Pseudo Autoregressive Inference Using Diffusion Model for Piano Audio Synthesis

Yuze Wang, Chengkai Kang.

Motivation

Diffusion Probabilistic Model show promising results in generating consistent data distribution. Can we use it in the domain of audio generation?

However, big issue exist in the architecture for DDPM model as it is non autoregressive. This is fine for Images, but for music, autoregressive behavior is essential for track creations…

Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” 2020.

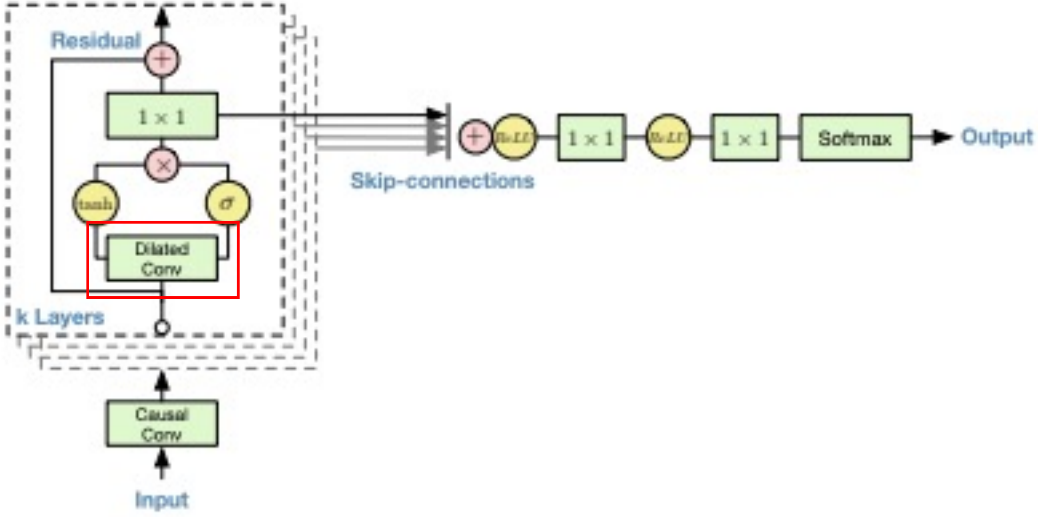
Circumvention

Remember that autoregressive model is just a way to use accumulated historical data to forecast the next token or sample.

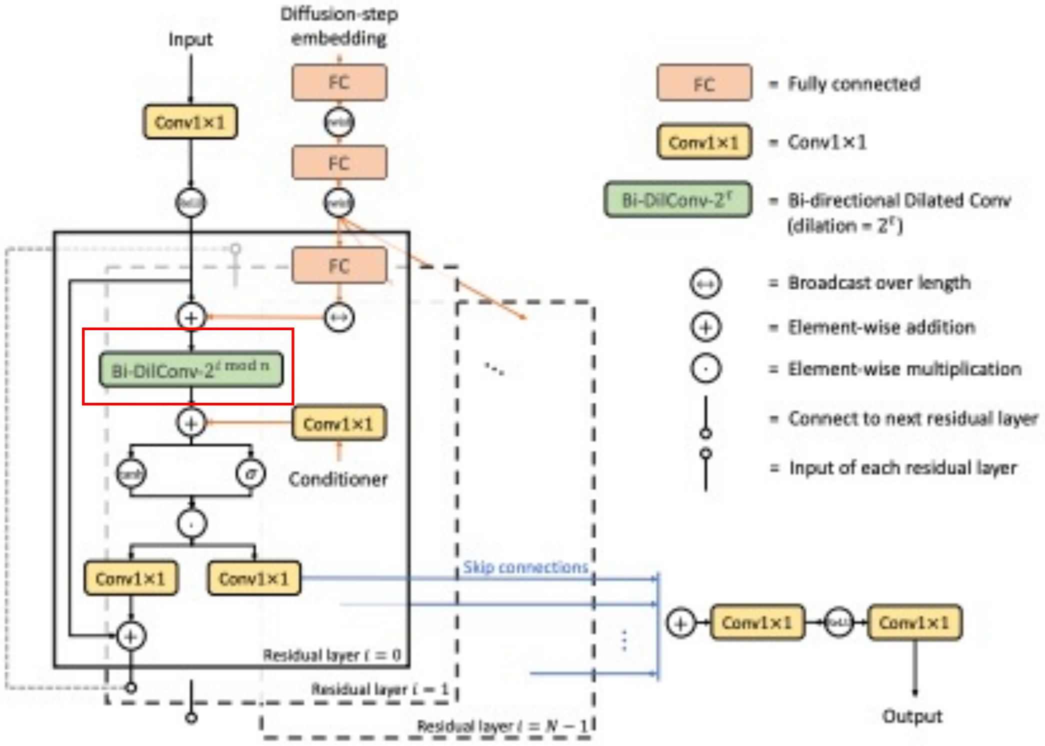
$$p(\mathbf{x}) = \prod_{i=1}^{D} p(x_i|\mathbf{x}_{<i})$$

In essence, it utilizes the temporal dependency during generation. As long as our model captures the temporal dependency, it should be modifiable it to mimic the autoregressive behavior.

Wavenet and Diffwave



Wavenet



Diffwave

A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, “Wavenet: A generative model for raw audio,” 2016.

B. Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, “Diffwave: A versatile diffusion model for audio synthesis,” 2021.

