!

24 kHz **mono** music



https://arxiv.org/pdf/2301.11325.pdf

State-of-the-art: Google **MusicLM**

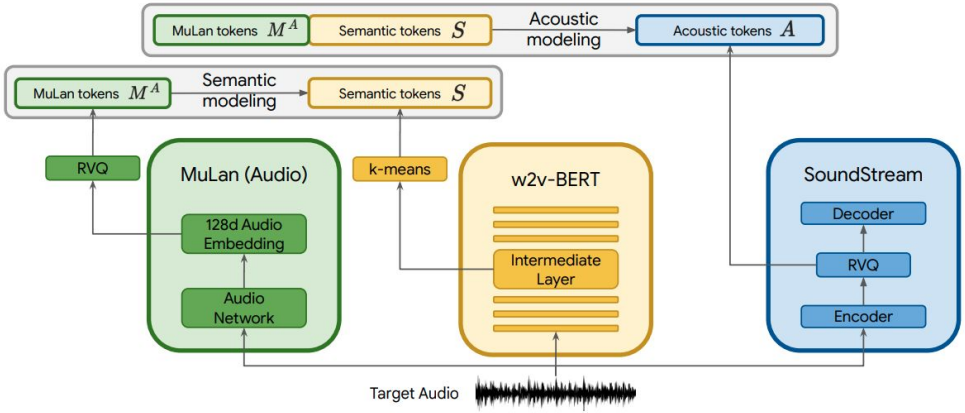
Seq2Seq framework

Similar to **Jukebox**

Text/Melody **Conditioning**

No need for **music-text** pairs!

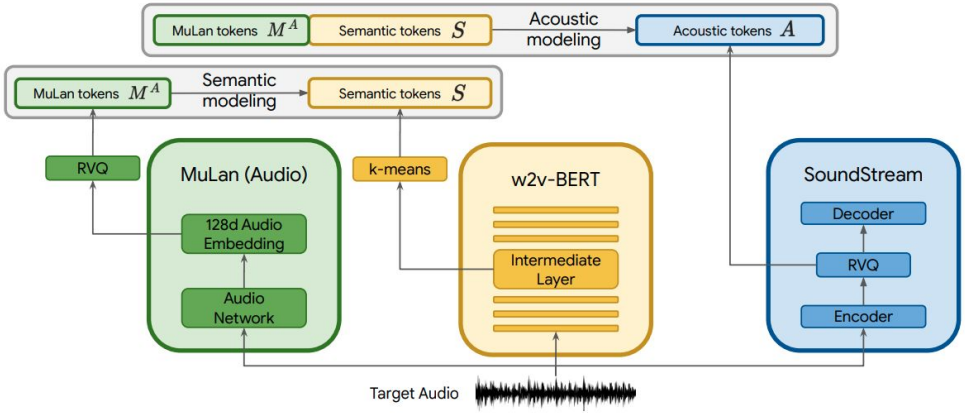
24 kHz **mono** music



Slow, Model/Data **not** publicly **released**

https://arxiv.org/pdf/2301.11325.pdf

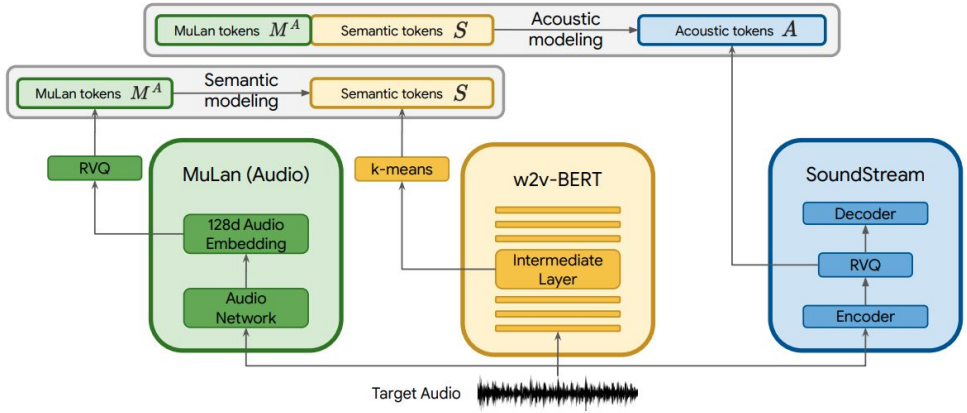
State-of-the-art: Google **MusicLM**



*"**This is an r&b/hip-hop music piece.** There is a male vocal rapping and a female vocal singing in a rap-like manner. The beat is comprised of a piano playing the chords of the tune with an electronic drum backing. The atmosphere of the piece is playful and energetic. This piece could be used in the soundtrack of a high school drama movie/TV show. It could also be played at birthday parties or beach parties."*

https://arxiv.org/pdf/2301.11325.pdf

State-of-the-art: Google **MusicLM**



*"**This is an r&b/hip-hop music piece.** There is a male vocal rapping and a female vocal singing in a rap-like manner. The beat is comprised of a piano playing the chords of the tune with an electronic drum backing. The atmosphere of the piece is playful and energetic. This piece could be used in the soundtrack of a high school drama movie/TV show. It could also be played at birthday parties or beach parties."*

https://arxiv.org/pdf/2301.11325.pdf

Our contribution

Our contribution

SOTA models:

100s GPUs, ∞$

Our contribution

SOTA models:

100s GPUs, ∞$

Us:

1 Gaming GPU, **0**$

Our contribution

SOTA models:

100s GPUs, ∞$

Us:

1 Gaming GPU, **0**$

How/What can we contribute?

Our contribution:

Our contribution:

Generate stereo waveform **music** of **arbitrary length**

Our contribution:

Generate stereo waveform **music** of **arbitrary length**

Stylistically coherent through time

Our contribution:

Generate stereo waveform **music** of **arbitrary length**

Stylistically coherent through time

Can be conditioned on signals (tempo, note density)

Our contribution:

MUSIKA!

Generate stereo waveform **music** of **arbitrary length**

Stylistically coherent through time

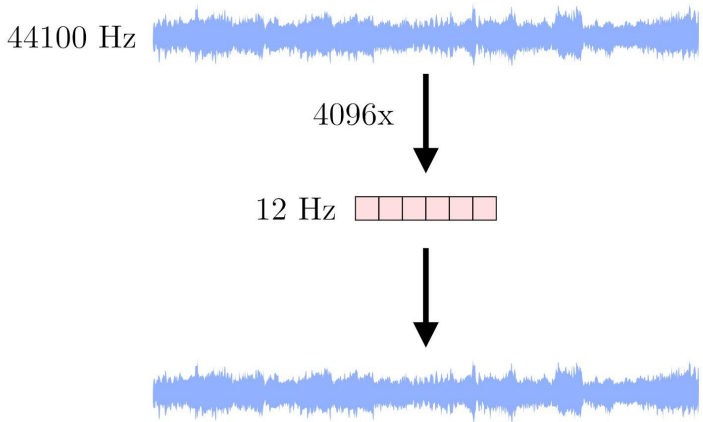
Can be conditioned on signals (tempo, note density)

Fast! Much **faster than real-time** on CPU

How **Musika** works

How **Musika** works

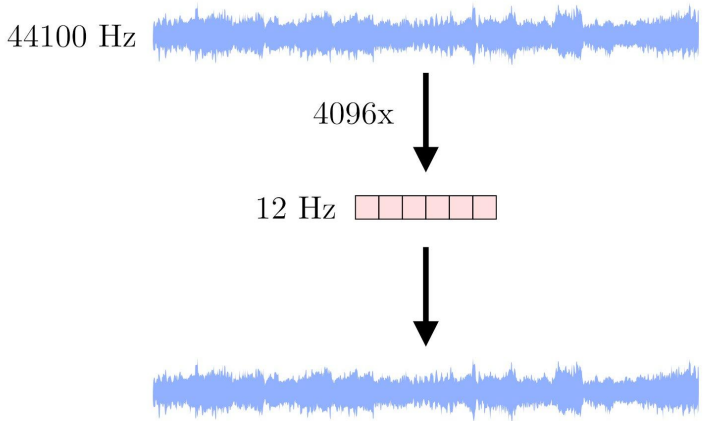
Encode audio to compressed **invertible representations**



How **Musika** works

Encode audio to compressed **invertible representations**

Train **generative model** on latent representations

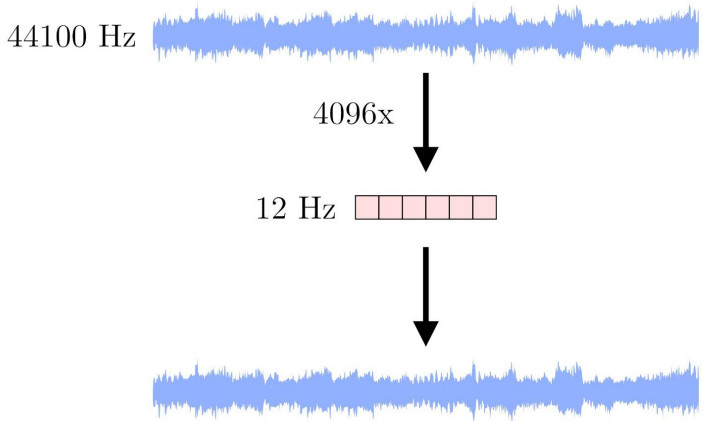


How **Musika** works

Encode audio to compressed **invertible representations**

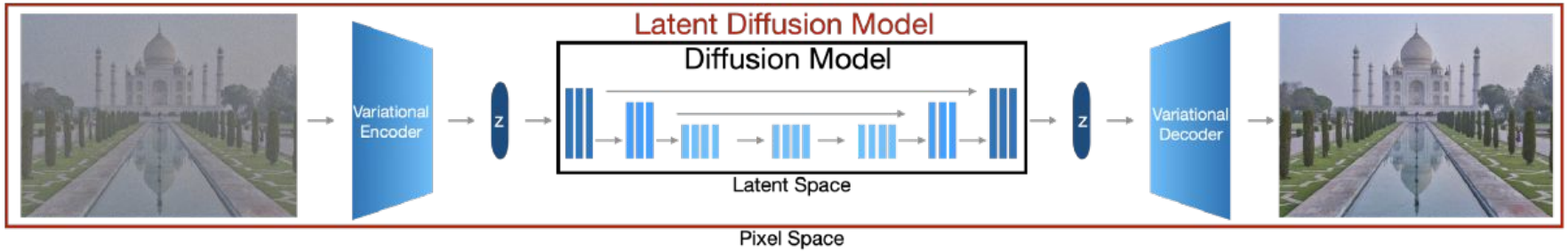
Train **generative model** on latent representations

Use **coordinate system** for **infinite generation**



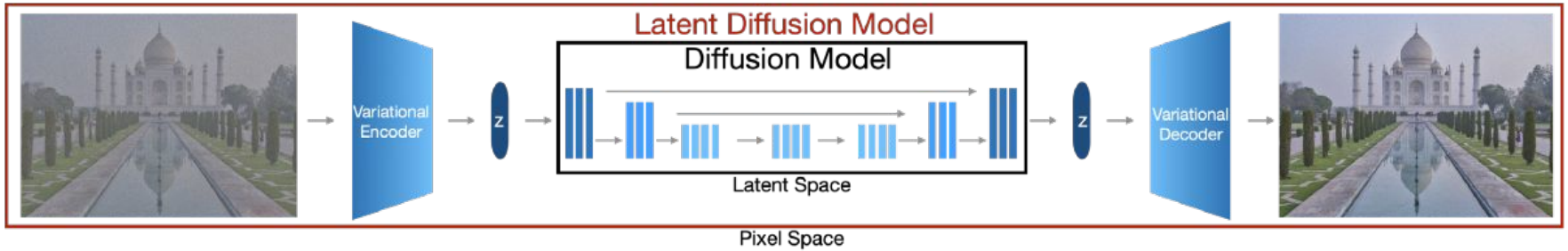
Latent **Generation**

Latent **Generation**



https://lih-verma.medium.com/latent-diffusion-model-68f6bdae1f05

Latent **Generation**



Recent common **approach** to **Fast**/Efficient **Generative Modeling**

https://lih-verma.medium.com/latent-diffusion-model-68f6bdae1f05

Audio **Autoencoder**

Audio **Autoencoder**

Priorities:

Audio **Autoencoder**

Priorities:

Inference **Speed**

Audio **Autoencoder**

Priorities:

Inference **Speed**

Parallel Encoding/Decoding

Audio **Autoencoder**

Priorities:

Inference **Speed**

Parallel Encoding/Decoding

Fast Training

Audio **Autoencoder**

Priorities:

- Inference **Speed**
- **Parallel** Encoding/Decoding
- **Fast** Training

Generating directly **Waveforms** (1D) is **expensive**

Audio **Autoencoder**

Priorities:

Inference **Speed**

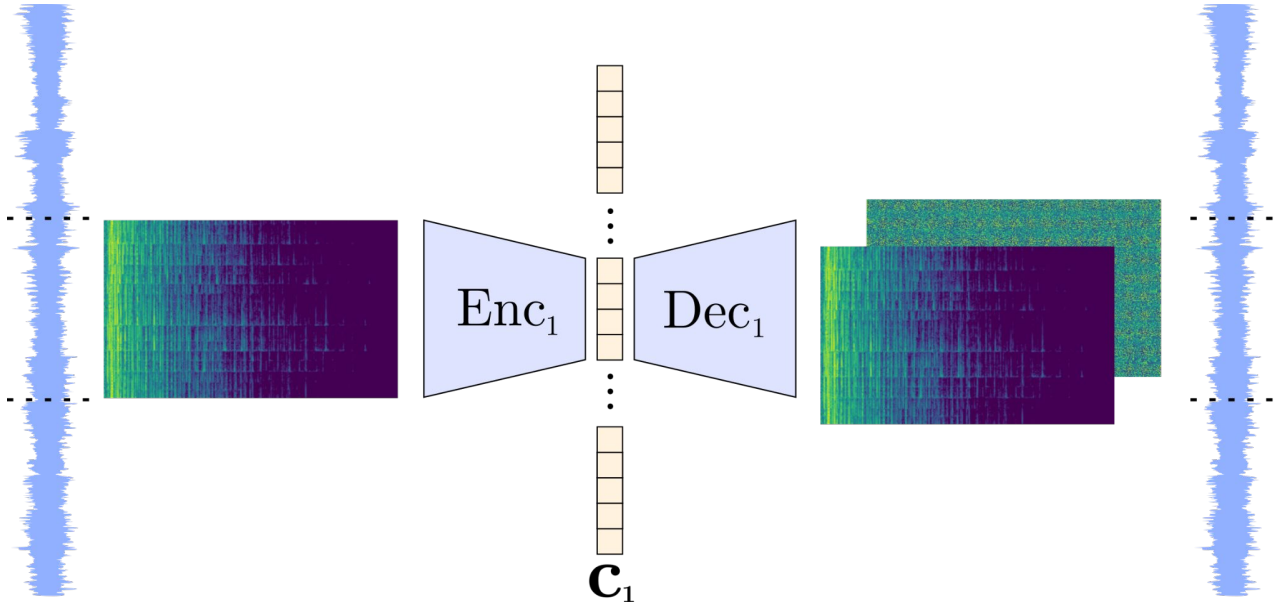
Parallel Encoding/Decoding

Fast Training

Generating directly **Waveforms** (1D) is **expensive**

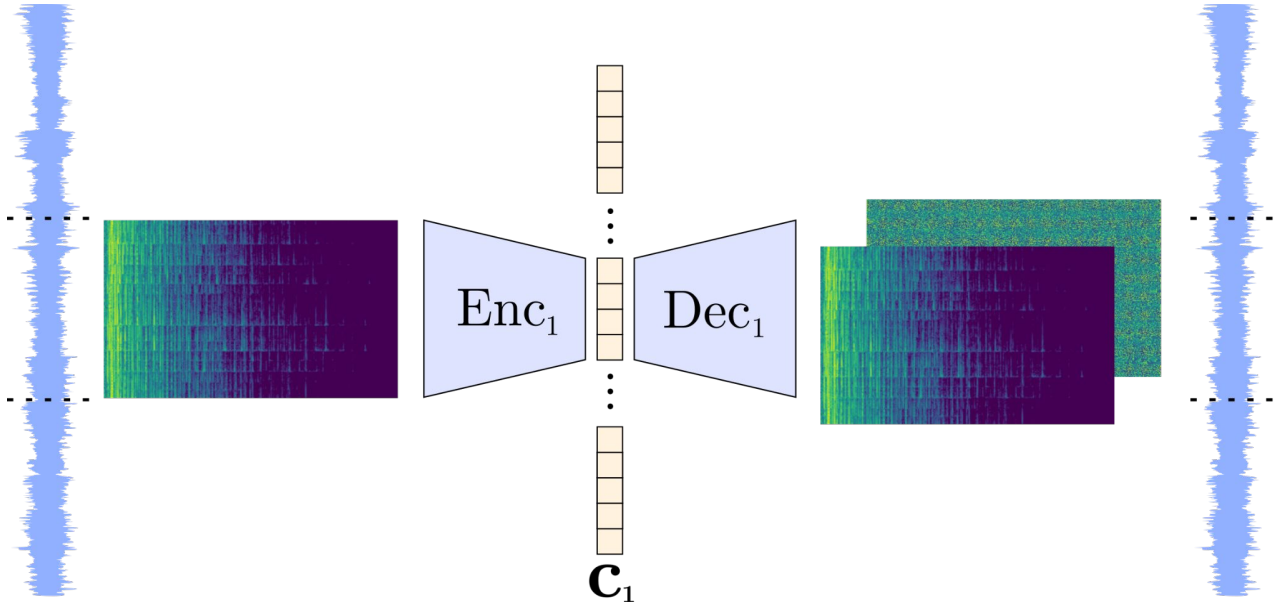
→ **Invertible 2D** Representation

Audio **Autoencoder**: **1st** Training Phase



Decoder outputs **magnitude+phase** spectrograms: **invertible!**

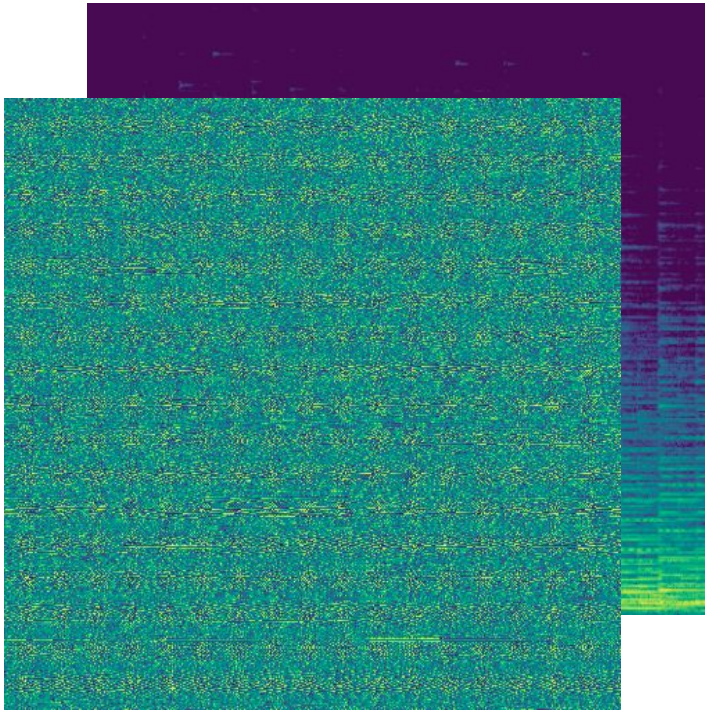
Audio **Autoencoder**: **1st** Training Phase



