One-step Diffusion with Distribution Matching Distillation

Maxim Bekoev, Ulyana Klyuchnikova, Ekaterina Grishina

Motivation

Standard DDPM:

- needs many timesteps
- has long generation time
- consumes much resources
- is too slow for interactive generation

Other 1-Step alternatives significantly lose in performance

Solution: Diffusion Distillation

1. choose pre-trained diffusion model μbase(xt, t)
2. pre-compute the denoising output
3. initialize three same models with pre-trained weights from μbase: netG, μreal, μfake
4. learn real and fake distributions from pretrained DDPM

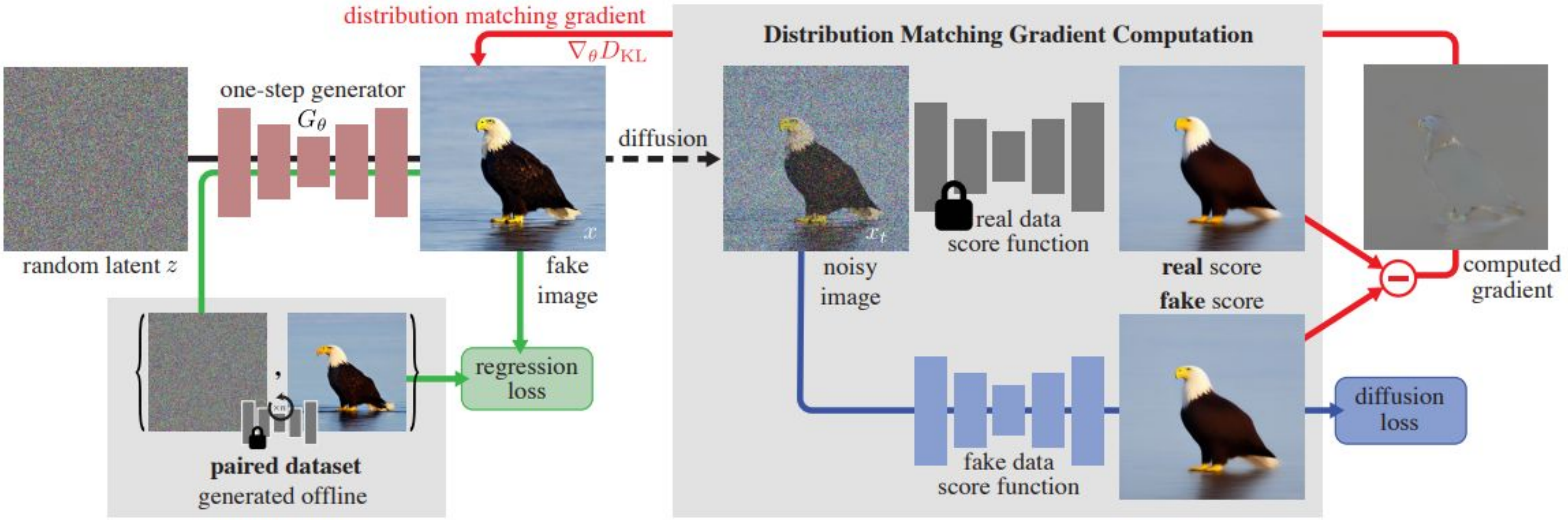


Figure 2. **Method overview.** We train one-step generator G_θ to map random noise z into a realistic image. To match the multi-step sampling outputs of the diffusion model, we pre-compute a collection of noise–image pairs, and occasionally load the noise from the collection and enforce LPIPS [85] regression loss between our one-step generator and the diffusion output. Furthermore, we provide **distribution matching gradient** $\nabla_\theta D_{KL}$ to the fake image to enhance realism. We inject a random amount of noise to the fake image and pass it to two diffusion models, one pretrained on the real data and the other continually trained on the fake images with a diffusion loss, to obtain its denoised versions. The denoising scores (visualized as mean prediction in the plot) indicate directions to make the images more realistic or fake. The difference between the two represents the direction toward more realism and less fakeness and is backpropagated to the one-step generator.