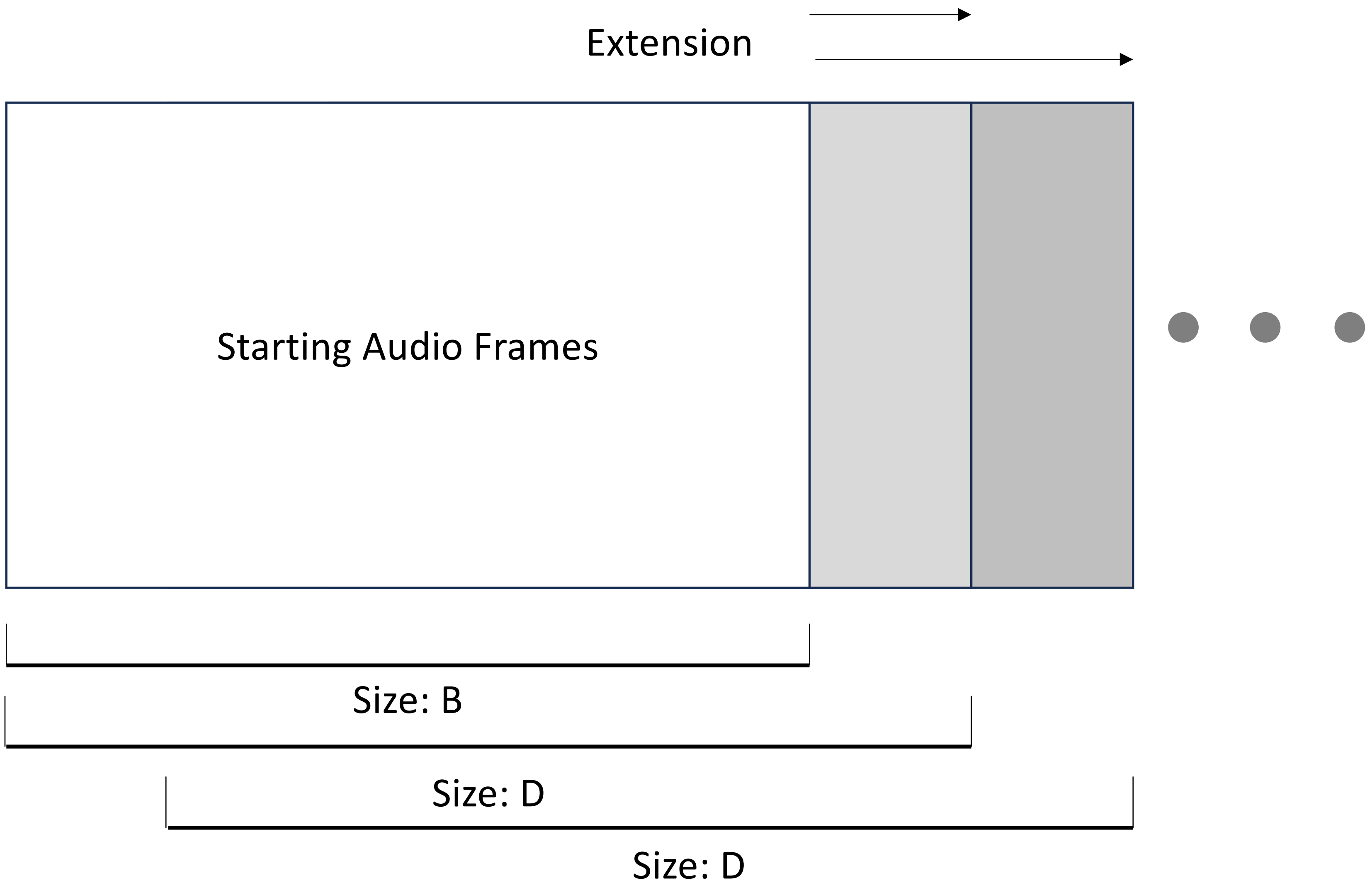
Audio Extension

Since each generation of diffusion model samples from a new distribution, (which means the loss of original reference), we can’t generate a new audio and concatenate them together to create a longer sequence. But rather, we need to extend the current generation to have a longer sequence length.

Therefore, our objective is to generate a new data with size D from known data (or given) of size B. such that D > B. This can be conceptualized as:

$$logp(data^{new}) = p(data^{new}|data^{known})$$

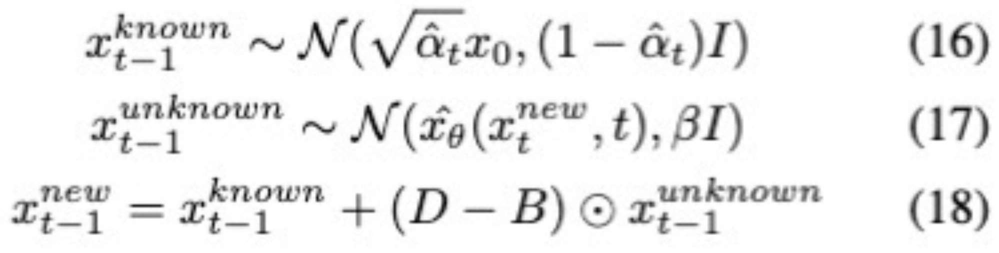
Continuous Extension



Because there is no limit in generation length for a real autoregressive model, we can replicate this behavior by stacking the generated frames.

Audio Inpainting

Extension problem can be expressed as filling problem, where we assume an audio is missing a length of D-B sequence. Inspired by RePaint, this is approachable with diffusion.



$$x_{t-1}^{known} \sim \mathcal{N}(\sqrt{\hat{\alpha}_t} x_0, (1 - \hat{\alpha}_t) I) \tag{16}$$

$$x_{t-1}^{unknown} \sim \mathcal{N}(\hat{x}_\theta(x_t^{new}, t), \beta I) \tag{17}$$

$$x_{t-1}^{new} = x_{t-1}^{known} + (D - B) \odot x_{t-1}^{unknown} \tag{18}$$

In these equations, The + indicates the concatenation of two 1-D matrices, while ⊙ signifies the selection of a length segment

Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. V. Gool, “Repaint: Inpainting using denoising diffusion probabilistic models,” 2022.

Audio Inpainting

However, there are problems with this method that led to unharmonized results, which are identified by Repaint:
1. The sampling of x^new excludes the B ⊙ x^unkown region, leading to a loss of crucial information in each reverse step.
2. The diminishing β value during the diffusion process limits the model's ability to introduce significant changes to the latent distribution at lower t values

Repaint author approach this problem by introducing additional diffusion steps to harmonize the laten cross time t. However, this method introduces additional runtime can resource usage, causing the diffusion process to run slower.

Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. V. Gool, "Repaint: Inpainting using denoising diffusion probabilistic models," 2022.