Only reconstruct **magnitude**, leave **phase** for later

Magnitude and **Phase**

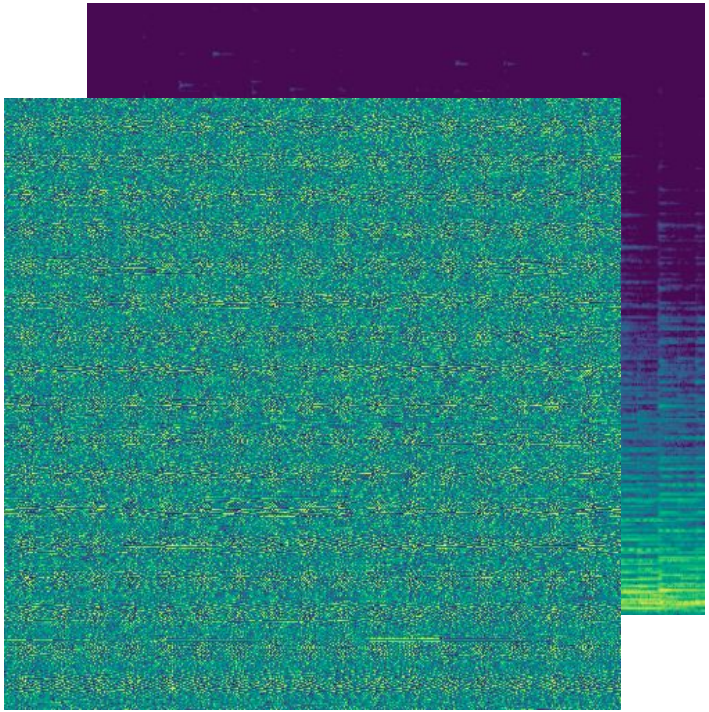
Modeling **Phase** is difficult!



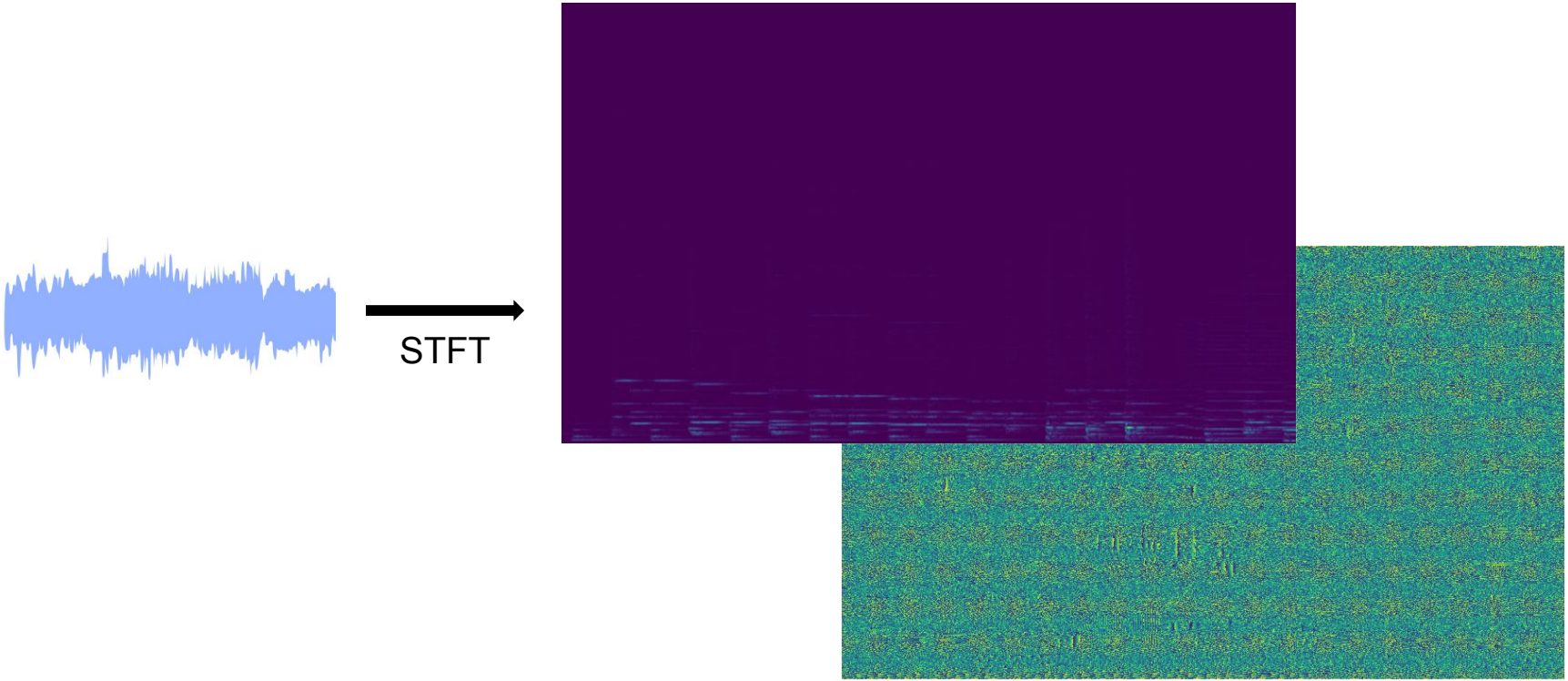
Magnitude and **Phase**

Modeling **Phase** is difficult!

