# Fine tuning Stable Diffusion

Zaborovskaia Valeria
*Optimization Class Project. MIPT*

## Introduction

There are many approaches to solving generation problem such as GAN, VAE, Flow-based models etc. Another idea is Diffusion Model which showed a great performance in this field. The key idea is to slowly destroy structure in a data distribution through an iterative forward diffusion process. The process of training such model requires large computational resources, and even fine-tuning will be expensive. I've explored methods used for speeding up learning.

## Diffusion models

Diffusion model is based on a Markov Chain of diffusion steps which slowly adds random noise to data and then learns to reverse the diffusion process to construct desired data samples from the noise.

- (Forward diffusion) For $t \in \overline{1,T}$ add Gaussian noise to the picture

  $$x_t = \sqrt{\overline{\alpha}_t} x_0 + \sqrt{1-\overline{\alpha}_t}\varepsilon, \quad \varepsilon \sim \mathcal{N}(0,1)$$

  $x_0$ – original picture, $\varepsilon$ – noise, $\overline{\alpha}_t$ – noise regularization coefficient for step $t$.
- (Reverse diffusion) For $t \in \overline{T,1}$ try to predict added noise and remove it

  $$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\overline{\alpha}_t}}\varepsilon_\theta(x_t,t)\right) + \sqrt{\beta_t}\varepsilon$$

  $x_t$ – noised picture, $\varepsilon_\theta(x_t,t)$ – predicted noise and $\beta_t = 1-\alpha_t, \prod \alpha_t = \overline{\alpha}_t$

## UNet

UNet is one of the core building blocks of Stable Diffusion. It has been applied to the reverse diffusion process as a noise predictor. It has the following architecture:

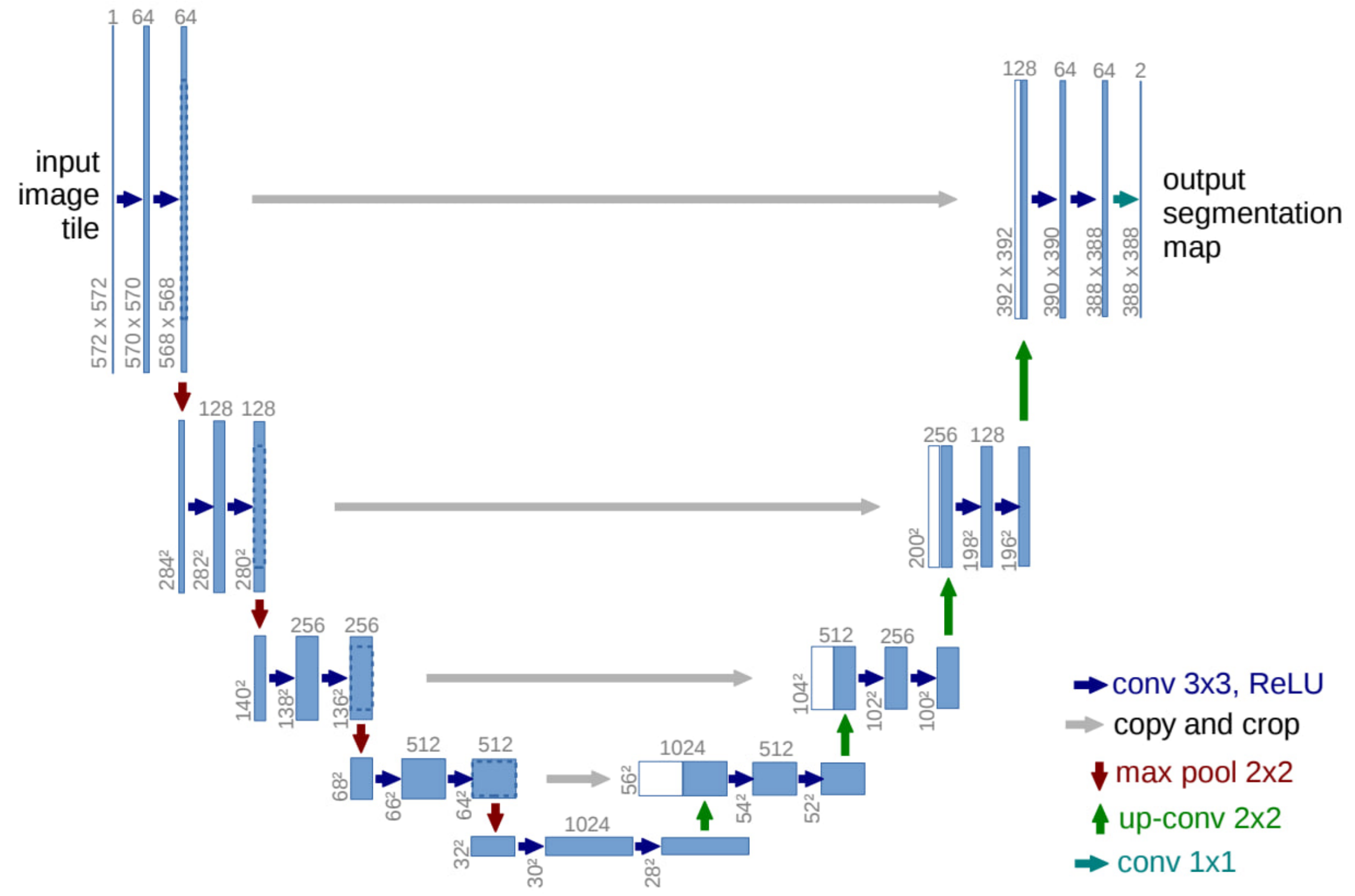


Figure 1: [1]

## CLIPText

CLIP (Contrastive Language-Image Pre-Training) is a neural network trained on a variety of (image, text) pairs. It can be instructed in natural language to predict the most relevant text snippet, given an image. This is used for creating text embeddings for Stable Diffusion [2].

## Stable diffusion

Stable diffusion is designed to solving conditioning problems, such as text2image. The key idea is to map image data in a latent space with an autoencoder (VAE) and to perform a conditioning forward and reverse diffusion with such networks as UNet and CLIP (image-text similarity model).