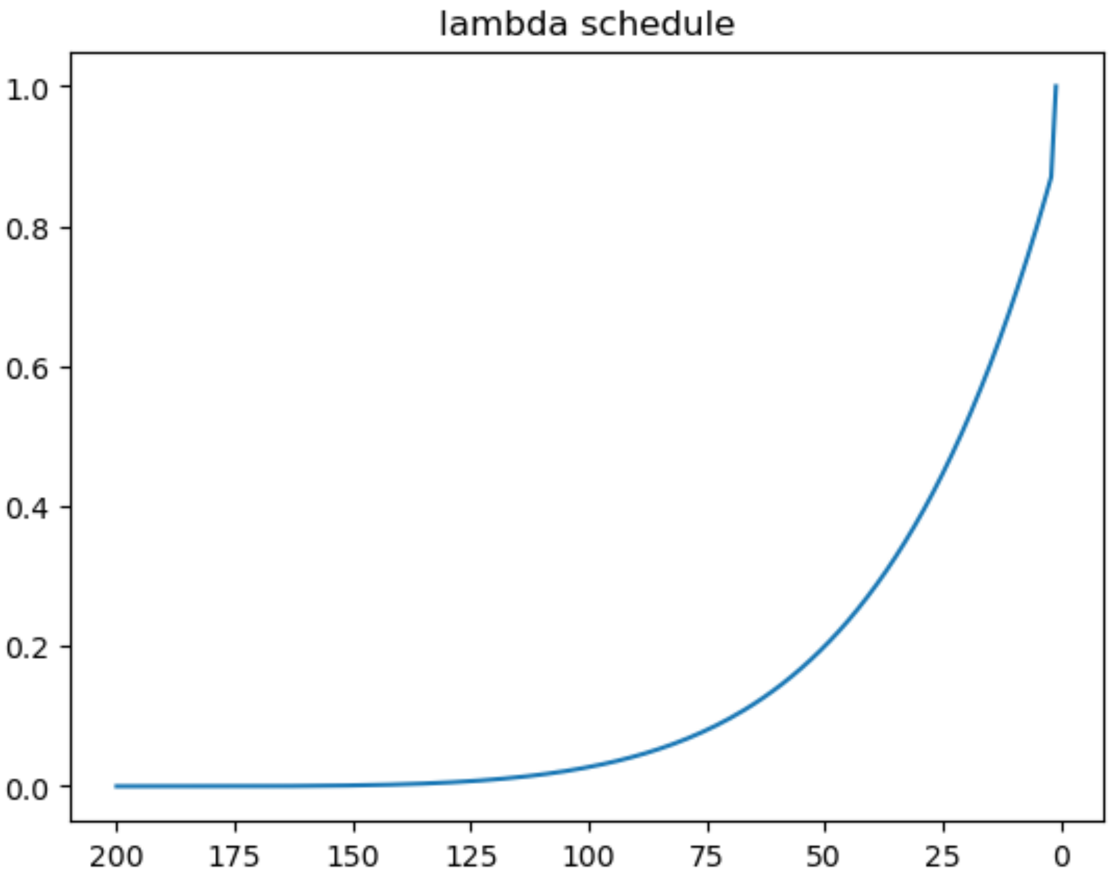
Interpolation and Reverse Lambda

We can solve the harmonization issue by tackling the problems identified prior.
To solve the lost of information issue, we can need to incorporate the information x^unkown and x^known.
However, we can’t simply add them together because it would lead to exploding numeric issue, interpolation lambda is used to ensure that the latent are still within a numeric bound.

$$x^{combined} = \lambda x^{unknown} + (1 - \lambda) x^{known}.$$

This interpolation creates a latent distribution that is intermediate between x^unkown and x^known. Which is used to guide the (D-B) ⨀ x^unkown region.

Interpolation and Reverse Lambda



Ultimately, we want the latent to be an independent distribution instead of an intermediate, which lambda = 1. We decides to design our lambda as function that gradually changes to 1 as t approach 0. This is done to ensure that model have a smoother transition from one step to another.

We designed our lambda function as an exponential function because most denoising activity occurs at lower steps, therefore a more dramatic change necessitate for effective latent shaping.

lambda at the final t will always equal to 1 to avoid generate an intermediate between two distribution
We found that reconstruction quality is a lot worse using a linear schedule

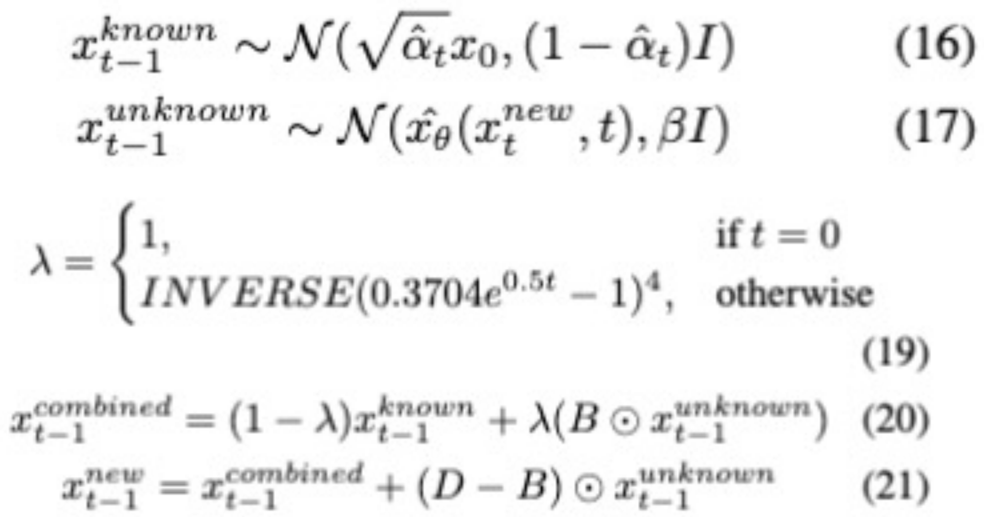
Interpolation and Reverse Lambda

Although is logical to increase lambda for x^known as t approaches 0, which ensure the diffused x0 have a matching region as x^known, this actually doesn’t solves the harmonization issue because of diminishing β issue. (When the x^known are taking significant influence, little β is left allowing model to make meaningful changes)

Counterintuitively, we decrease the lambda for x^known as t approaches 0, while increase lambda for x^known at high t value. We found out that not only does this solves the diminishing β value, x0 still contain regions that **resembles** the original x^known.

We suspect that Model at Lower T value are specifically focused on ”denoising” the latent, while higher T value are focused on creating a uniformed the distribution.