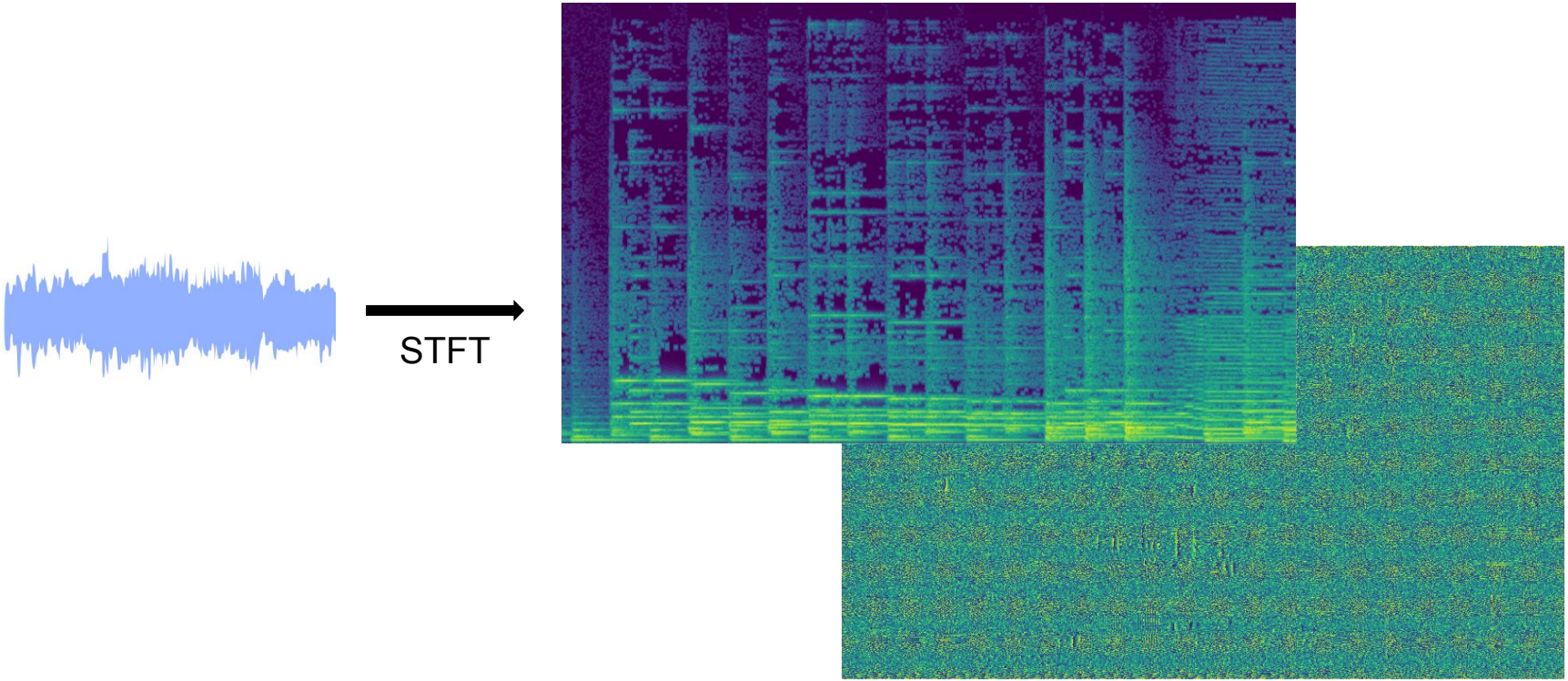
Lack of **Structure**



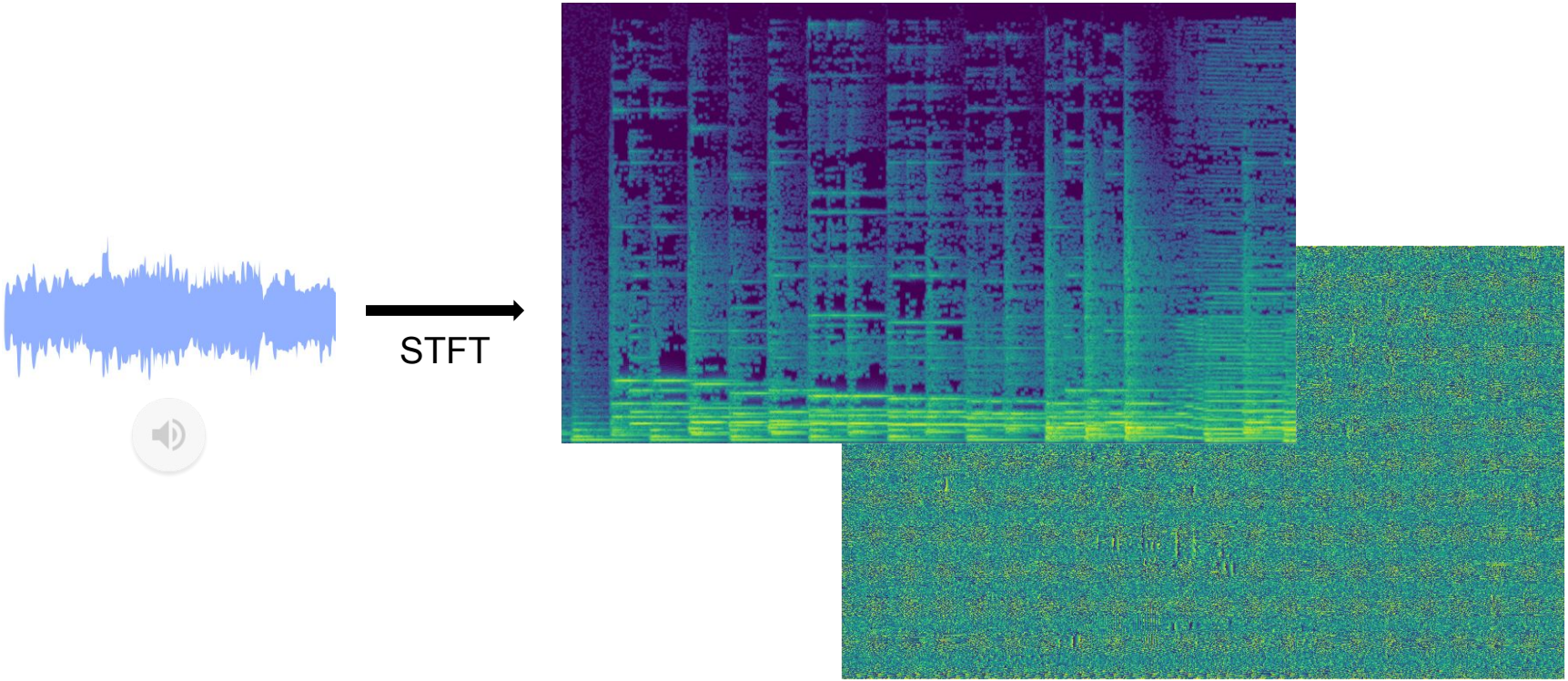
Magnitude and **Phase: STFT**



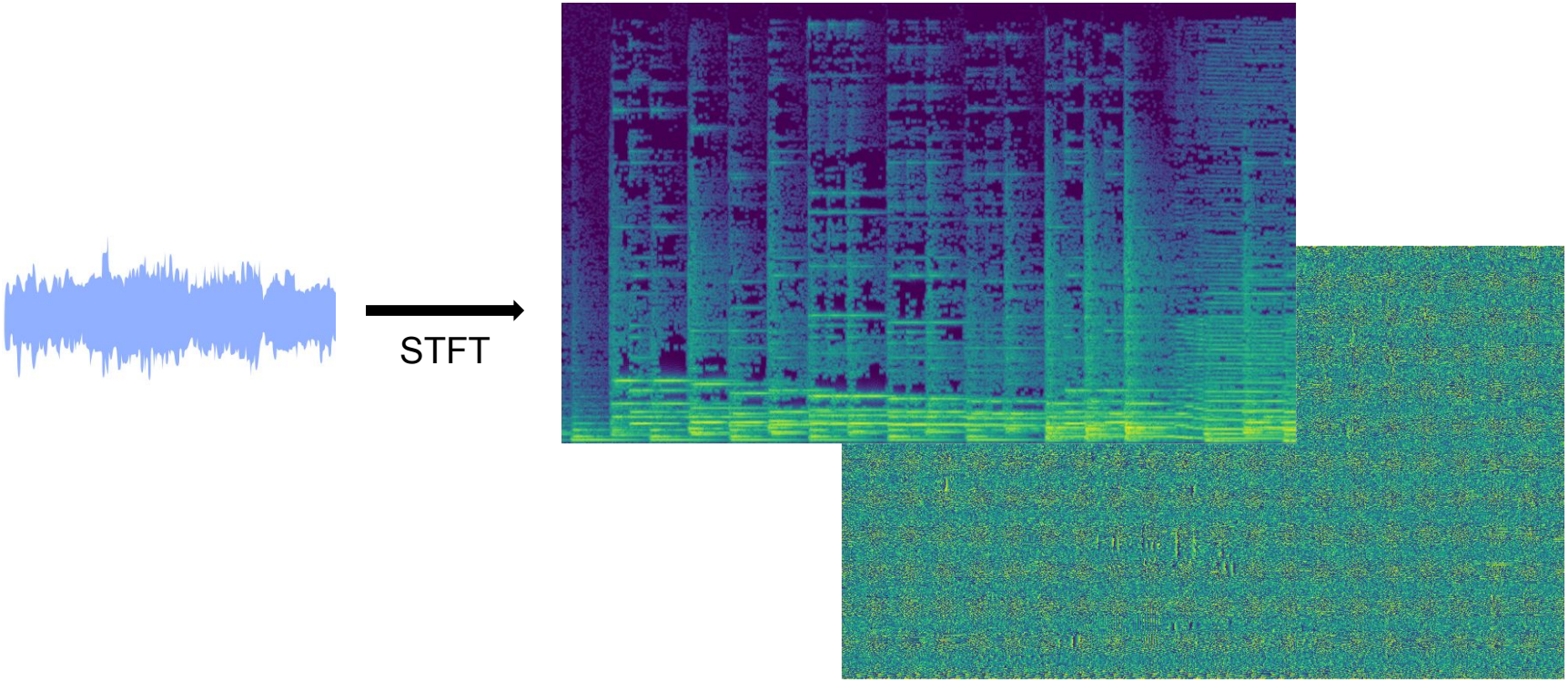
Magnitude and **Phase: STFT**



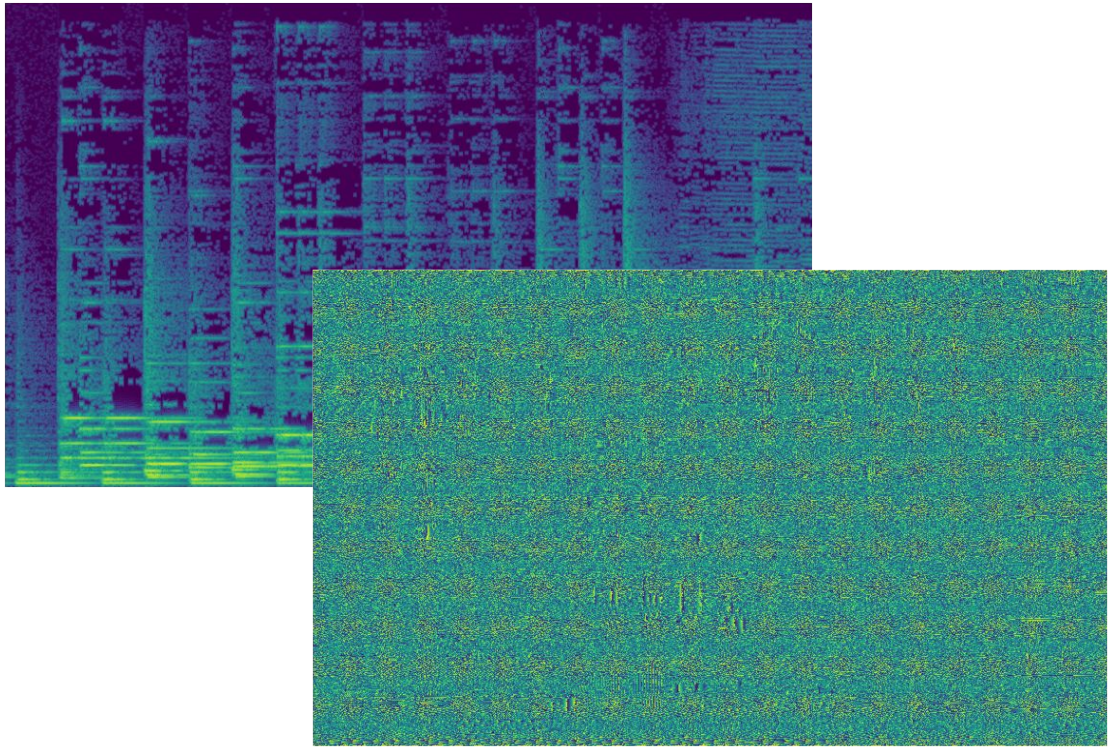
Magnitude and **Phase: STFT**



Magnitude and **Phase: STFT**

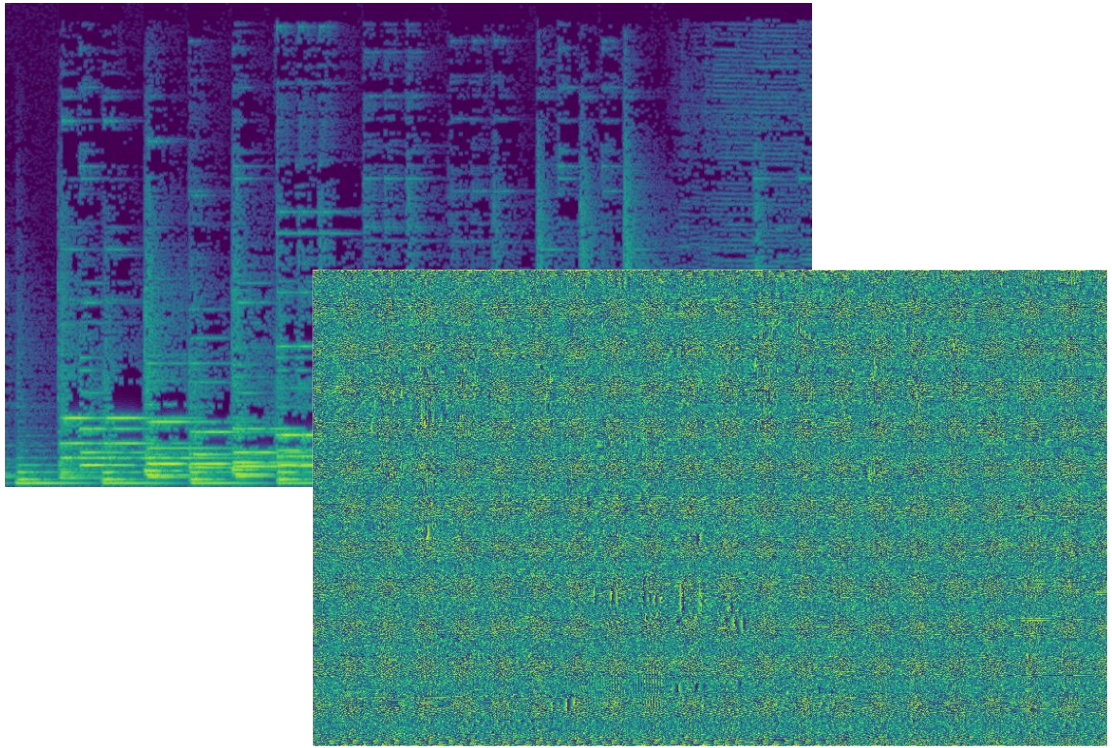


Magnitude and **Phase:** inverse**STFT**



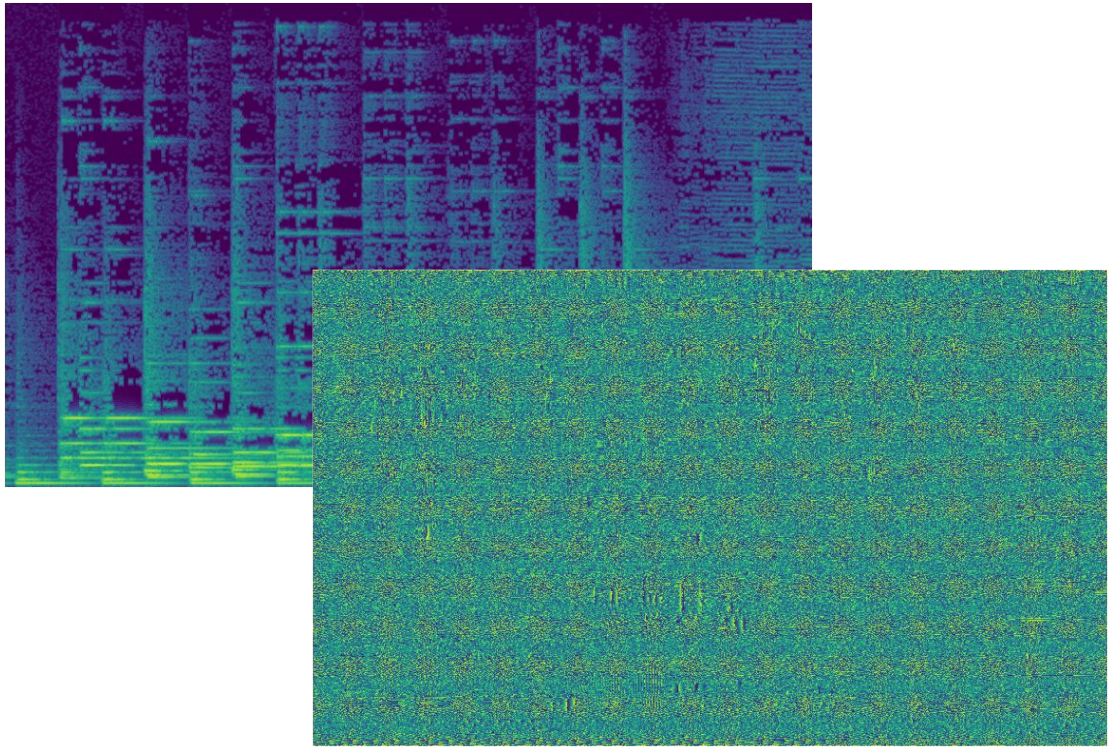
iSTFT

Magnitude and **Phase:** inverse**STFT**



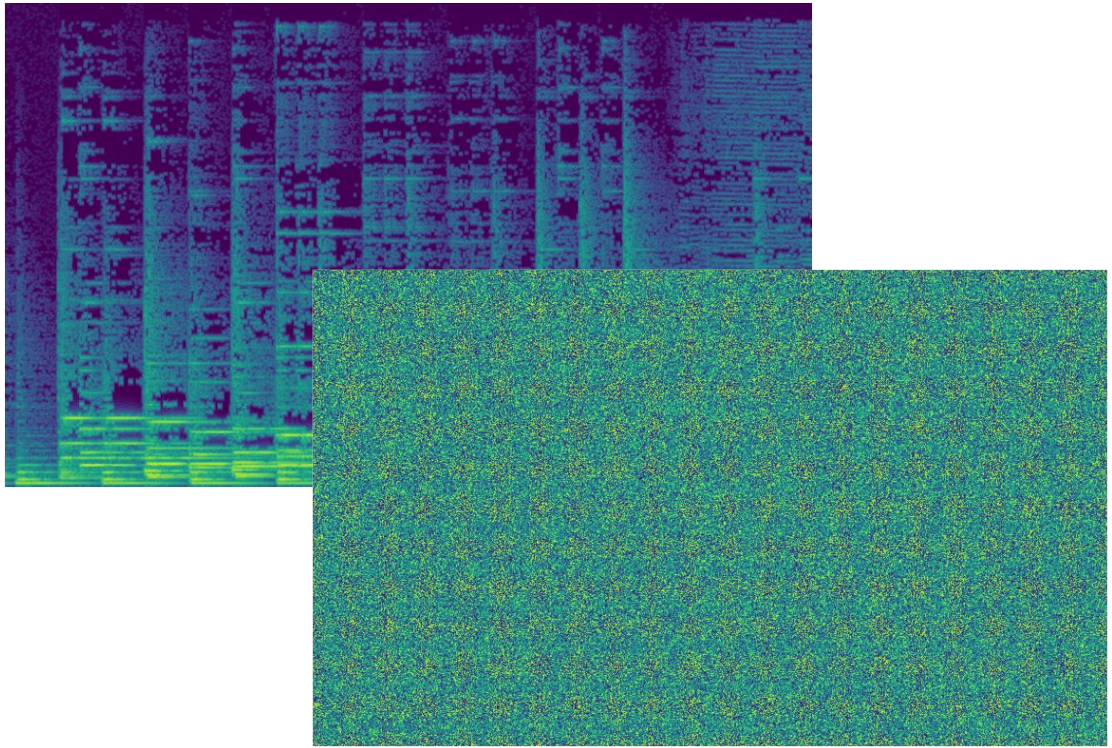
iSTFT

Magnitude and **Phase:** inverse**STFT**



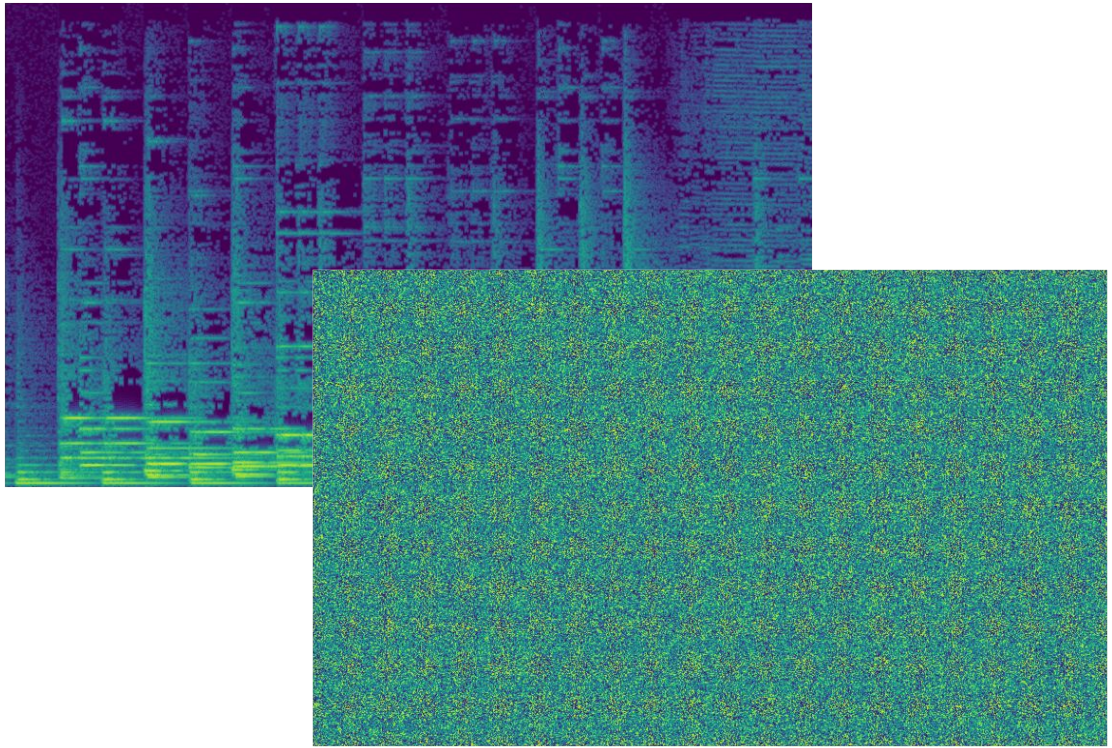
iSTFT

Magnitude and random **Phase:** inverse**STFT**

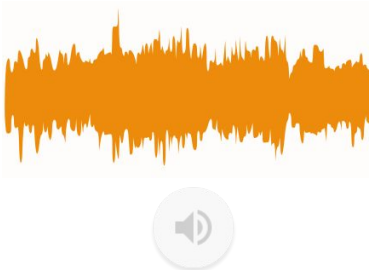


iSTFT

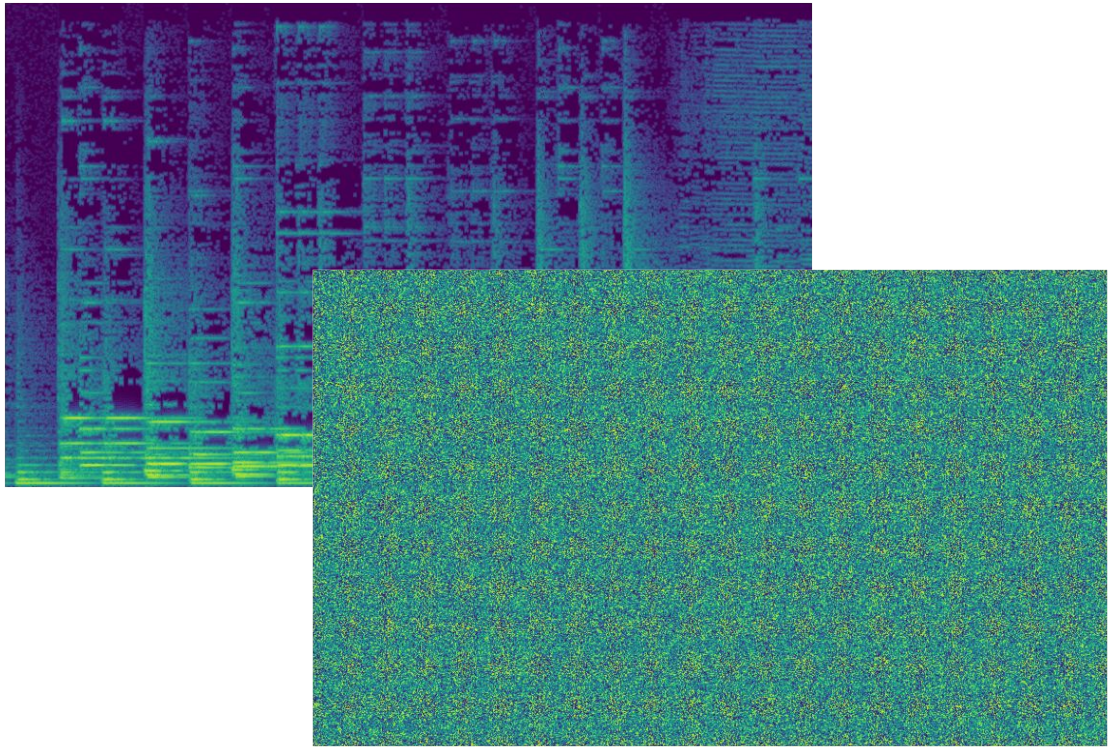
Magnitude and random **Phase:** inverse**STFT**