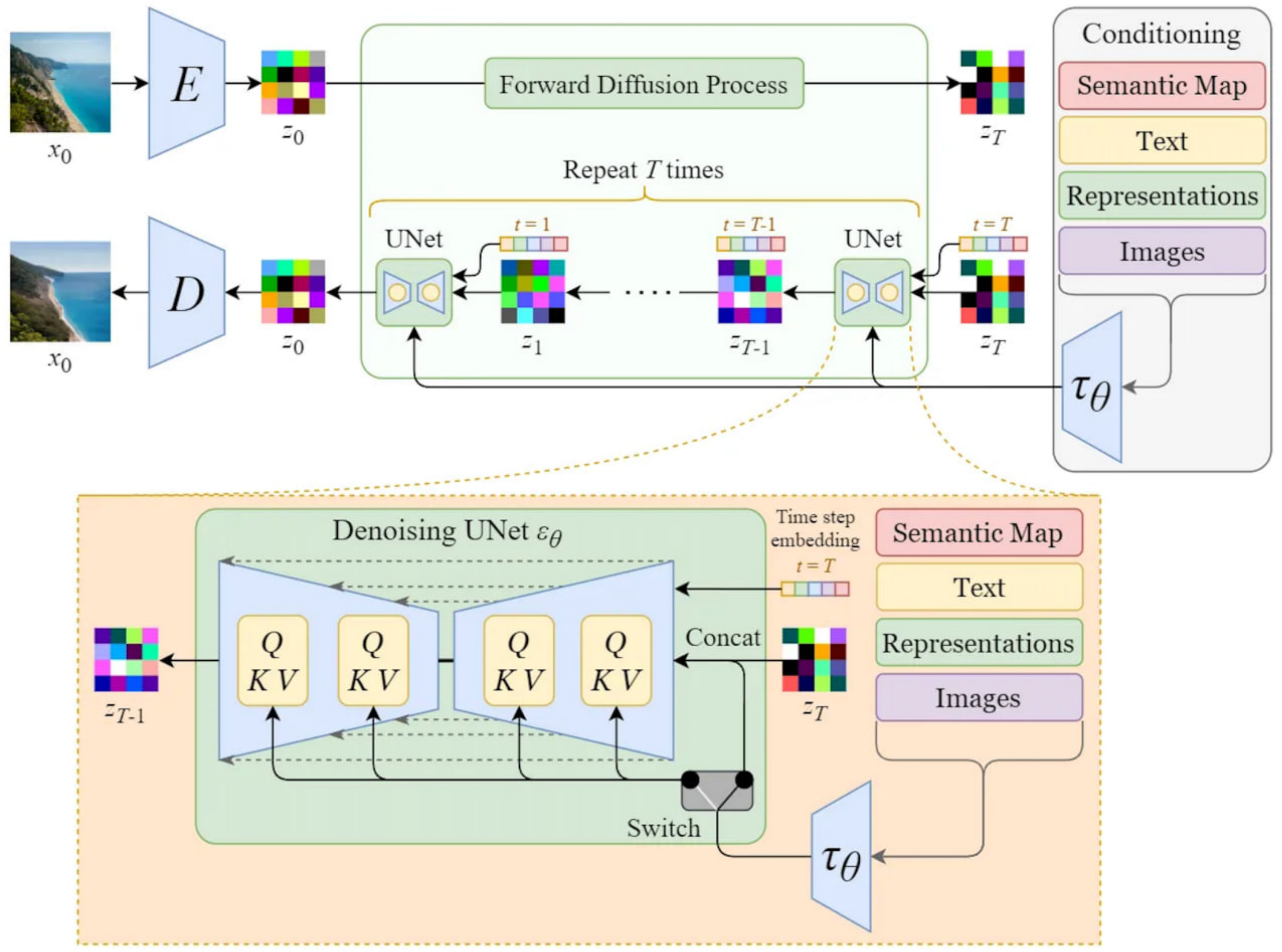


Figure 2: [3]

## Algorithm

### Loss function

In paper [4] loss function defined as

$$L_{\text{LDM}} = \mathbb{E}_{t,z_0,\varepsilon,y}[\|\varepsilon - \varepsilon_\theta(\sqrt{\overline{\alpha}_t}z_0 + \sqrt{1-\overline{\alpha}_t}\varepsilon, t, \tau_\theta(y))\|], z_0 = E(x_0)$$

### Training (DM)

```
repeat
    $x_0 \sim q(x_0)$
    $t \sim Uniform(\{1,\dots,T\})$
    $\varepsilon \sim \mathcal{N}(0,\mathbb{I})$
    Take gradient descent step on $\nabla_\theta\|\varepsilon - \varepsilon_\theta(\sqrt{\overline{\alpha}_t}x_0 + \sqrt{1-\overline{\alpha}_t}\varepsilon)\|^2$
until converged
```

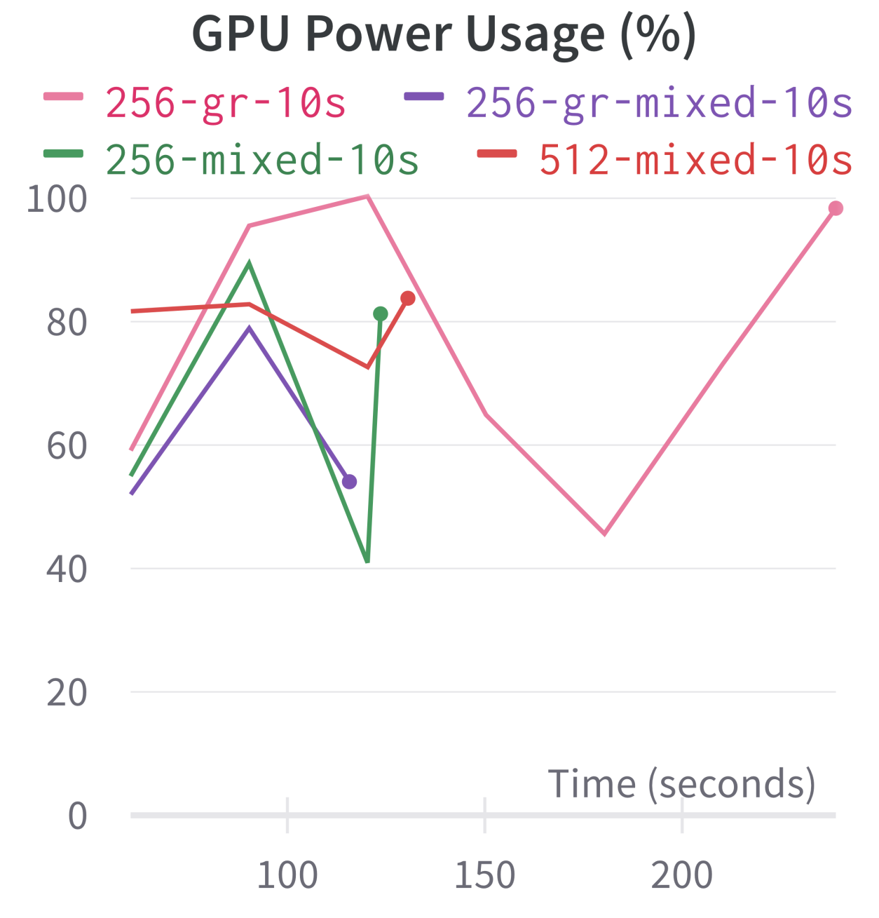
### Sampling (DM)

```
$x_T \sim \mathcal{N}(0,\mathbb{I})$
for $t = T,\dots,1$ do
    $z \sim \mathcal{N}(0,\mathbb{I})$ if $t > 1$, else $z = 0$
    $x_{t-1} = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\overline{\alpha}_t\varepsilon_\theta(x_t,t)}}) + \sigma_t z$
end for
return $x_0$
```