Distribution Matching Loss

$$D_{KL}\left(p_{\text{fake}} \parallel p_{\text{real}}\right) = \mathop{\mathbb{E}}_{\substack{z \sim \mathcal{N}(0;\mathbf{I}) \\ x = G_\theta(z)}} -\left(\log\ p_{\text{real}}(x) - \log\ p_{\text{fake}}(x)\right)$$

$$\nabla_\theta D_{KL} = \mathop{\mathbb{E}}_{\substack{z \sim \mathcal{N}(0;\mathbf{I}) \\ x = G_\theta(z)}} \left[-\left(\underline{s_{\text{real}}(x)} - \underline{s_{\text{fake}}(x)}\right) \nabla_\theta G_\theta(z) \right]$$

Real score

Dynamically-learned fake score

Training process

for epoch in epochs:

for noise, generated_image in loader:

- Generate images by netG from noise
- Update netG by **KL loss** + **Reg loss** (**lpips) * α**
- Update mu_fake by **denoising loss**

$$\mathcal{L}^{\phi}_{\text{denoise}} = ||\mu^{\phi}_{\text{fake}}(x_t, t) - x_0||^2_2$$

Drawbacks of diffusion distillation

- Need to have 3 models simultaneously, 2 of them are trained
- Need to generate dataset of noise-image pairs
- We have 3 losses - need to adjust hyperparameters

Although the method achieves high generation quality, it consumes considerable amount of resources and requires a wise choice of learning rate and hyperparameters.

Tasks

Ekaterina:

- experimented with conditional ddpm trained on MNIST

Maxim:

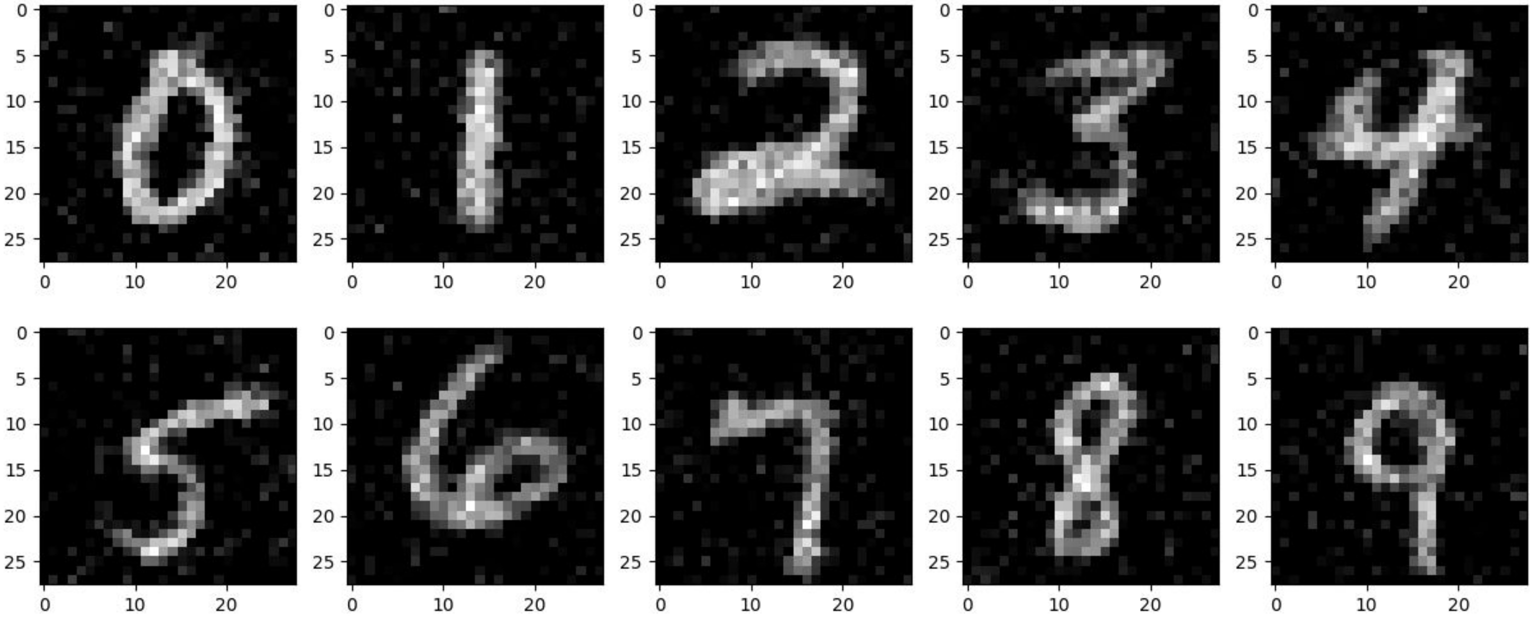
- experimented with ddpm trained on CelebA

Ulyana:

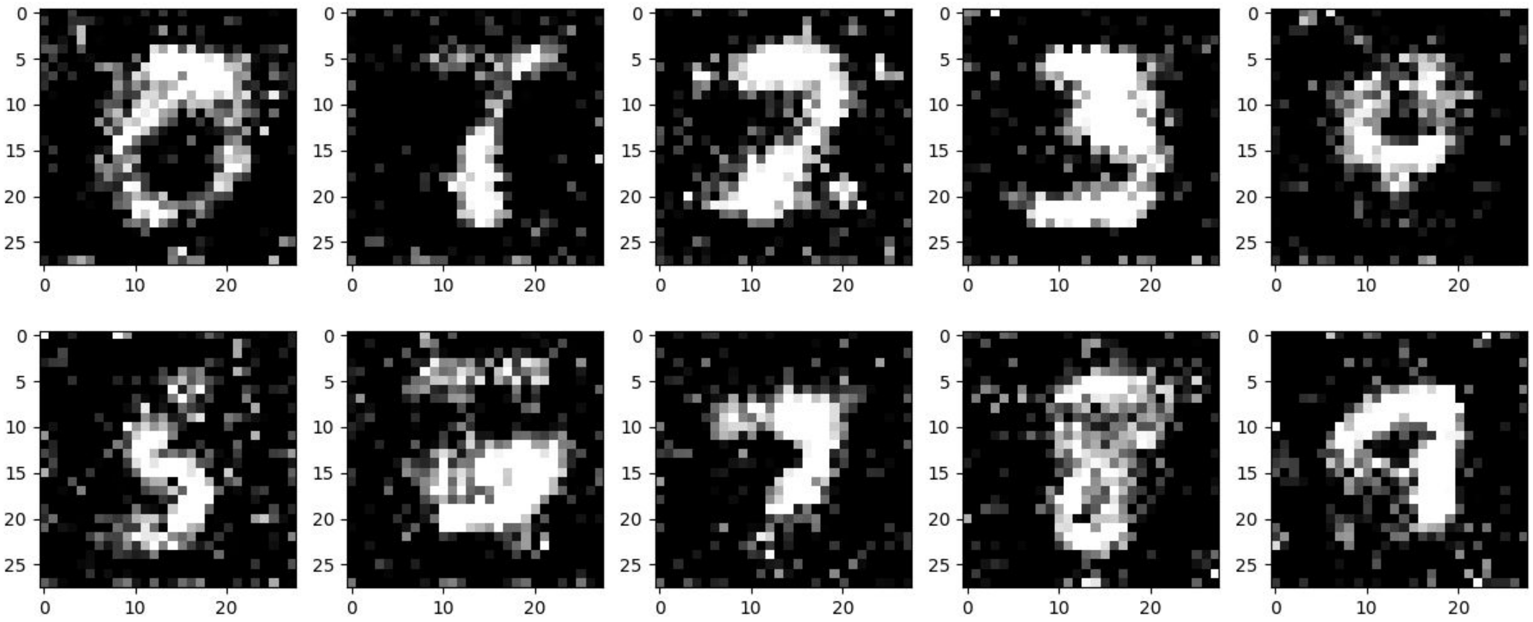
- experimented with net from diffusers pretrained on CIFAR10

Parameter	MNIST	CelebA	CIFAR10
Total parameters (μbase)	6585601	426691	35746307
Pretrained model source	https://github.com/TeaPearce/Conditional_Diffusion_MNIST	our model	google/ddpm-cifar10-32
Number pre-computed pairs	12800	12800	4864
Learning Rate μreal	5e-6	5e-6	1e-6
Learning Rate μfake	5e-6	5e-6	1e-6
Batch Size	64	64	64
Number of Epochs	10	10	20
Number of steps (total)	2000	2000	1520
Betas	1e-4, 1e-2	1e-4, 2e-2	1e-4, 1e-2