iSTFT

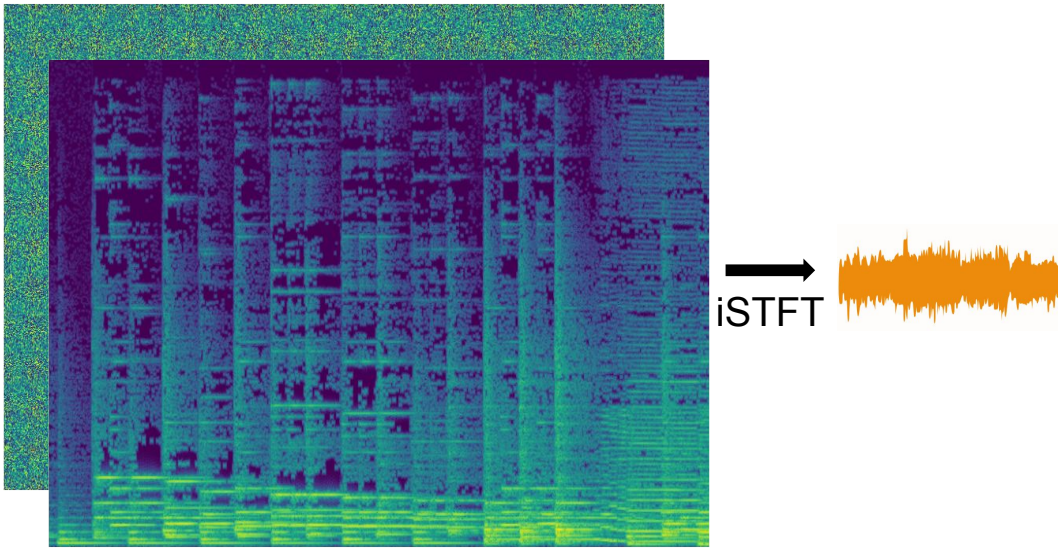


Magnitude and random **Phase:** inverse**STFT**

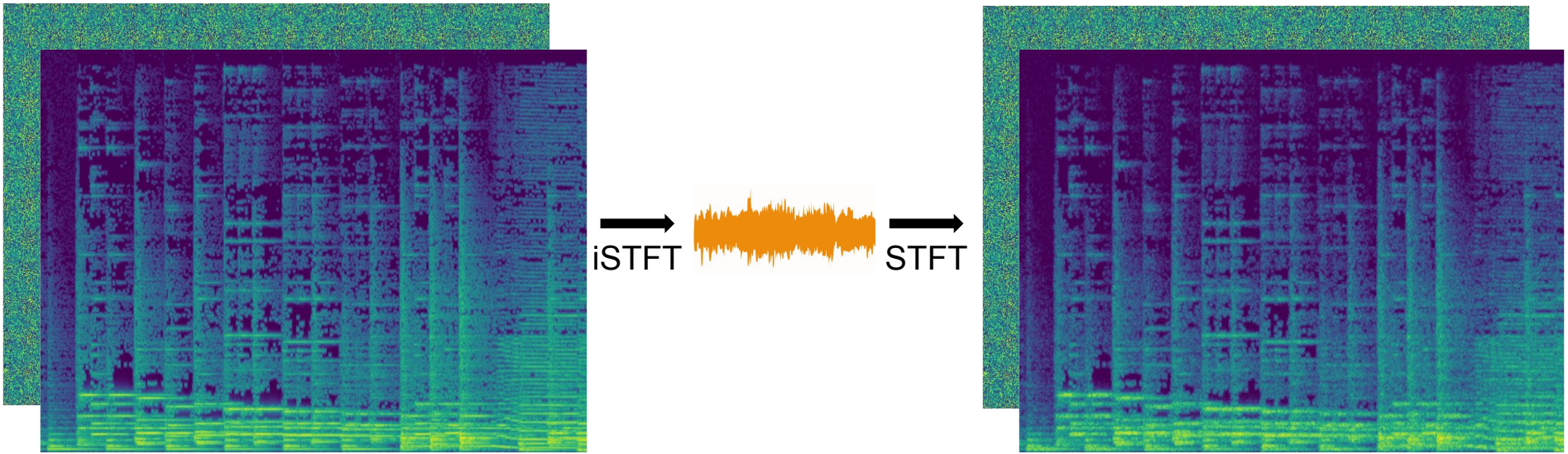


iSTFT

Magnitude and random **Phase:** inverse**STFT**



Magnitude and random **Phase:** inverse**STFT**

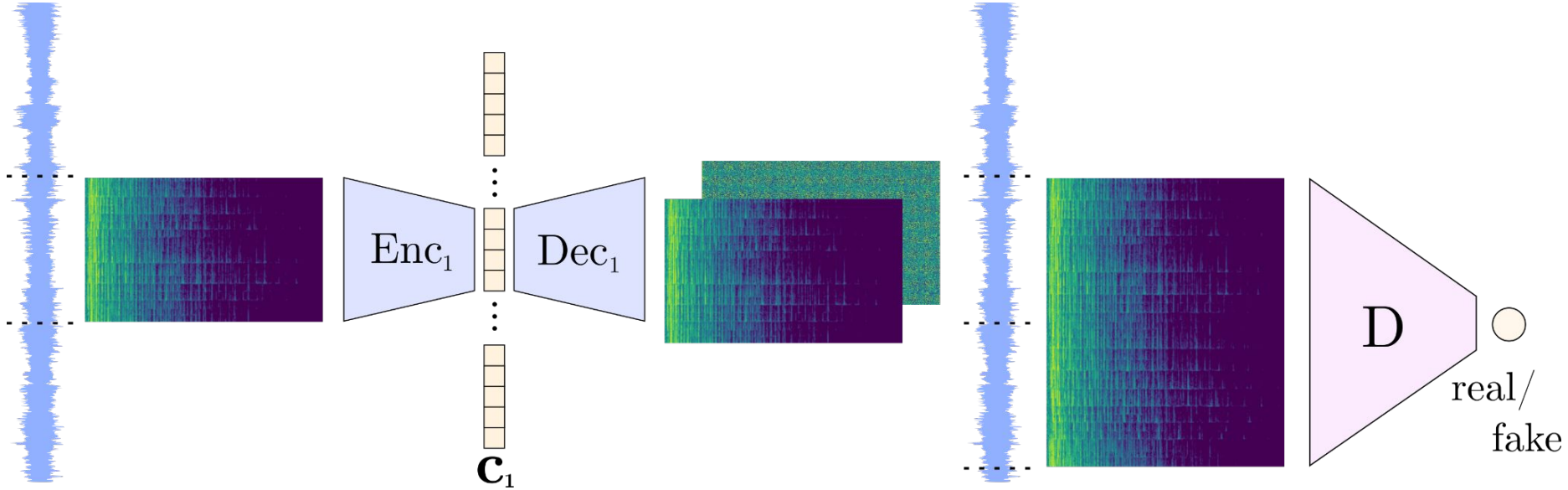


Magnitude and random **Phase:** inverse**STFT**

Magnitude and random **Phase:** inverse**STFT**

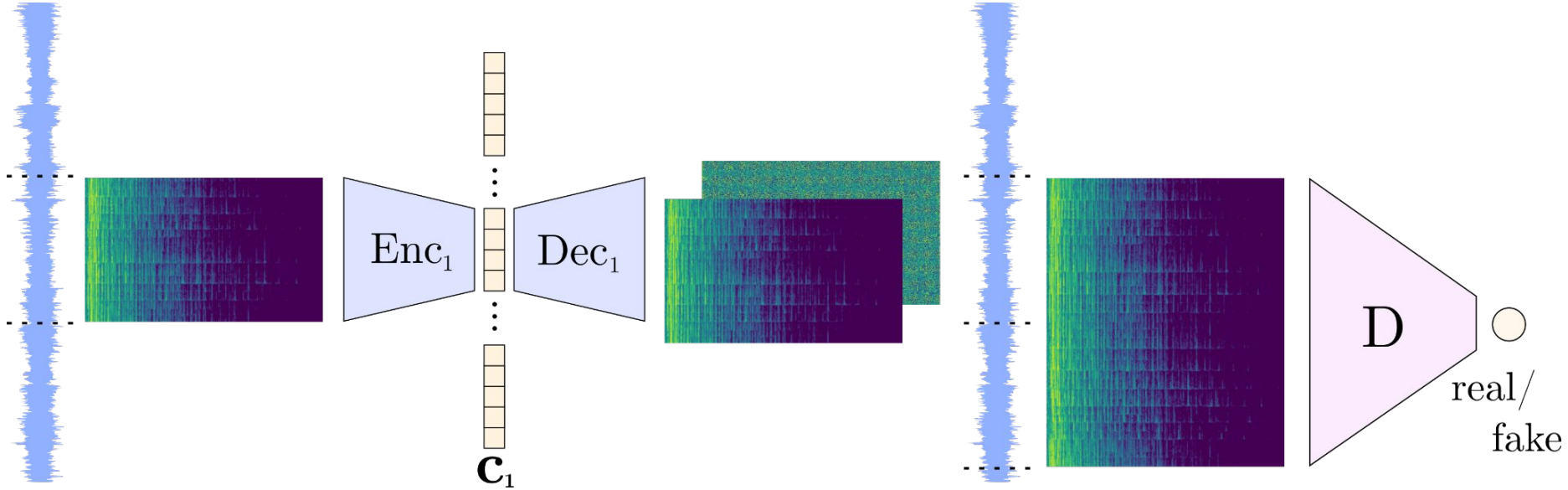
Use **Magnitude** instead of **Waveform** to model **Phase**!

Audio **Autoencoder**: **2nd** Training Phase



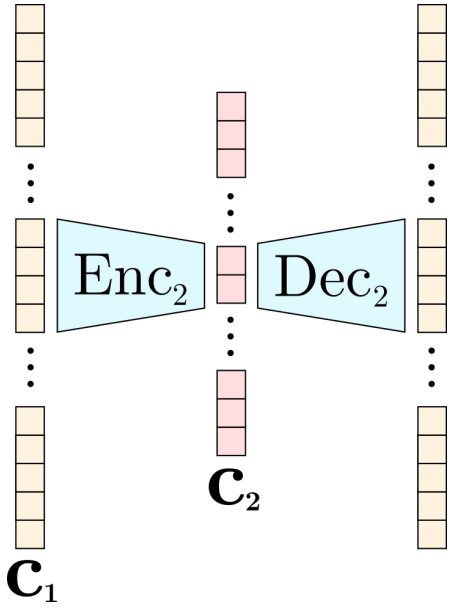
Adversarial objective models **phases** indirectly

Audio **Autoencoder**: **2nd** Training Phase



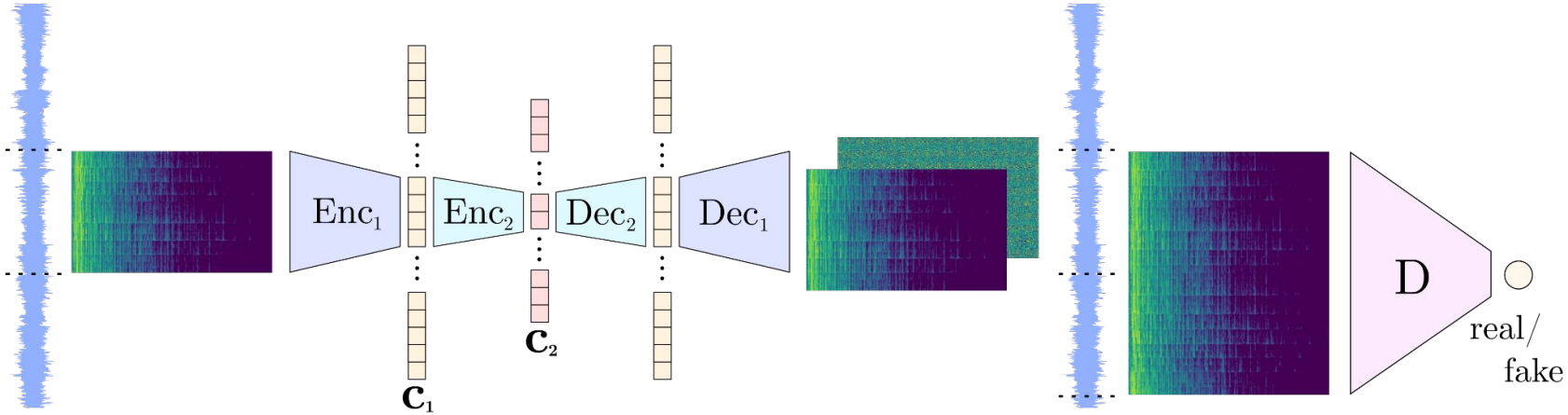
Adversarial objective removes **boundary artifacts**

Audio **Autoencoder**: **3rd** Training Phase



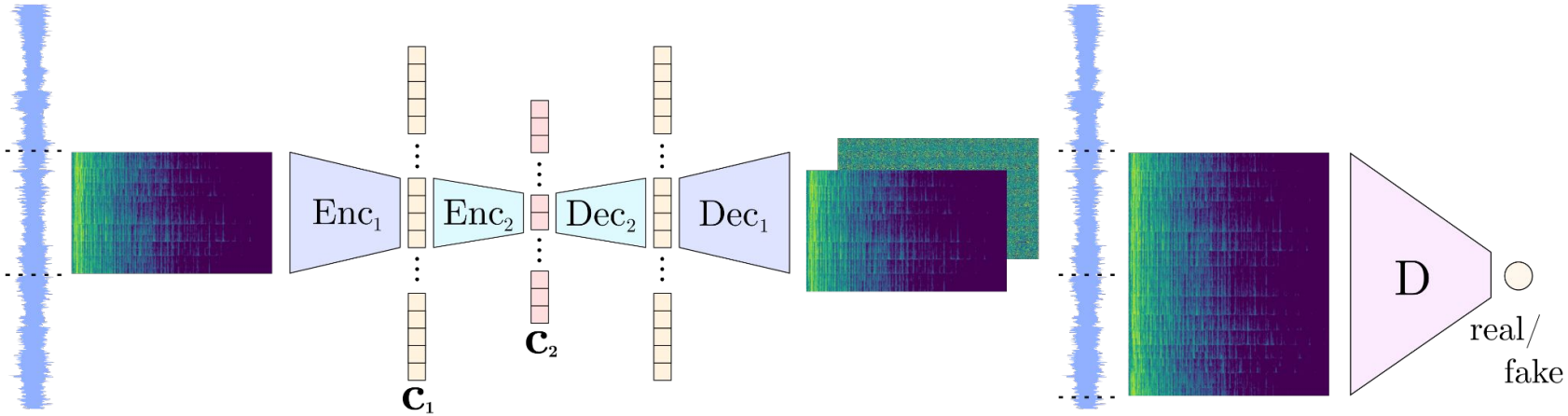
Reconstruct **representations** from **1st** level Autoencoder

Audio **Autoencoder**: **4th** Training Phase



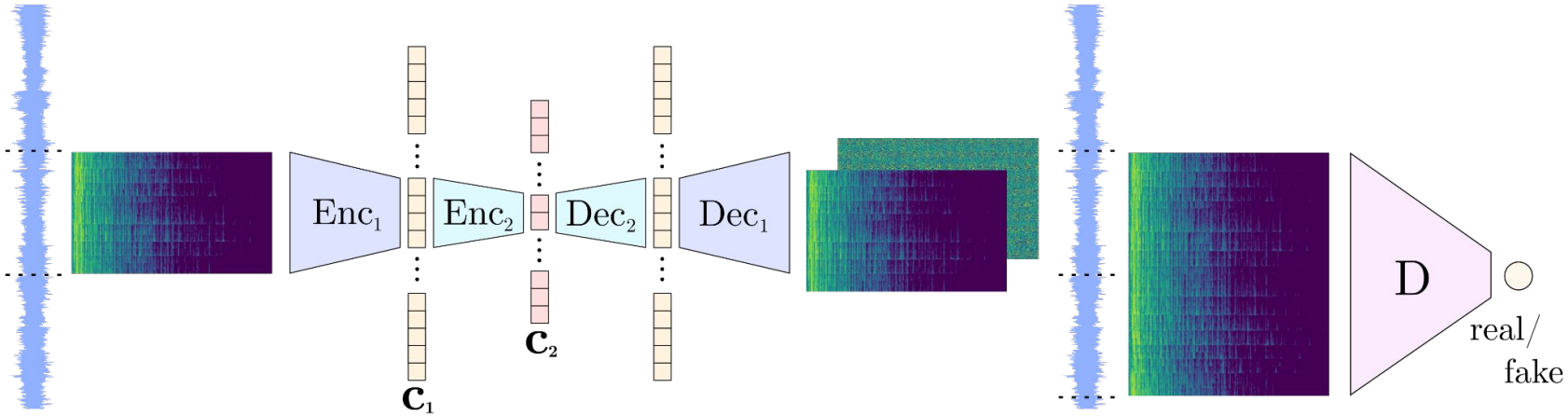
Gradients through **frozen 1st level** Autoencoder

Audio **Autoencoder**



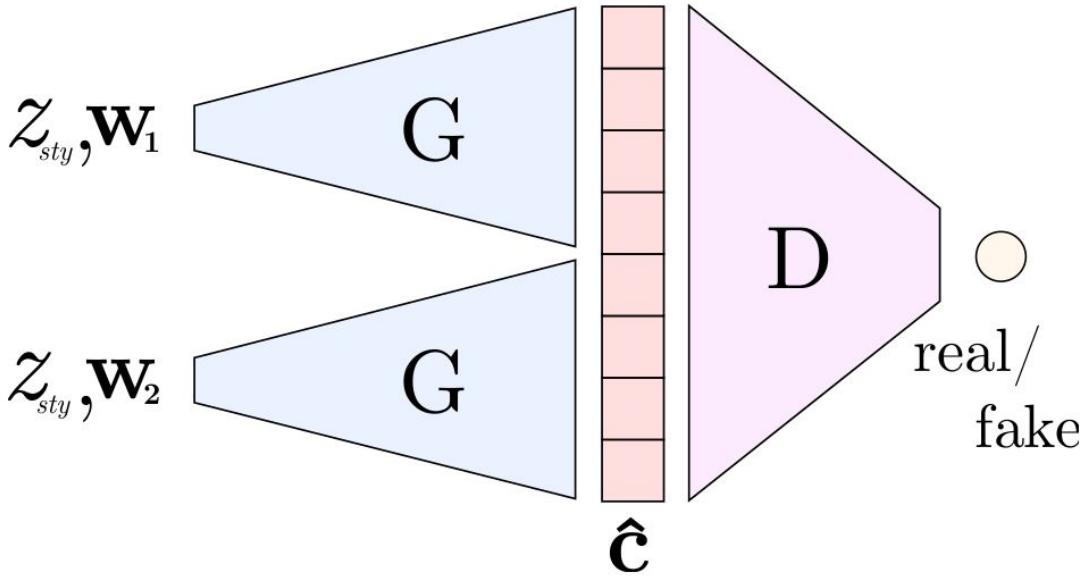
Why 2-stage design?