Interpolation and Reverse Lambda



$$x_{t-1}^{known} \sim \mathcal{N}(\sqrt{\hat{\alpha}_t}x_0, (1-\hat{\alpha}_t)I) \tag{16}$$

$$x_{t-1}^{unknown} \sim \mathcal{N}(\hat{x_\theta}(x_t^{new}, t), \beta I) \tag{17}$$

$$\lambda = \begin{cases} 1, & \text{if } t = 0 \\ INVERSE(0.3704e^{0.5t} - 1)^4, & \text{otherwise} \end{cases} \tag{19}$$

$$x_{t-1}^{combined} = (1-\lambda)x_{t-1}^{known} + \lambda(B \odot x_{t-1}^{unknown}) \tag{20}$$

$$x_{t-1}^{new} = x_{t-1}^{combined} + (D - B) \odot x_{t-1}^{unknown} \tag{21}$$

Eq 20, “+” means matrix addition while Eq 21 “+” means matrix concatenation

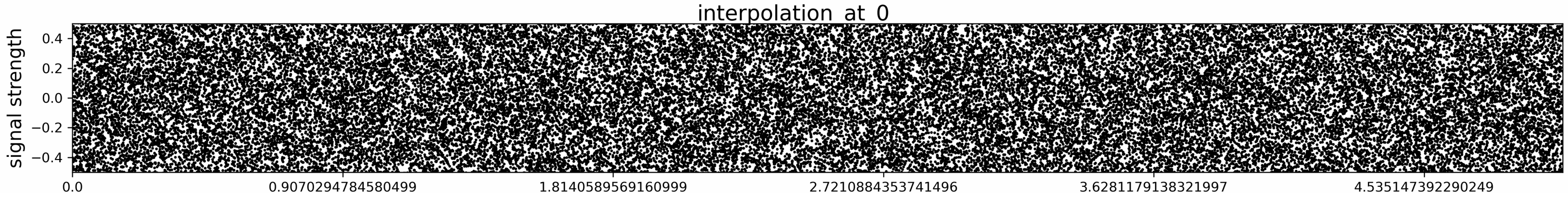
lambda function is designed specifically for 200 steps diffusion, Do not use for diffusion models uses more or less than 200 steps

Experiment Setup

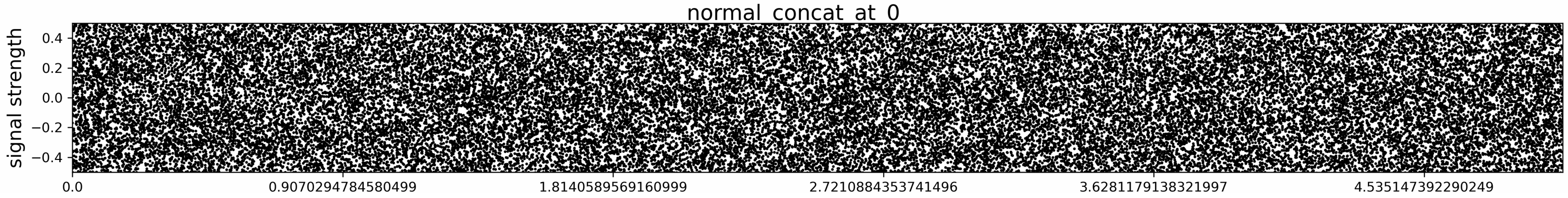
- 30 residual layers and maximum of 64 residual channels.
- Dataset compose of 14 monophonic piano sounds of varying length sampled at 22.05KHz. Total duration is 1217 seconds or approximately 20.283 minutes.
- trained over 730,000 steps on A5000 GPU for 14 days. Final L2 loss is around 0.03

Maximum length Diffwave can generate a 5 second audio equal to a tensor of [1, 110250].