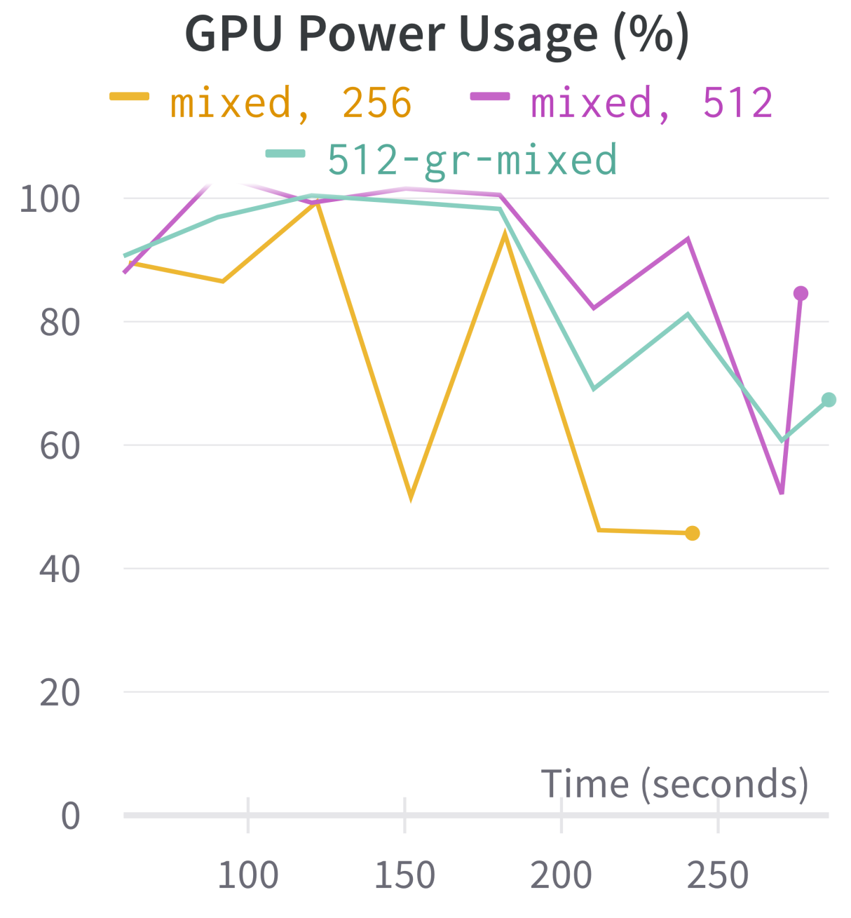
## Model and results

I used 🤗Accelerate and 🤗Diffusers frameworks [5] with fine-tuning ready to use configs and trained model with different configurations of parameters [6]: mixed_precision, resolution and gradient_checkpointing.

- Model: runwayml/stable-diffusion-v1-5 [4]
- GPU: Nvidia Tesla T4, 15 GB of RAM



(a)



(b)

Conclusions [7]:

- mixed_precision is essential (reduces training time and GPU usage by $50\%$)
- with gragient_checkpoining training requires less GPU power ($8\%$ gain)
- image downscaling speeds up training (reduces training time by $10\%$)
- with image downscaling training requires less GPU power

## References

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[2] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[3] J Alammar. The illustrated stable diffusion, 2022.

[4] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022.

[5] A. Lozhkov P. Cuenca N. Lambert K. Rasul M. Davaadorj T. Wolf P. von Platen, S. Patil. Diffusers: State-of-the-art diffusion models. `https://github.com/huggingface/diffusers`, 2022.

[6] Code. Google colab.

[7] Wandb.