Audio **Autoencoder**



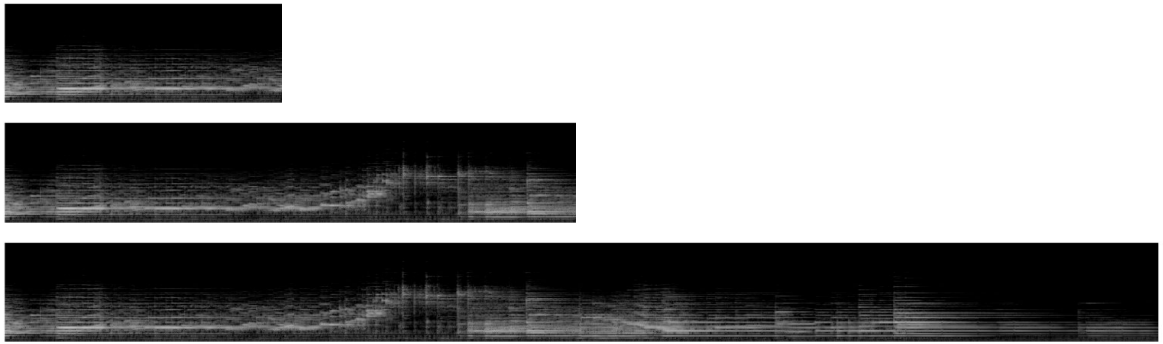
Why 2-stage design? Reconstruct **phase** + semantic **compression**

Latent **Generator**



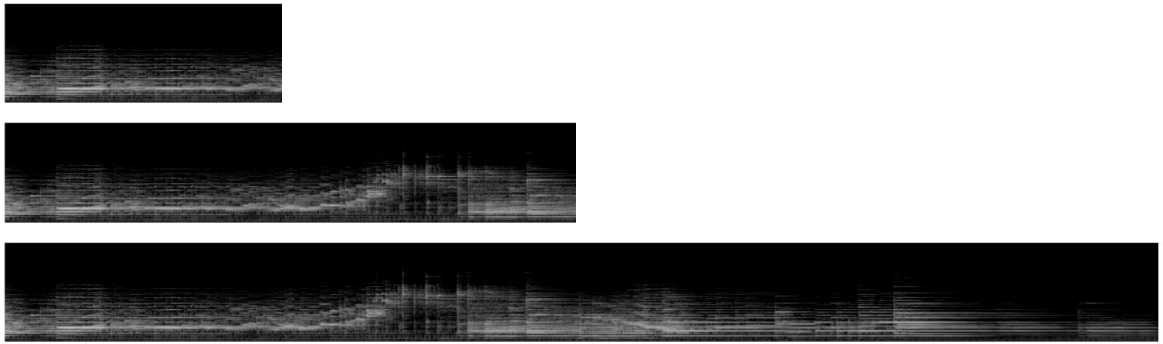
A **Latent GAN** generates **new** sequences of **latent vectors**

Arbitrary Length Generation



Arbitrary Length Generation

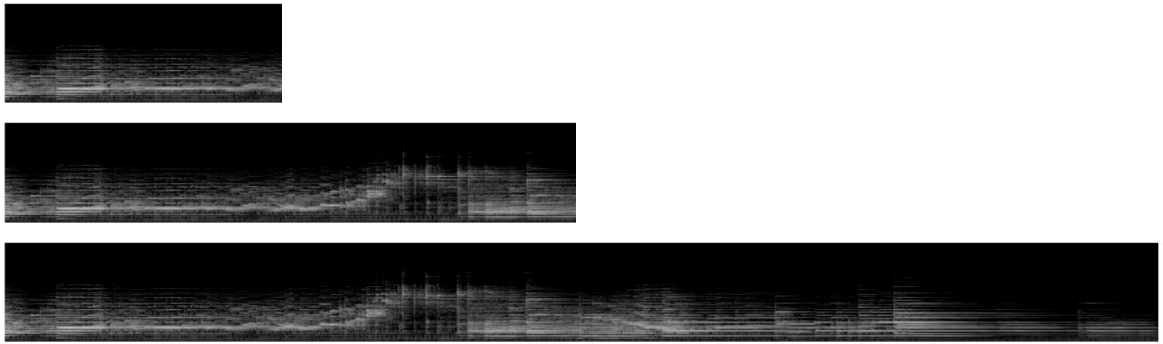
Autoregressive models (Jukebox): Trivial



Arbitrary Length Generation

Autoregressive models (Jukebox): Trivial

Non-Autoregressive models: not Trivial!

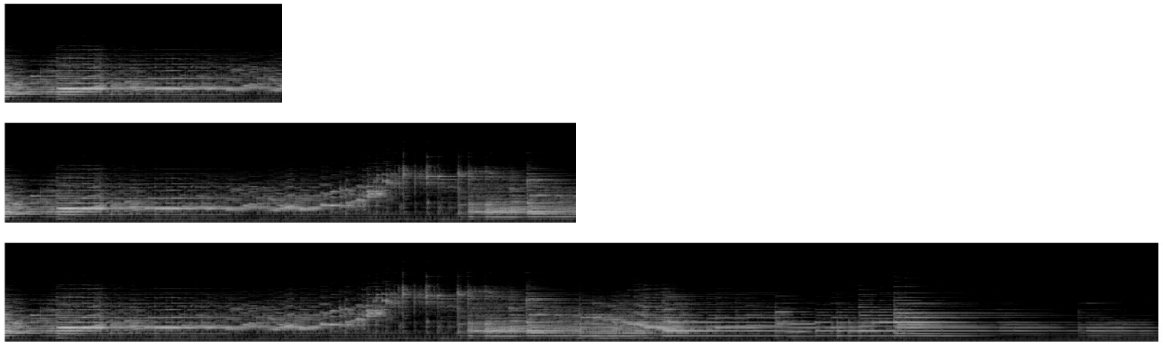


Arbitrary Length Generation

Autoregressive models (Jukebox): Trivial

Non-Autoregressive models: not Trivial!

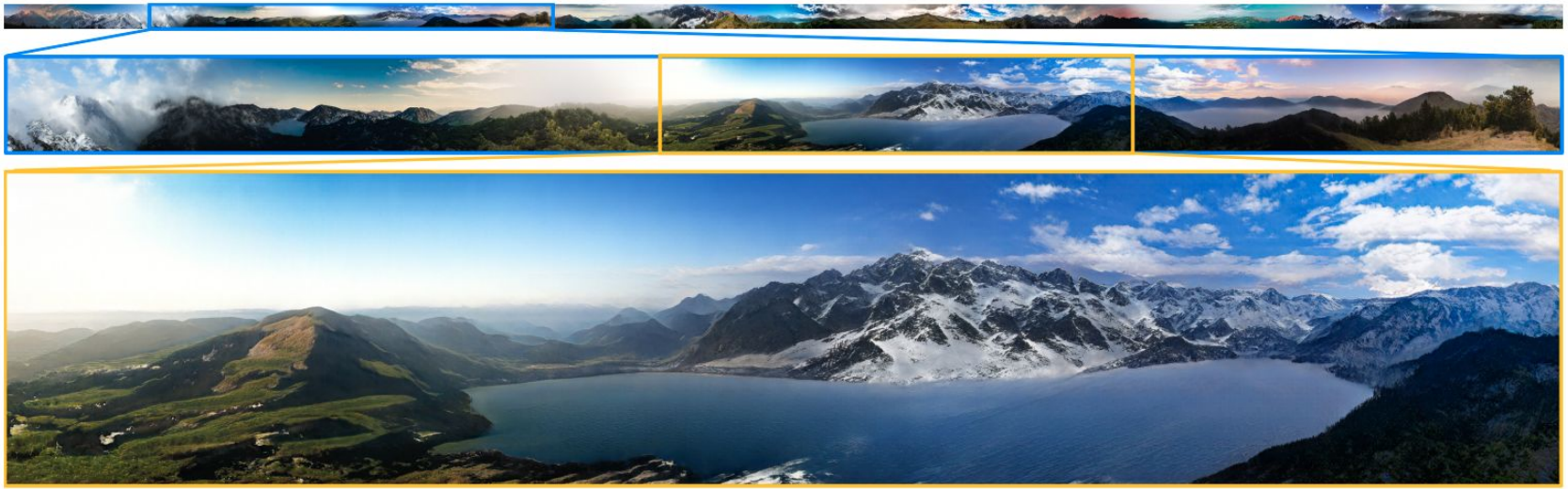
Ensure **coherent** parallely generated **samples**



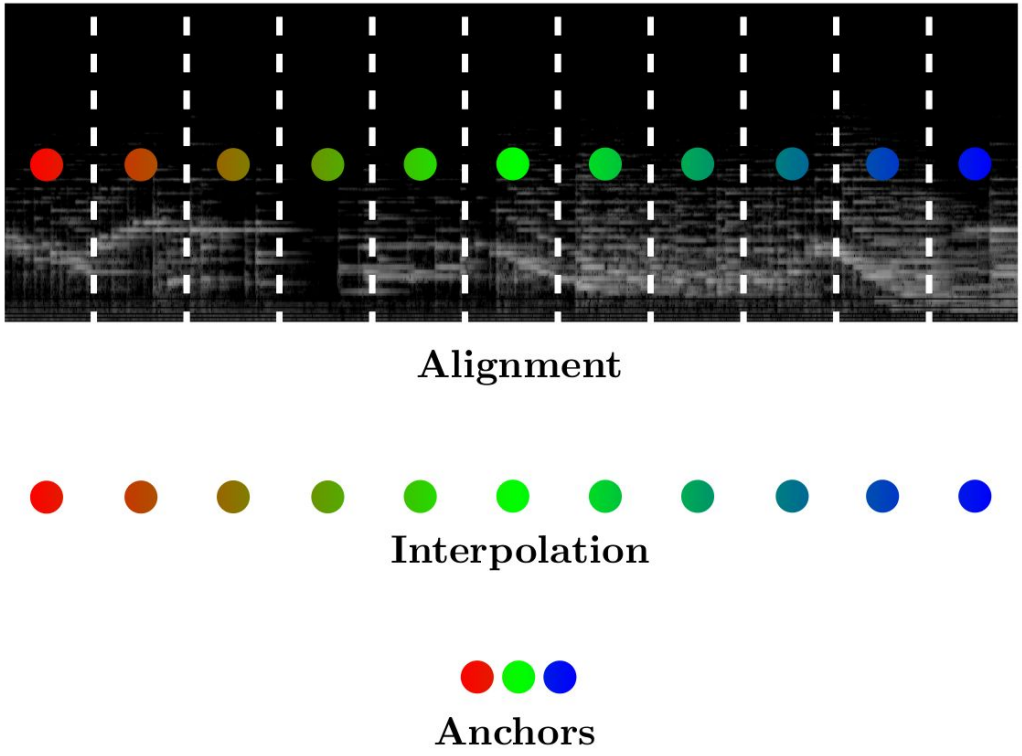
Aligning Latent and **Image Spaces**

to Connect the Unconnectable

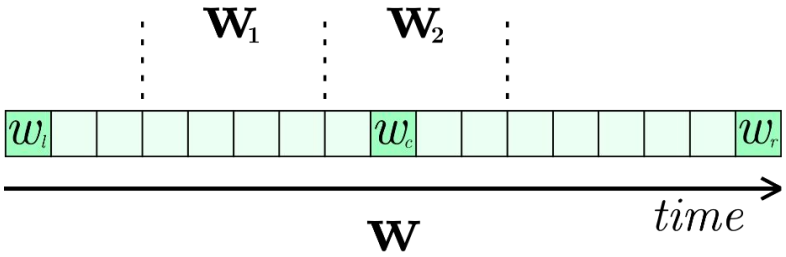
ICCV 2021

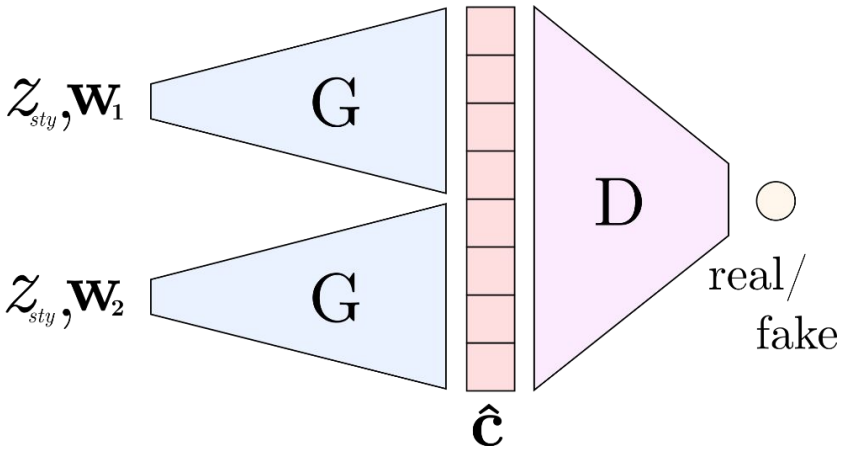


https://arxiv.org/pdf/2104.06954.pdf

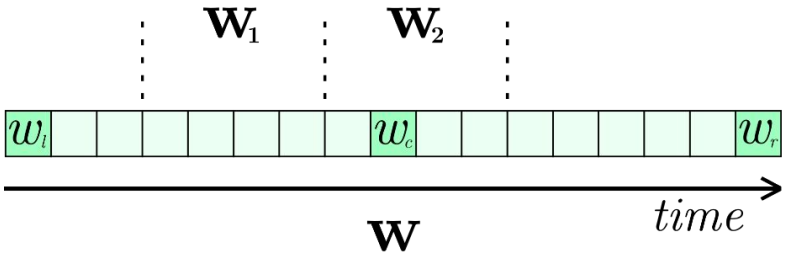


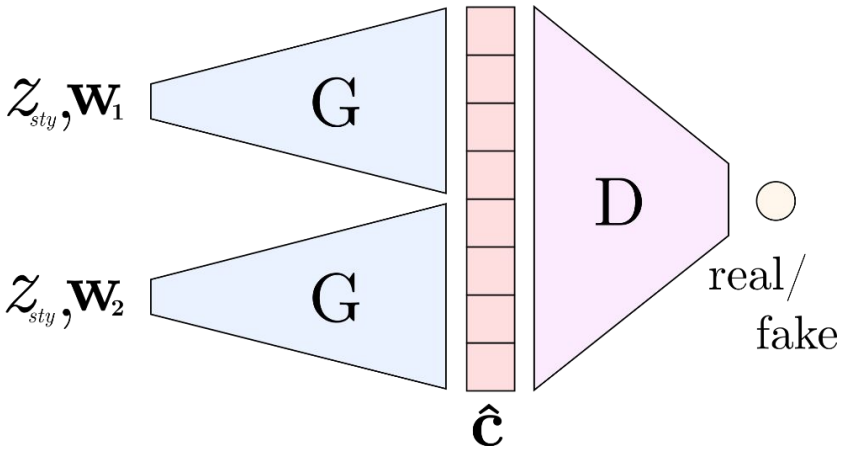
Training **aligns latent coordinates** with generated **sequences**