Our Method



Baseline

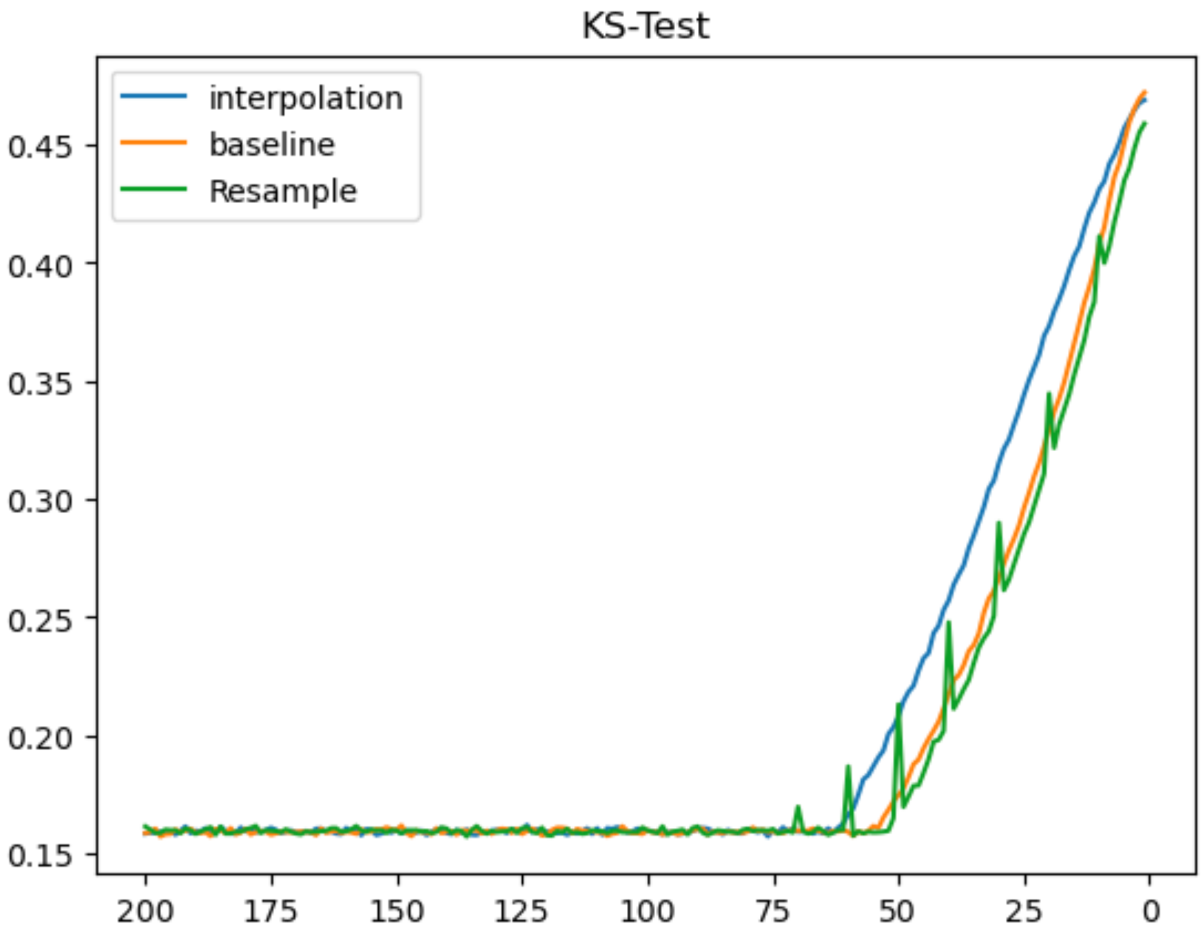


Repaint

repaint_concat_at_0
signal strength
0.4
0.2
0.0
−0.2
−0.4
0.0
0.9070294784580499
1.8140589569160999
2.7210884353741496
3.6281179138321997
4.535147392290249

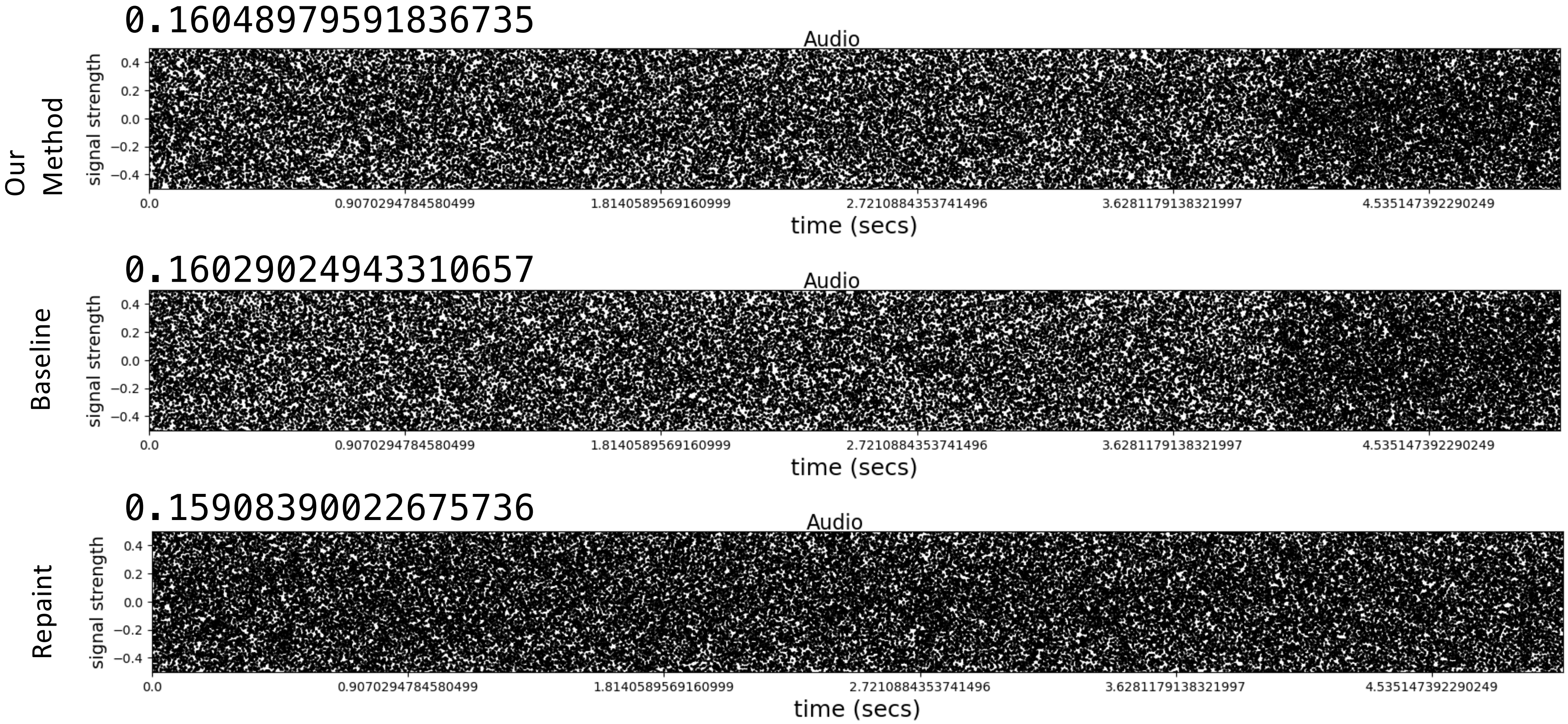
Baseline just follows the vanilla method. Eq 16-18

KS history



Kolmogorov-Smirnov Test against original audio, higher the number, better the result. Torch Seed: 2023