Training **aligns latent coordinates** with generated **sequences**



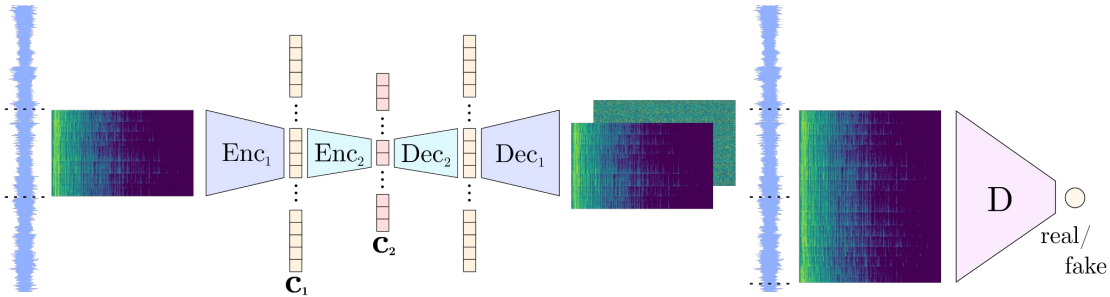


Single random vector is used as **conditioning** for **style**

Implementation

Autoencoder:

Fully convolutional, **1D** Convs

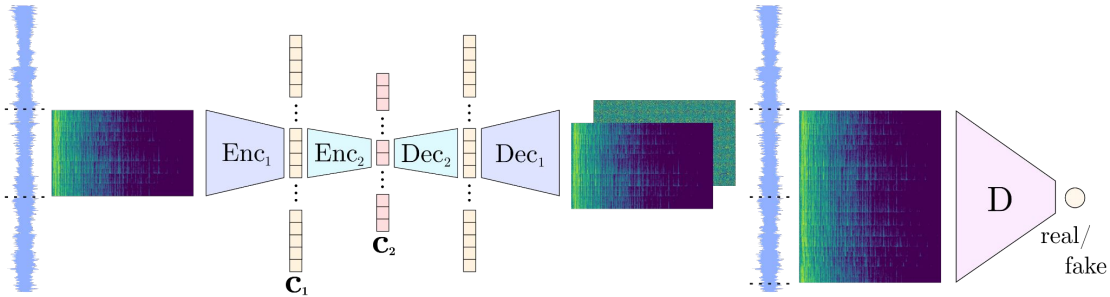


Implementation

Autoencoder:

Fully convolutional, **1D** Convs

High freq. resolution spectrograms



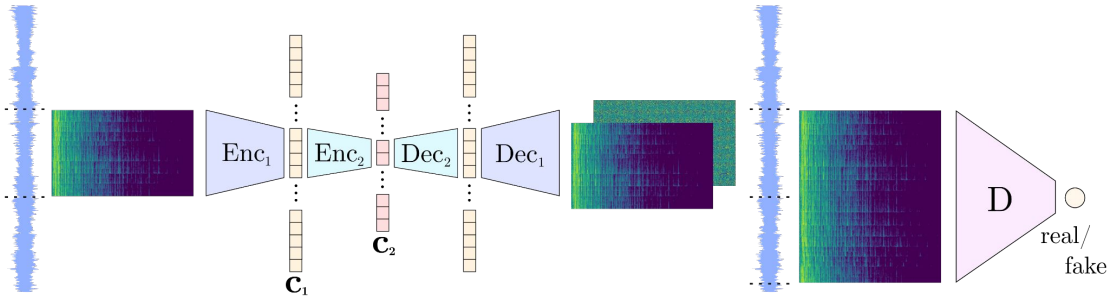
Implementation

Autoencoder:

Fully convolutional, **1D** Convs

High freq. resolution spectrograms

Critic with **2D** Convs



Implementation

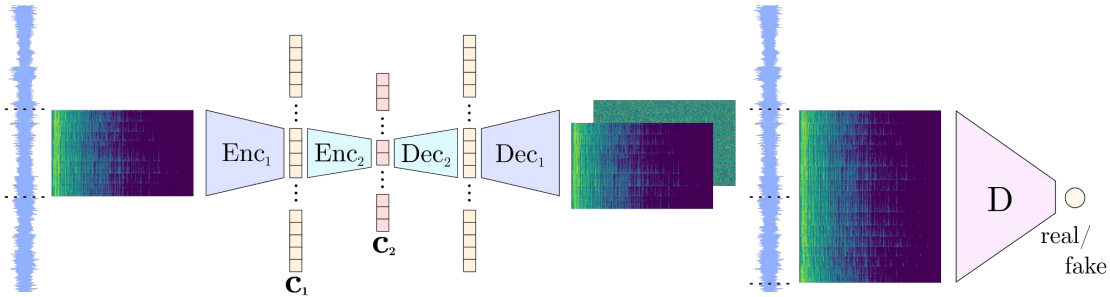
Autoencoder:

Fully convolutional, **1D** Convs

High freq. resolution spectrograms

Critic with **2D** Convs

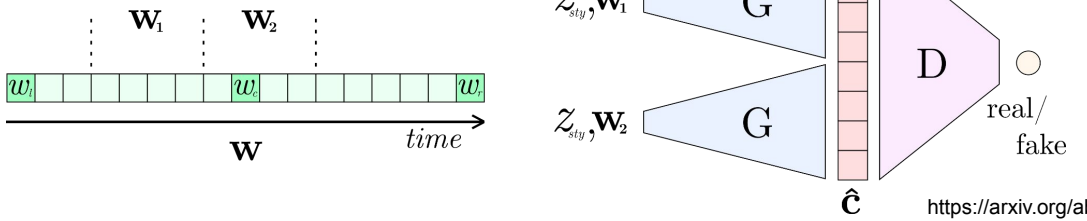
Multi-scale spectral distance to aid **reconstruction**



Implementation

Latent GAN:

FastGAN with **Residual 1D** Conv. Blocks



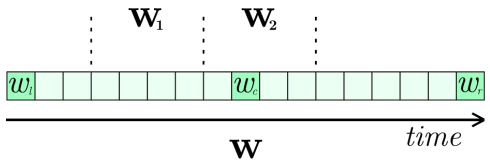
https://arxiv.org/abs/2111.01007 https://arxiv.org/abs/2101.04775

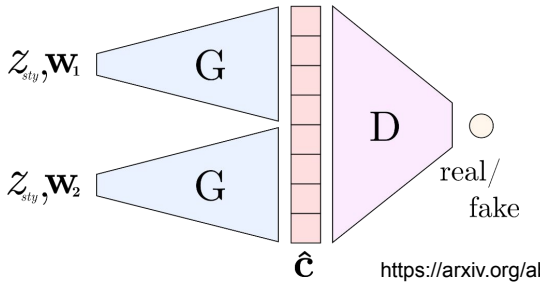
Implementation

Latent GAN:

FastGAN with **Residual 1D** Conv. Blocks

Spatial **AdaIN** with conditioning signal





https://arxiv.org/abs/2111.01007 https://arxiv.org/abs/2101.04775

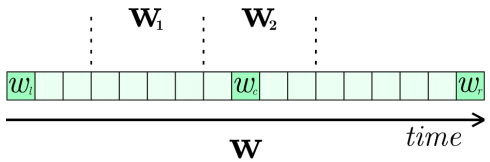
Implementation

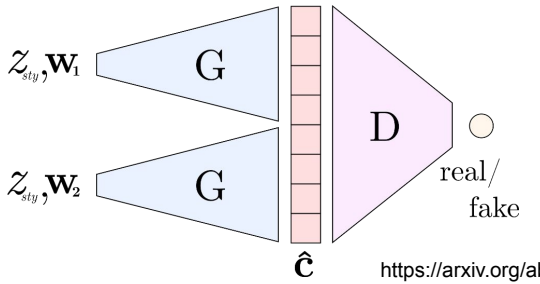
Latent GAN:

FastGAN with **Residual 1D** Conv. Blocks

Spatial **AdaIN** with conditioning signal

R1 Gradient Penalty





https://arxiv.org/abs/2111.01007 https://arxiv.org/abs/2101.04775

Implementation

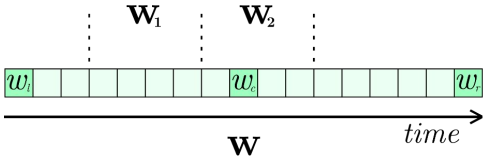
Latent GAN:

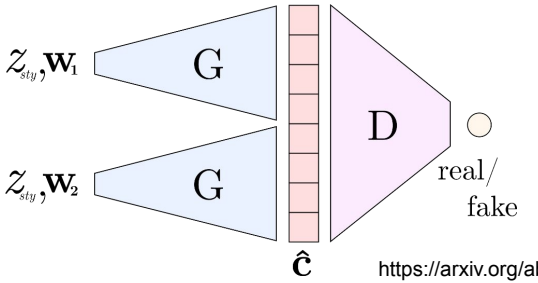
FastGAN with **Residual 1D** Conv. Blocks

Spatial **AdaIN** with conditioning signal

R1 Gradient Penalty

Stereo representations **stacked** channel-wise





https://arxiv.org/abs/2111.01007 https://arxiv.org/abs/2101.04775

Experiments

Autoencoder is **trained** on **general music** dataset (SXSW)

Experiments

Autoencoder is **trained** on **general music** dataset (SXSW)

Experiments

Autoencoder is **trained** on **general music** dataset (SXSW)

It produces **genre-agnostic** compressed **representations**!

Experiments

Autoencoder is **trained** on **general music** dataset (SXSW)

It produces **genre-agnostic** compressed **representations**!

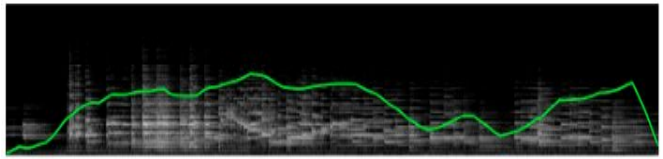
We train a **Latent GAN** on:

Piano Music dataset (*MAESTRO*), **note density** as conditioning
Techno Music dataset (*Jamendo*), **tempo** as conditioning

Conditioning Signals

Piano

MadMom CNN **Onset** detector

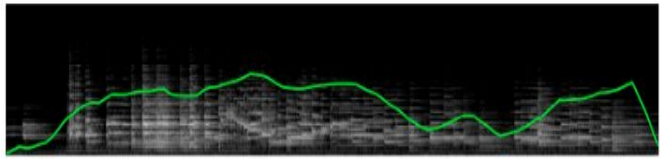


Conditioning Signals

Piano

MadMom CNN **Onset** detector

Note density from Onsets (KDE)

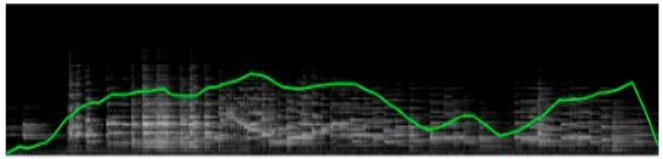


Conditioning Signals

Piano

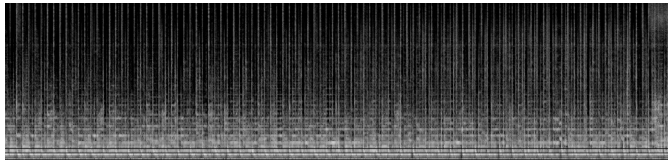
MadMom CNN **Onset** detector

Note density from Onsets (KDE)



Techno

Tempo-CNN to estimate **Tempo**



135 bpm

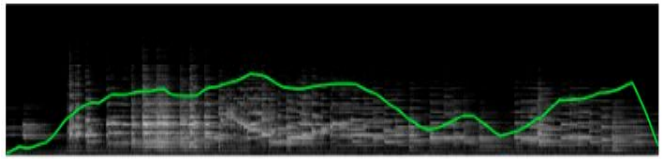
https://arxiv.org/abs/1903.10839

Conditioning Signals

Piano

MadMom CNN **Onset** detector

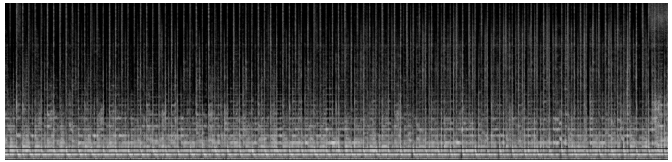
Note density from Onsets (KDE)



Techno

Tempo-CNN to estimate **Tempo**

Constant in training samples

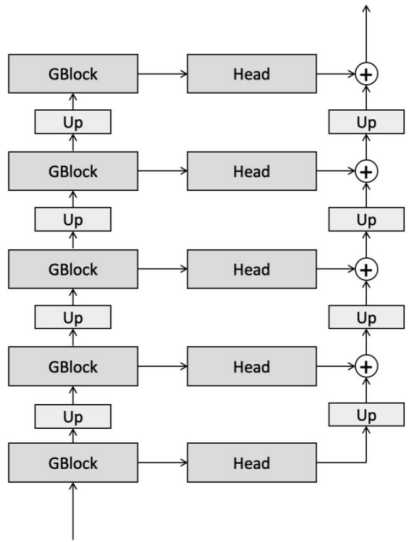


135 bpm

https://arxiv.org/abs/1903.10839

Baseline: UNAGAN

Sequence of **rand. vectors** ⟶ **Mel-spectrograms**

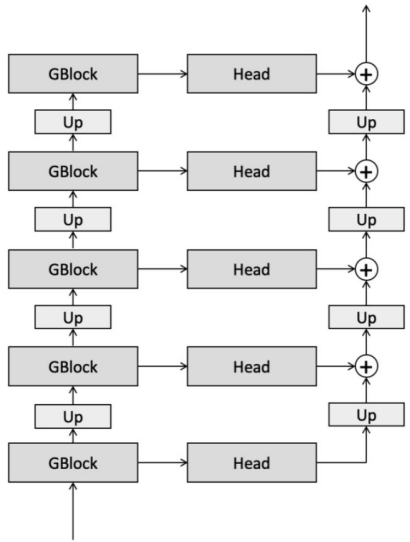


https://arxiv.org/abs/2005.08526

Baseline: UNAGAN

Sequence of **rand. vectors** ⟶ **Mel-spectrograms**

Vectors drawn independently: **short context!**



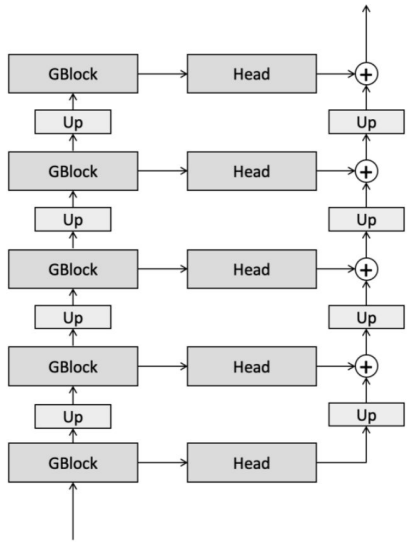
https://arxiv.org/abs/2005.08526

Baseline: UNAGAN

Sequence of **rand. vectors** ⟶ **Mel-spectrograms**

Vectors drawn independently: **short context!**

No parallel generation of same sample!



https://arxiv.org/abs/2005.08526

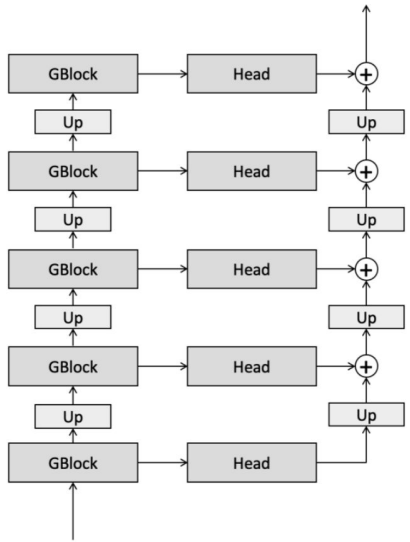
Baseline: UNAGAN

Sequence of **rand. vectors** ⟶ **Mel-spectrograms**

Vectors drawn independently: **short context!**

No parallel generation of same sample!