T = 100



Our Method

0.16048979591836735

Baseline

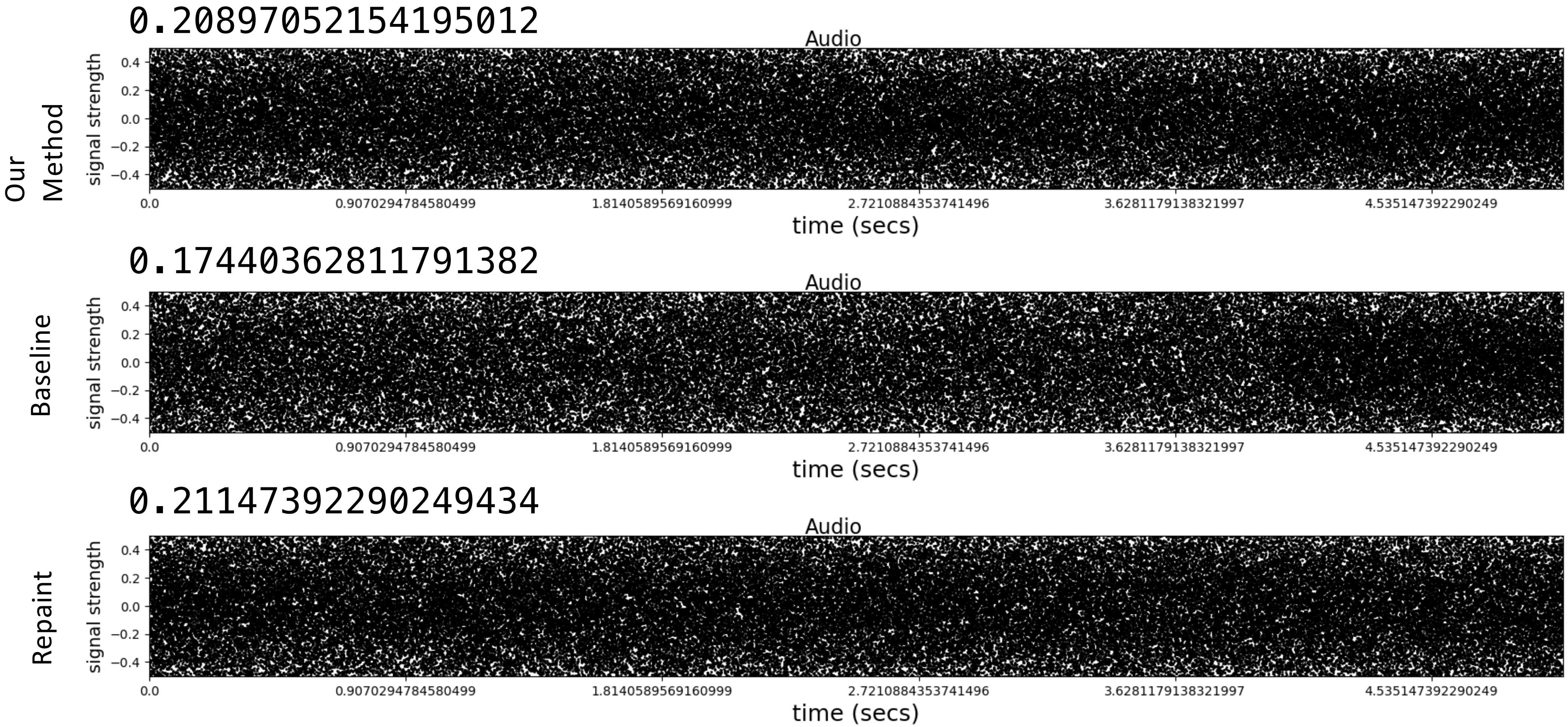
0.16029024943310657

Repaint

0.15908390022675736

Kolmogorov-Smirnov Test against original audio at t100, higher the number, better the result. Torch Seed: 2023

T = 50



Our Method: 0.20897052154195012

Baseline: 0.17440362811791382

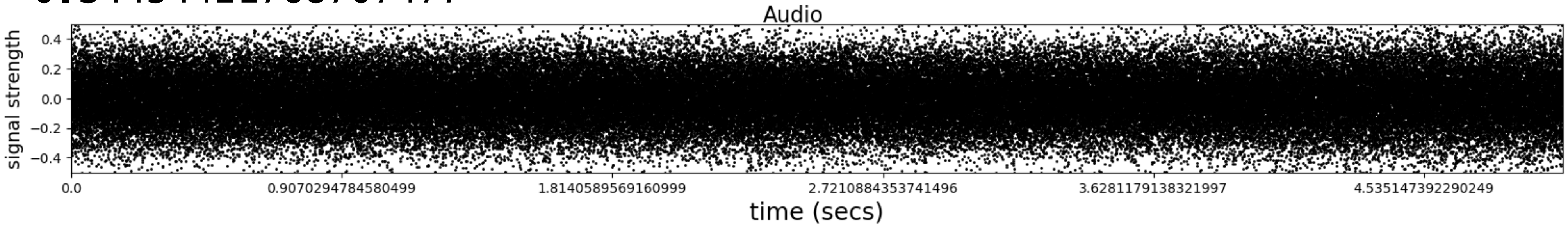
Repaint: 0.21147392290249434

Kolmogorov-Smirnov Test against original audio at t50, higher the number, better the result. Torch Seed: 2023

T = 25

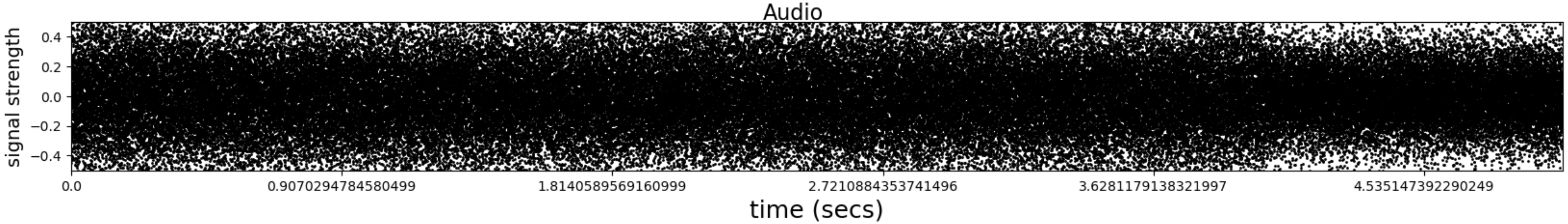
Our Method

0.34454421768707477



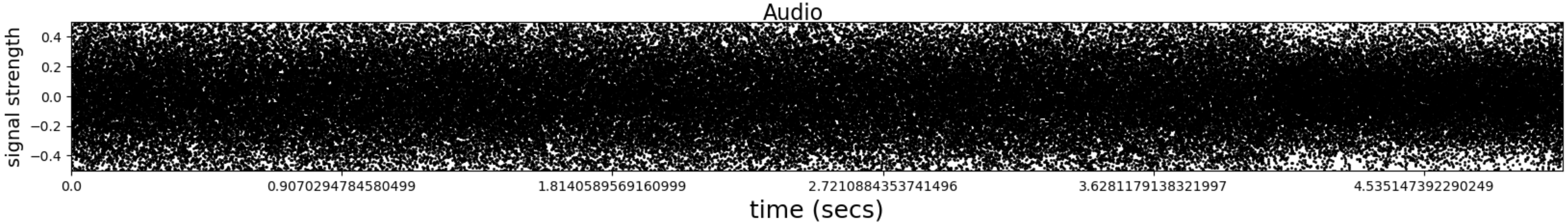
Baseline

0.29627210884353744



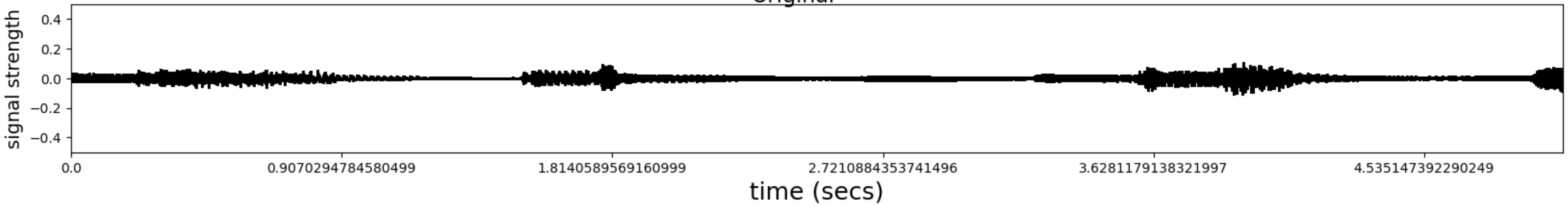
Repaint

0.2828390022675737

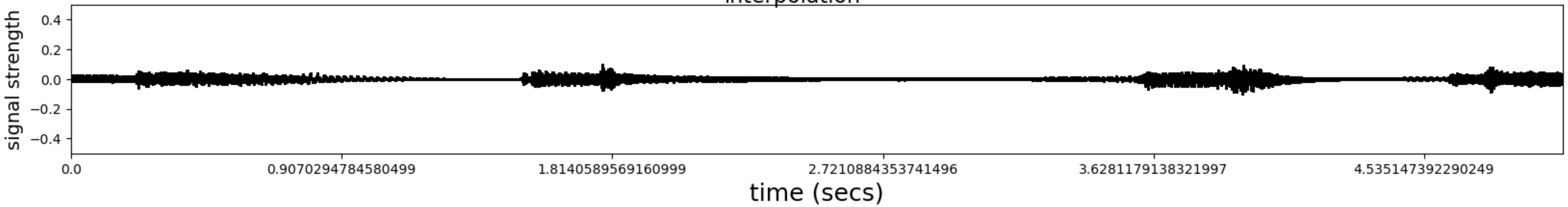


Kolmogorov-Smirnov Test against original audio at t25, higher the number, better the result. Torch Seed: 2023