Relies on **separate vocoder**



https://arxiv.org/abs/2005.08526

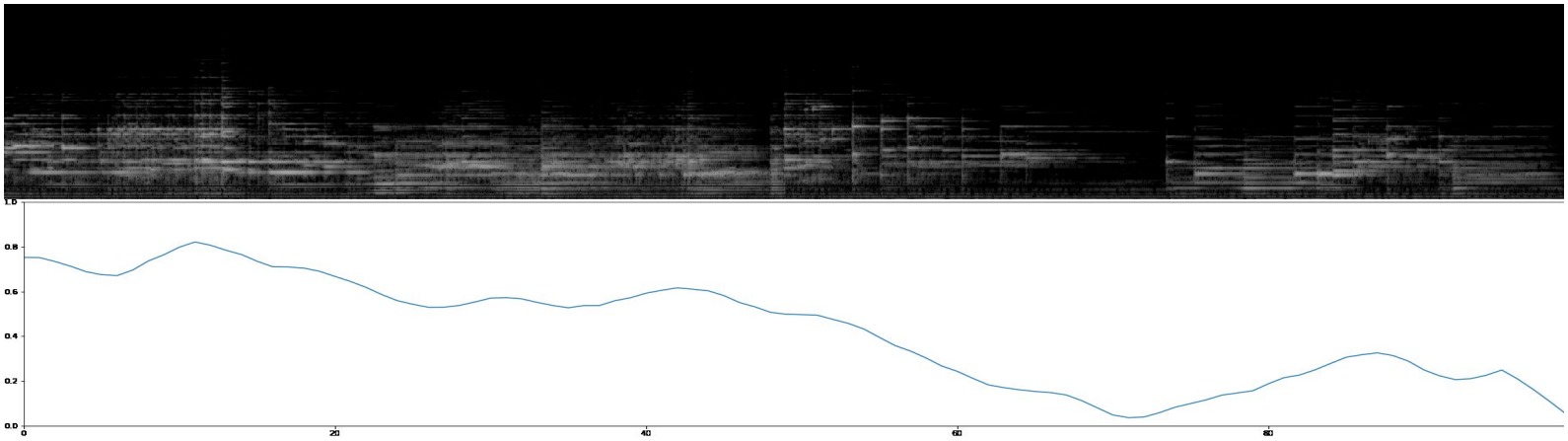
Results

UNAGAN Piano sample (**Baseline**)

Results

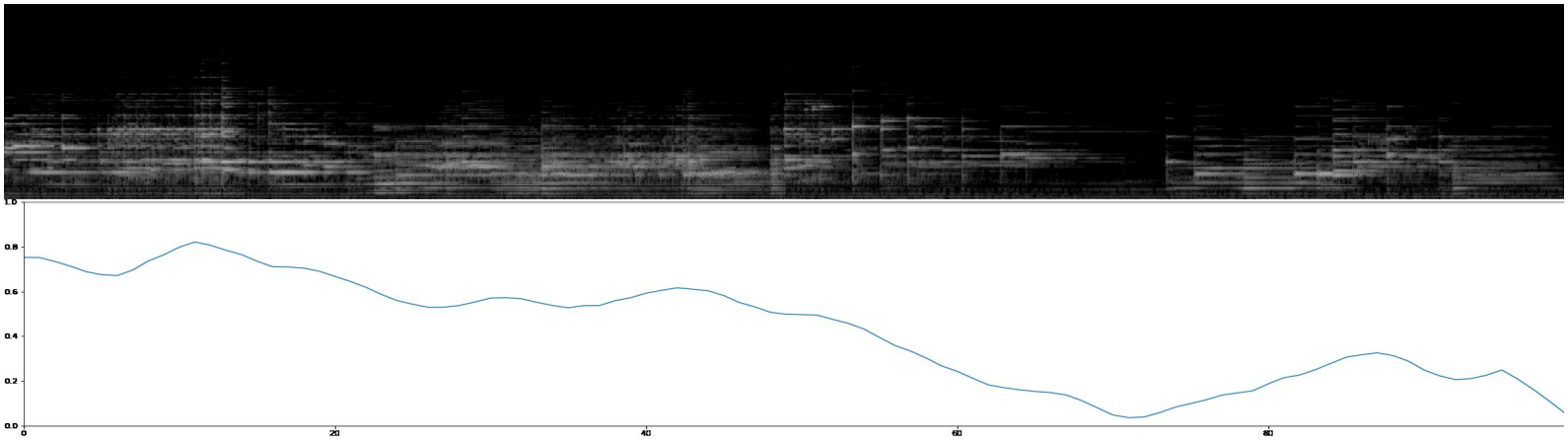
UNAGAN Piano sample (**Baseline**)

Results



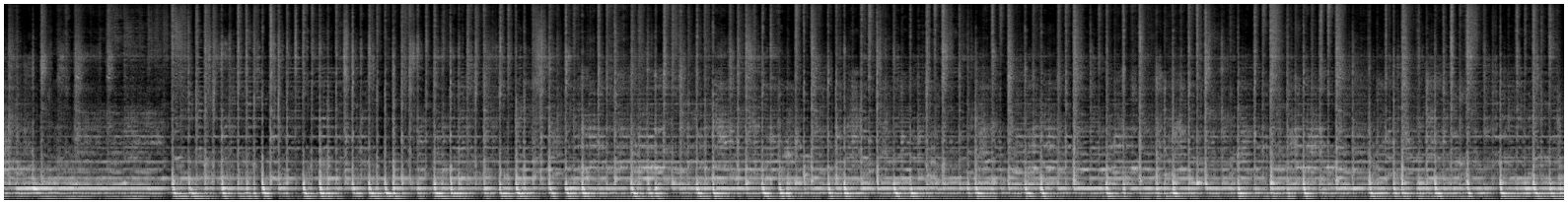
Musika **Piano** (Rand. conditioning)

Results



Musika **Piano** (Rand. conditioning)

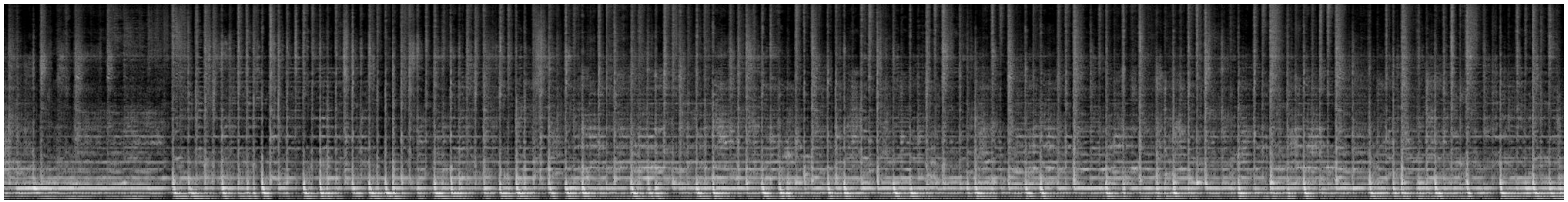
Results



120 bpm

Musika **Techno**

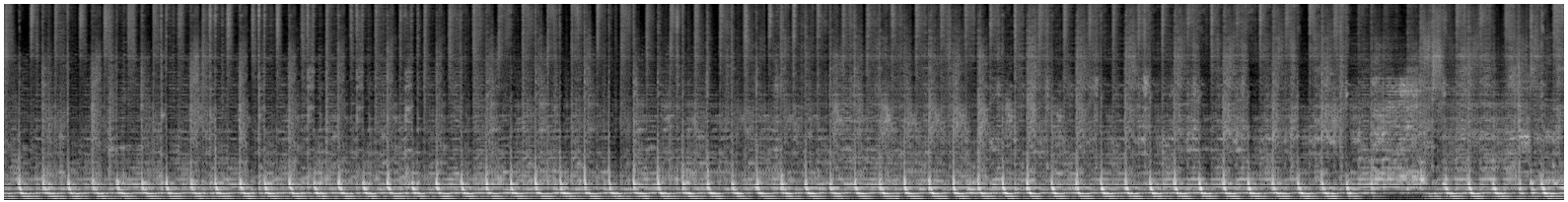
Results



120 bpm

Musika **Techno**

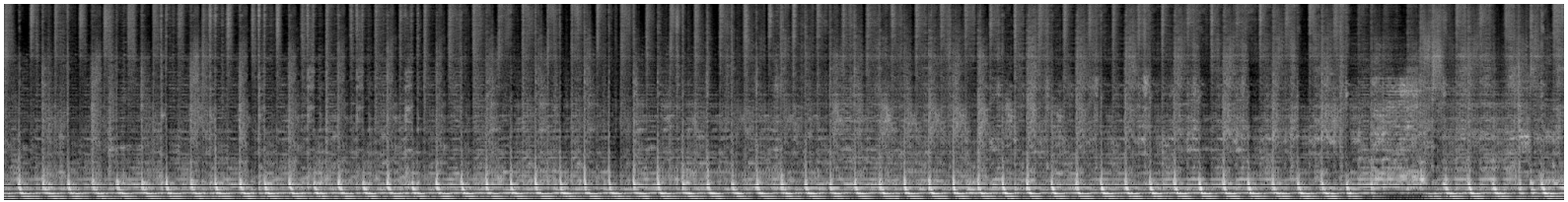
Results



160 bpm

Musika **Techno**

Results



160 bpm

Musika **Techno**

Generation Speed

Model (Faster than real-time)	GPU	CPU
Musika Uncond. Piano	972x	**40x**
Musika Cond. Piano	921x	**40x**
UNAGAN [20] Piano	28x	11x
Musika Uncond. Techno	**994x**	39x
Musika Cond. Techno	917x	39x

Much faster than UNAGAN!

[20] https://arxiv.org/abs/2005.08526

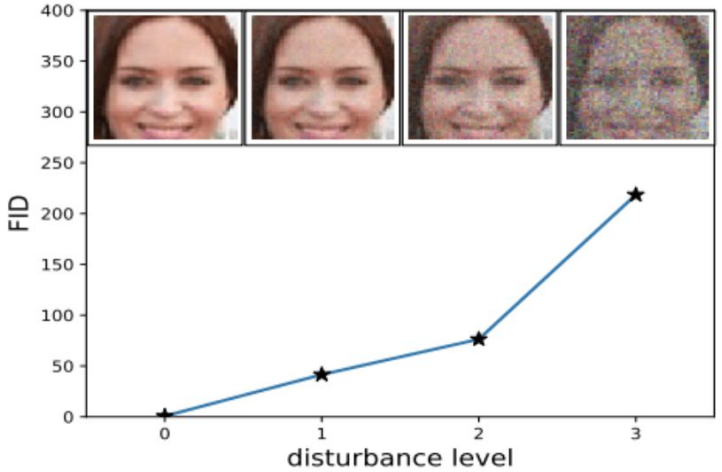
Evaluating GANs

Years of **GANs** research **without metric!**

Evaluating GANs

Years of **GANs** research **without metric!**

For images: IS, **FID**, KID



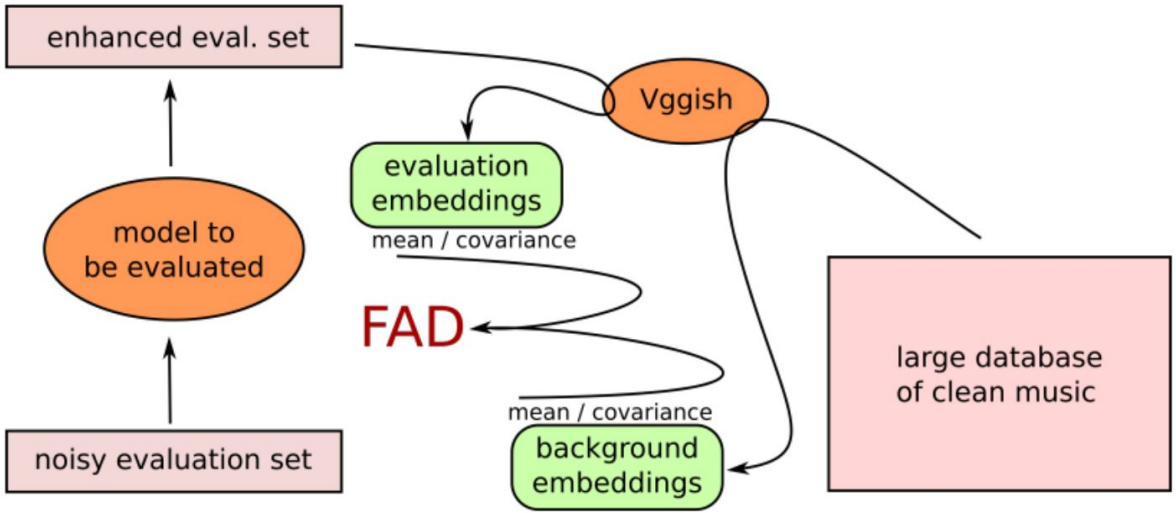
https://papers.nips.cc/paper/2017/hash/8a1d694707eb0fefe65871369074926d-Abstract.html

Evaluating GANs

Years of **GANs** research **without metric!**

For images: IS, **FID**, KID

For audio: **FAD** (Frechét Audio Distance)



$$\mathbf{F}(\mathcal{N}_b, \mathcal{N}_e) = \|\mu_b - \mu_e\|^2 + tr(\Sigma_b + \Sigma_e - 2\sqrt{\Sigma_b \Sigma_e})$$

https://arxiv.org/abs/1812.08466

FAD Problems

Calculated with **1 second snippets**

FAD Problems

Calculated with **1 second snippets**

Evaluates sound **quality**, **not** long-range **coherence**

FAD Problems

Calculated with **1 second snippets**

Evaluates sound **quality**, **not** long-range **coherence**

No quantitative **metrics** currently exist!

Model	FAD
Musika Uncond. Piano	**1.641**
Musika Cond. Piano Rand.	2.150
Musika Cond. Piano Const. 0.15	2.584
Musika Cond. Piano Const. 0.30	3.400
Musika Cond. Piano Const. 0.45	4.389
Musika Cond. Piano Const. 0.60	4.839
Musika Cond. Piano Const. 0.75	5.434
UNAGAN [20] Piano	11.183

Lower Frechét Audio Distance than UNAGAN!

[20] https://arxiv.org/abs/2005.08526

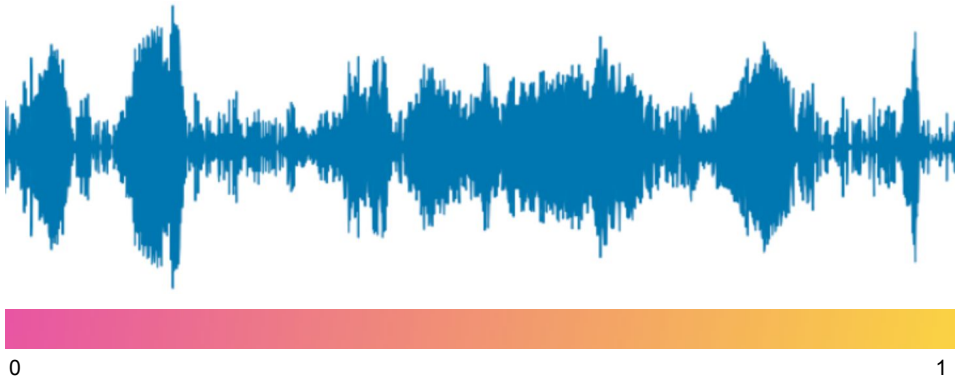
Future/Current Work

22.05 kHz  44.1 kHz **Not Trivial!**

Future/Current Work

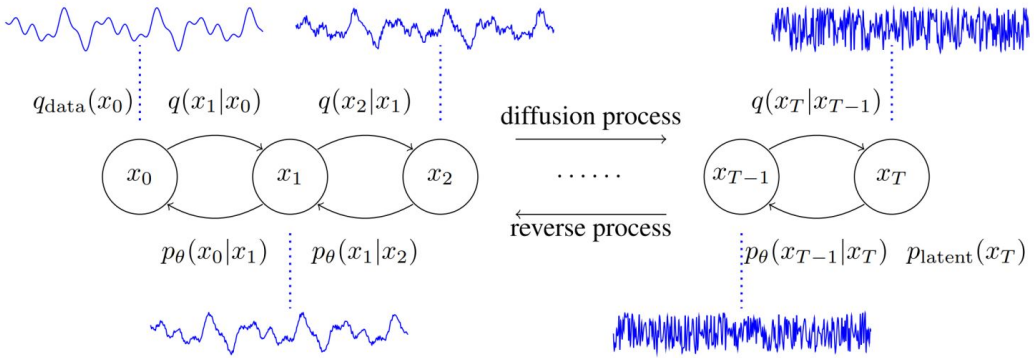
22.05 kHz → 44.1 kHz **Not Trivial!**

Positional conditioning for coherent **Song Structure**



Future/Current Work

GANs ⟶ Diffusion Models (Better with more data)

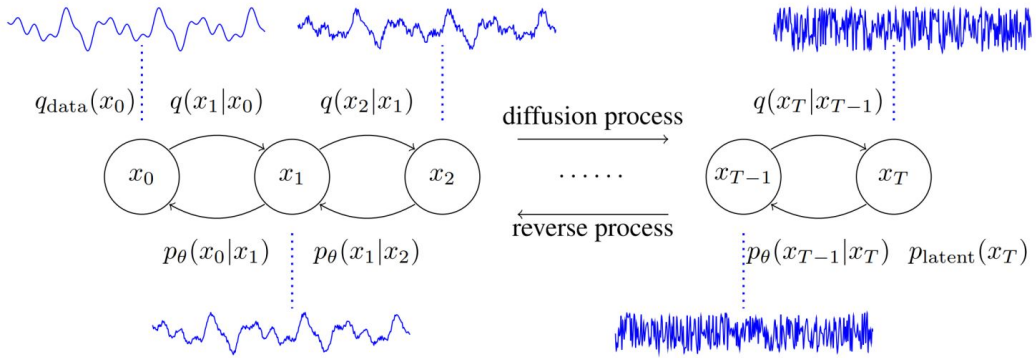


https://arxiv.org/abs/2009.09761

Future/Current Work

GANs → Diffusion Models (Better with more data)