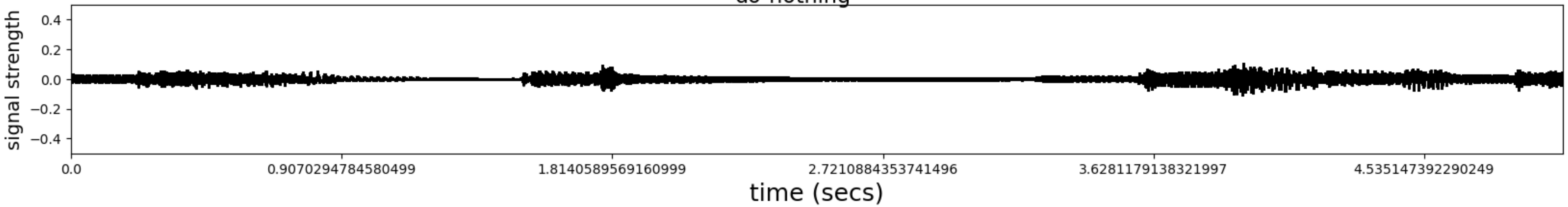
Original



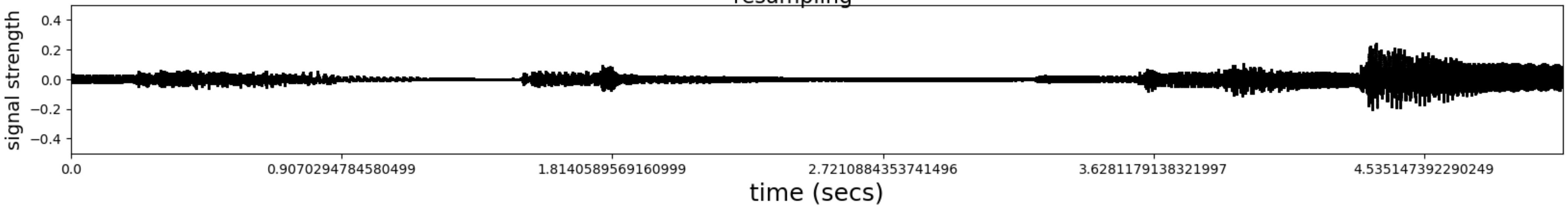
interpolation



do-nothing

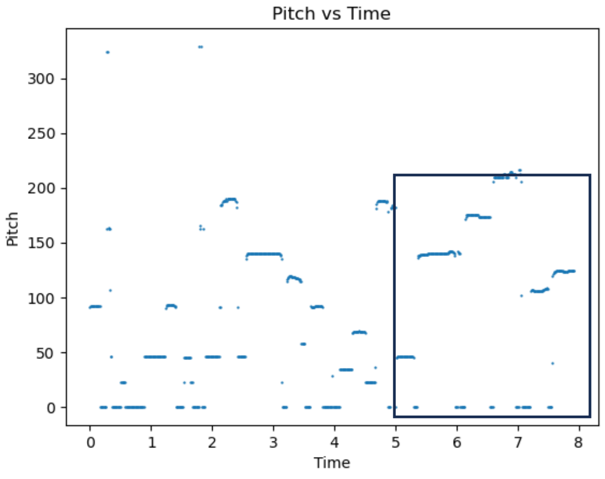


resampling

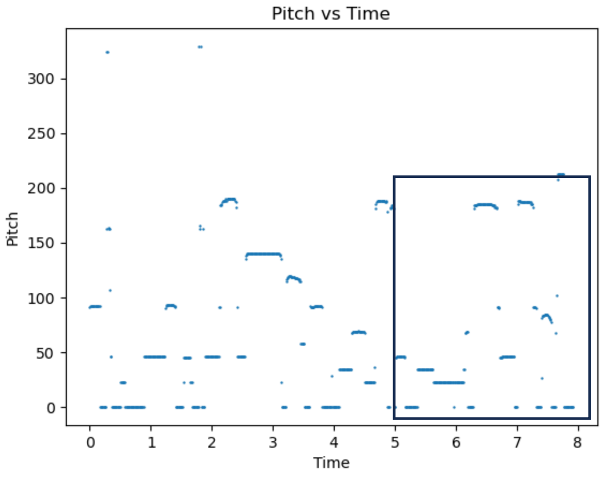


8 second extension pitch analysis

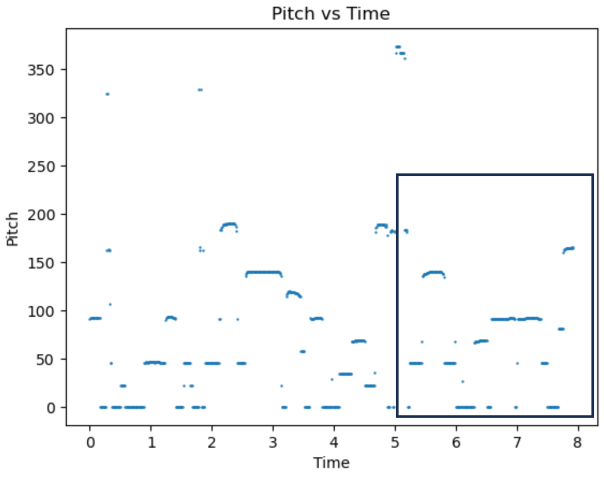
Our Method



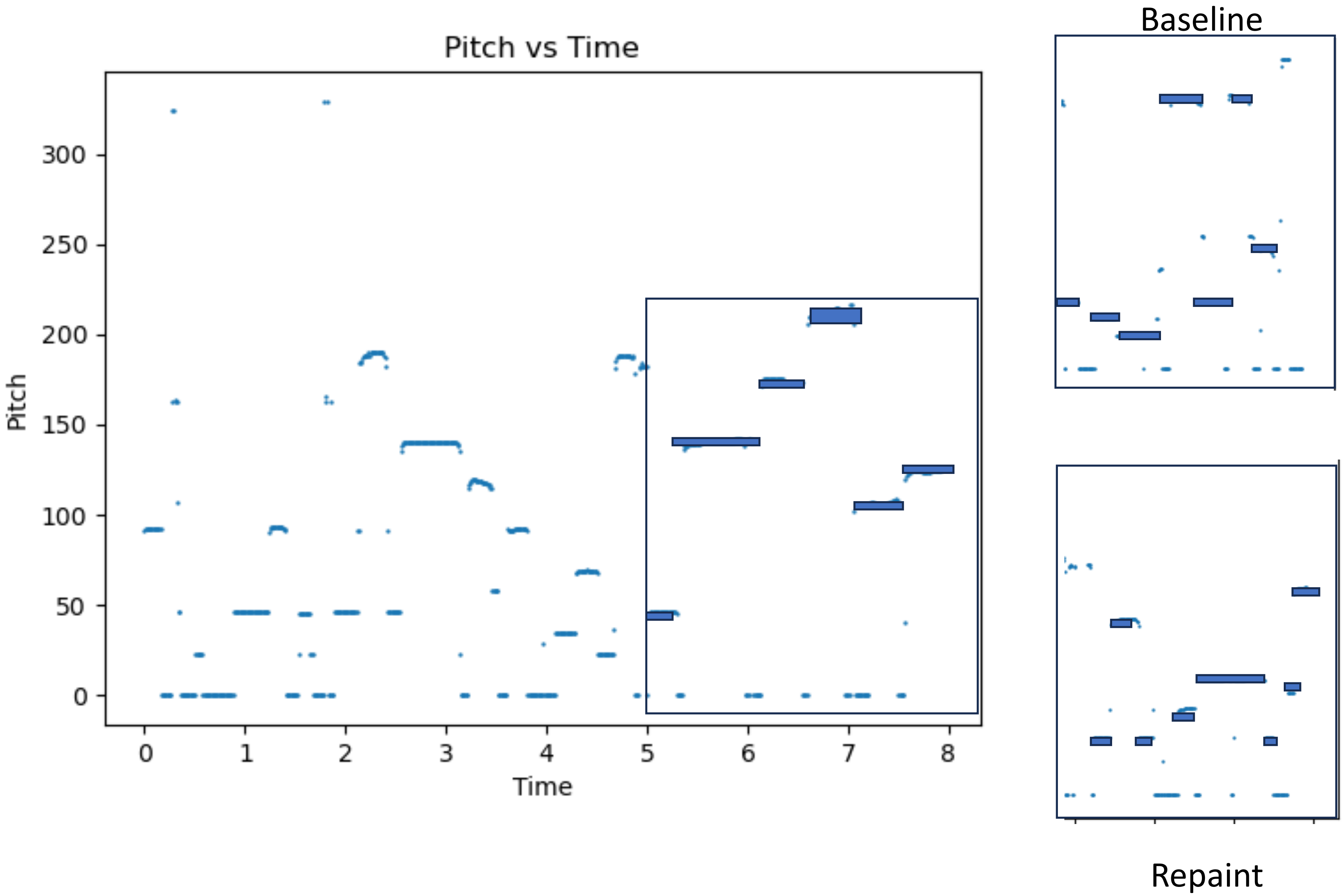
Baseline (do nothing)



Repaint



Yin algorithm, window size 1023, step size 220, f0 between 11 and 552, Random Seed


Pitch vs Time
Pitch
300
250
200
150
100
50
0
Time
0
1
2
3
4
5
6
7
8
Baseline
Repaint

Discussion

- Although Our study shows a superior results, it is not definitive. As the Diffwave we trained still hasn’t been able to create melody that contains any meaning. Ad hoc study only demonstrated our method creates a more uniformed sequence across diffusion time.

- Since we are working with limited resource, the metric we use are not comprehensive. Ideally, we would want to measures the Meaning Opinion Score (MOS) for some high-quality evaluation.

- Although the x0 have regions that resembles that x^known, it is not a perfect reconstruction even it is insusceptible to human ear.

- We are unsure this would work on other architecture or other problem domain. Since it is based on the fact that Diffwave generate audio sequence with consideration of temporal dependency.

Questions?

https://github.com/Yuze-Wang-RoyAnnd/DDPM_For_Audio/