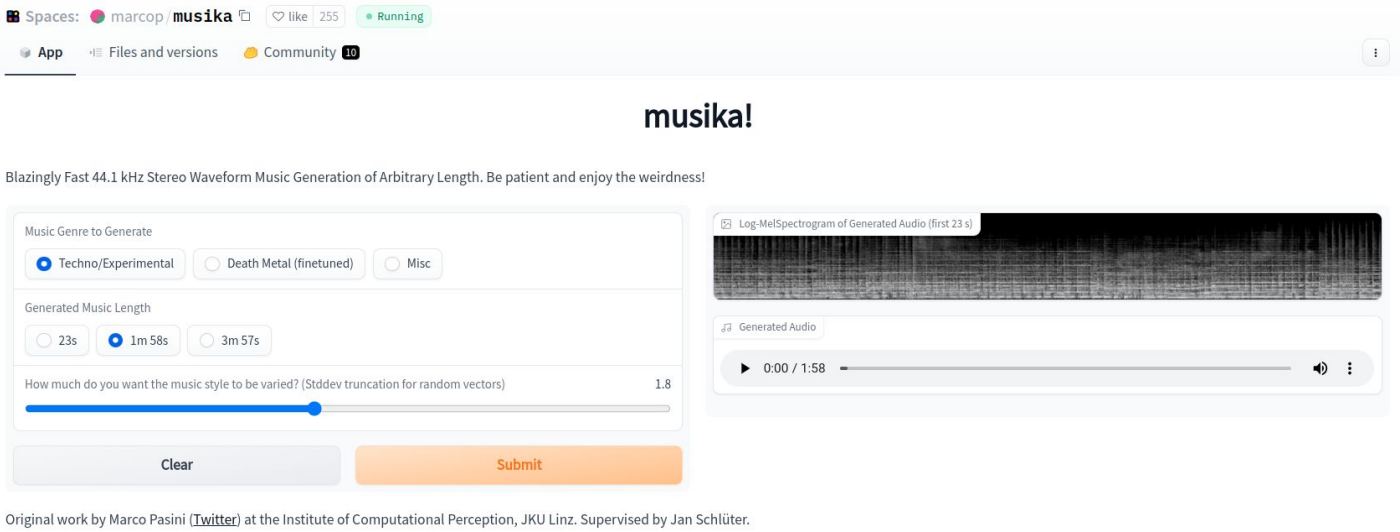
Arbitrary length parallel generation **not possible!**



https://arxiv.org/abs/2009.09761

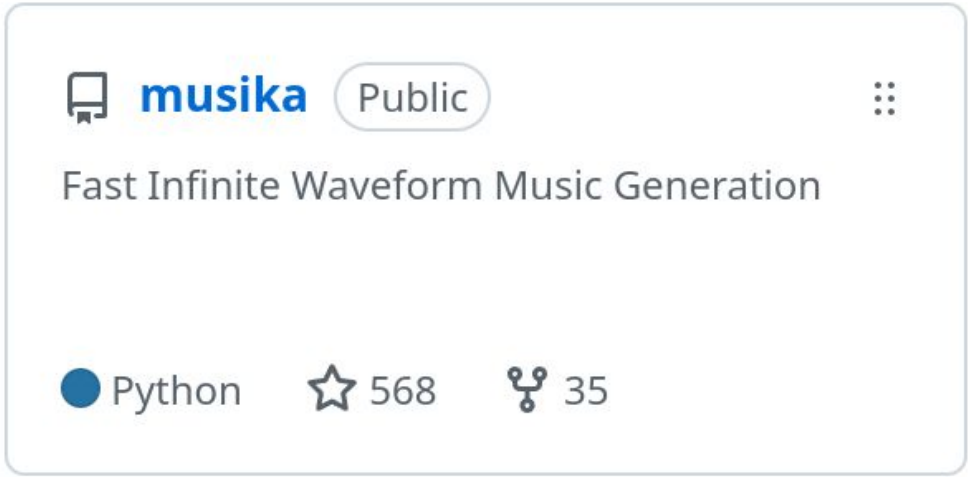
Reception

Reception



Popular **Online Demo**!

Reception



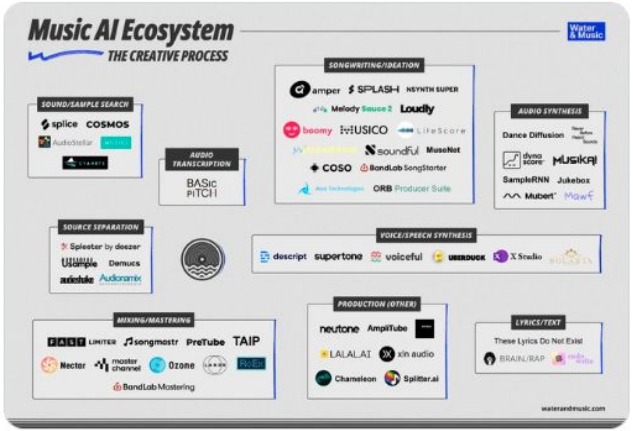
Popular **Github** repository!

Reception

Water & Music @water_and_music · 1 nov

we've seen lots of generative AI market maps making the rounds on Twitter. a huge gap: none of them cover music with the same amount of breadth as visual art and text.

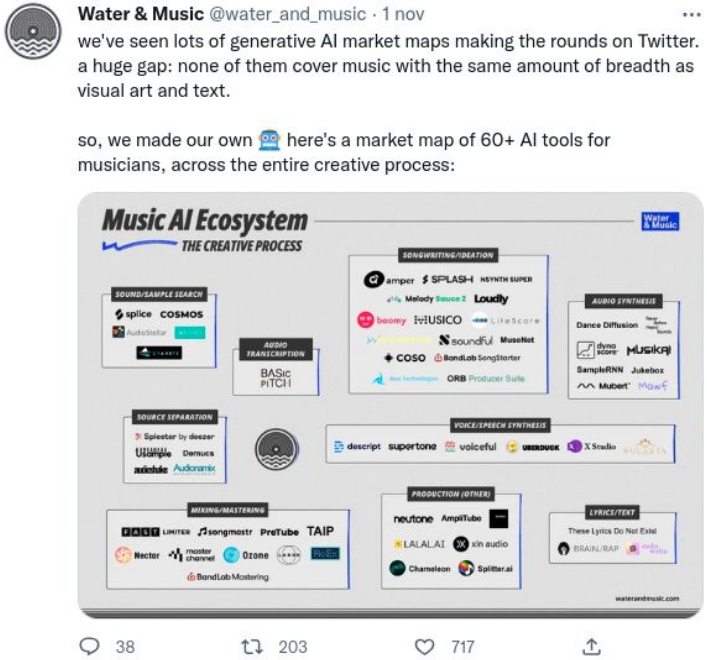
so, we made our own 🤖 here's a market map of 60+ AI tools for musicians, across the entire creative process:

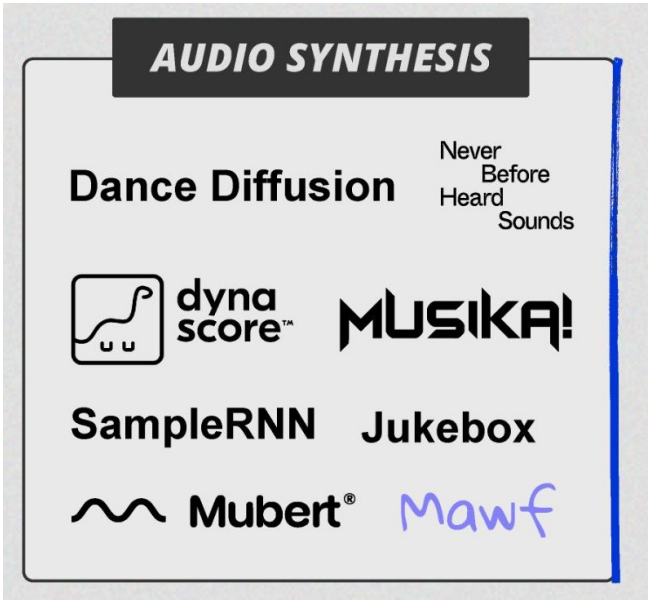


38 203 717

AI Music tools Market **Map**

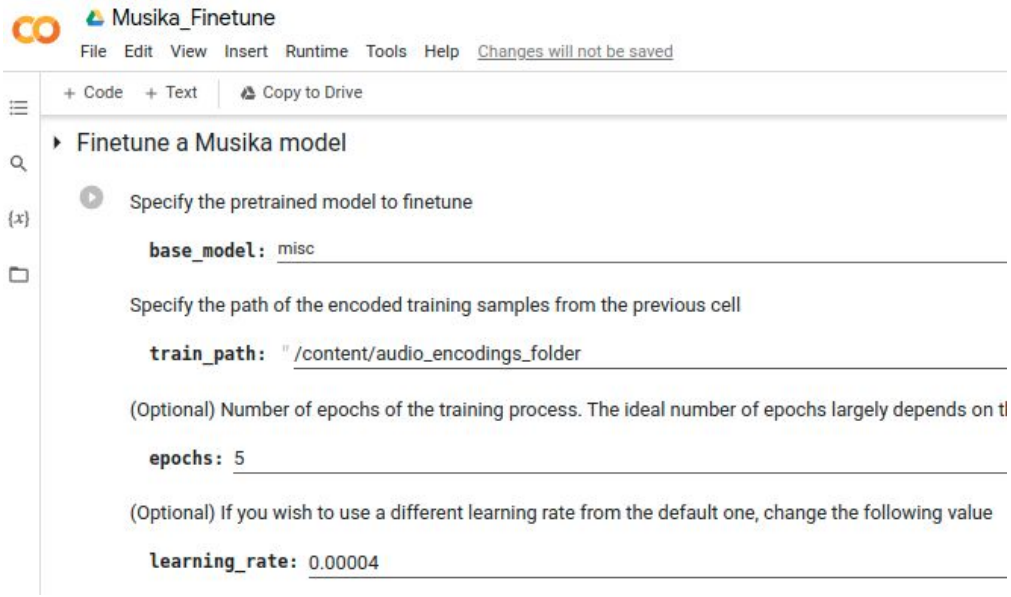
Reception





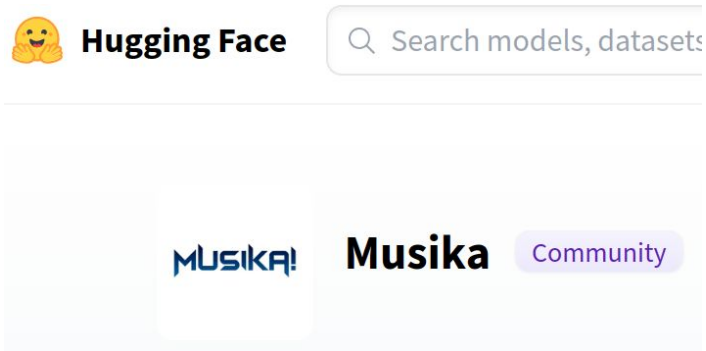
AI Music tools Market **Map**

Ecosystem



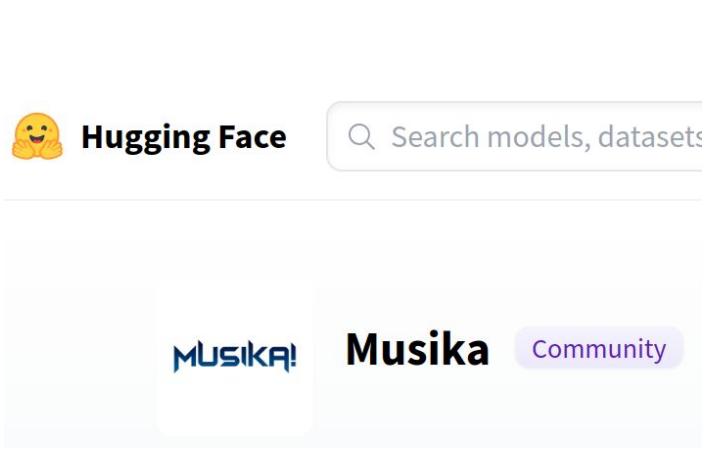
Custom **Training** made **Easy**!

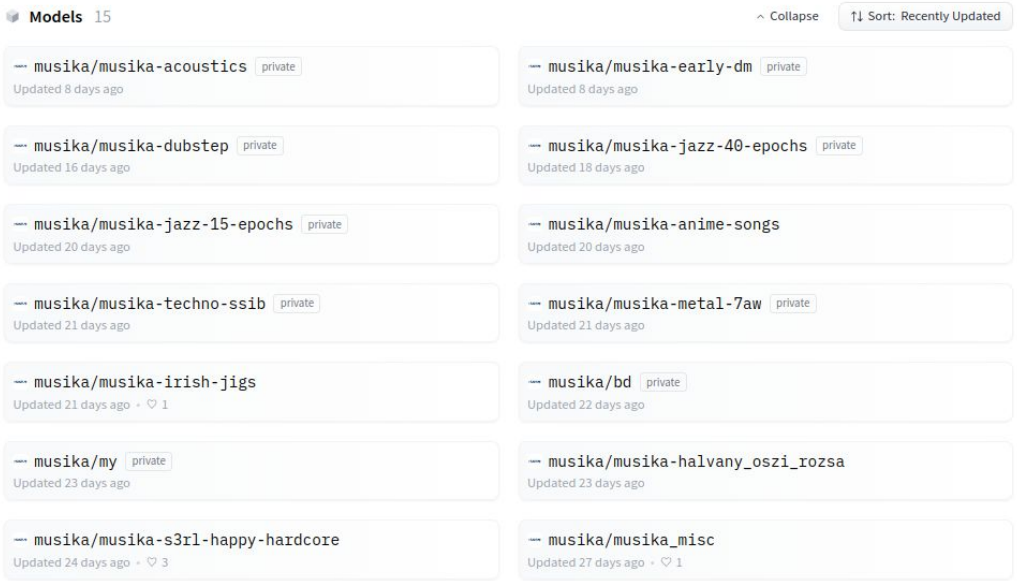
Ecosystem



Share your custom **Model**!

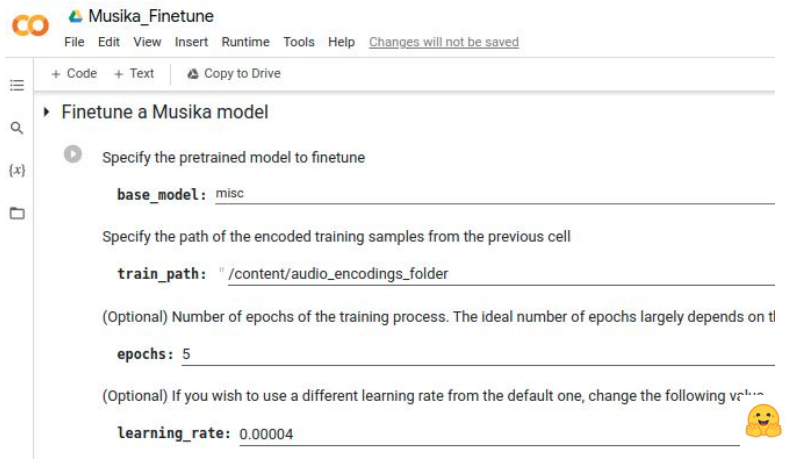
Ecosystem

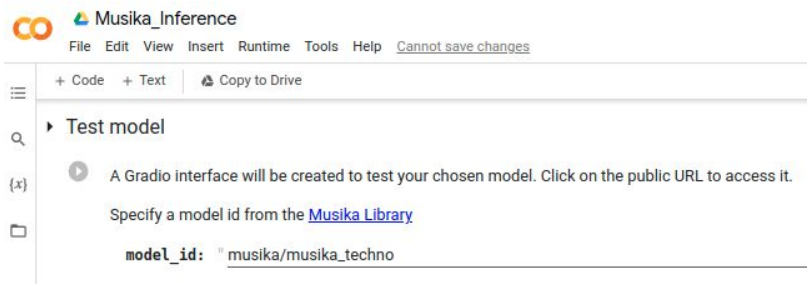




Share your custom **Model**!

Ecosystem

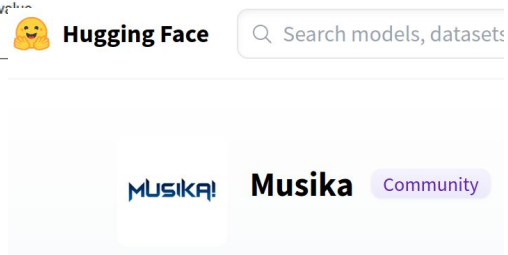




Train

Share

Test



Streamlined process

Conclusion

Solving a technical **challenge**:

Fast music **generation** of sufficient **quality**, conditioned on **user input**

Conclusion

Solving a technical **challenge**:

Fast music **generation** of sufficient **quality**, conditioned on **user input**

We **release code** and weights:

A **system** can be **trained on a new** music **domain in hours!**

What can Musika generate today?

(44.1 kHz + □ Structure + □ Sound Quality)


MUSIKAI

Thank you!

Huge thanks to Jan Schlüter
and to the Institute of Comp. Perception for the support!