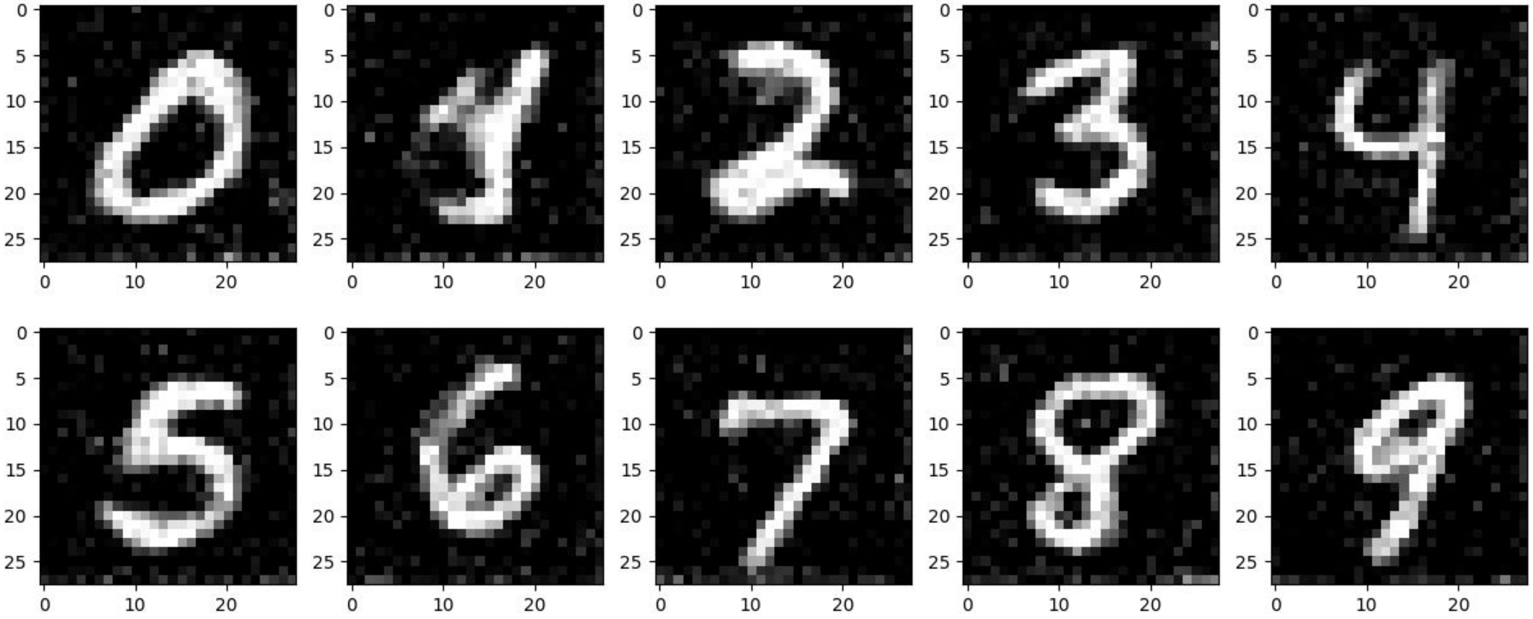
Usual 400-step generation



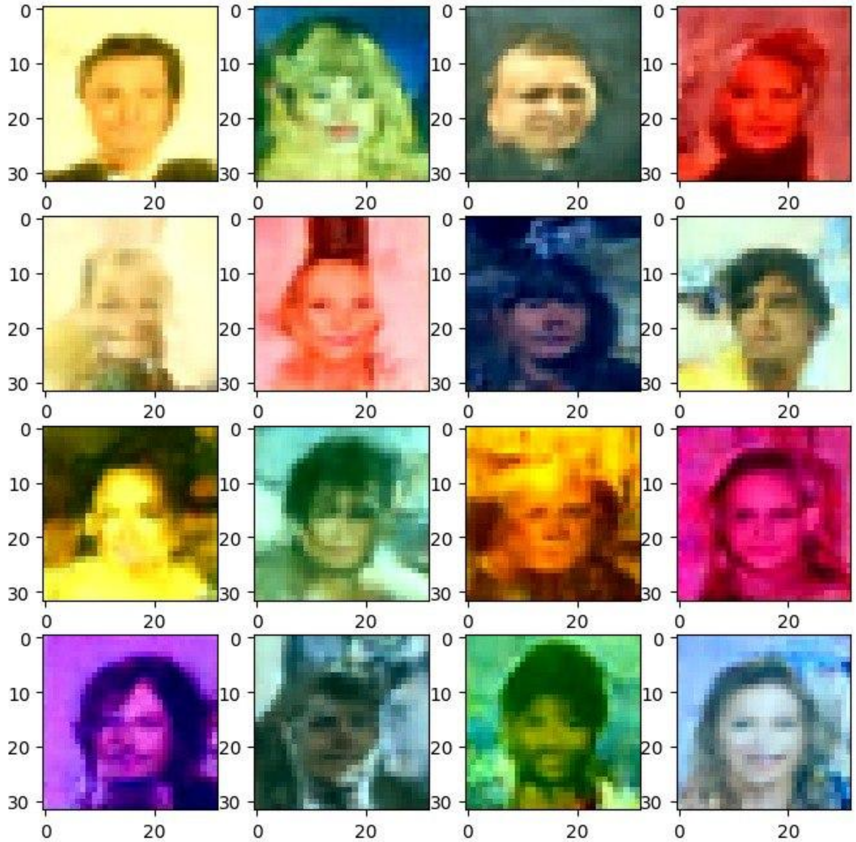
1-step generation before distillation



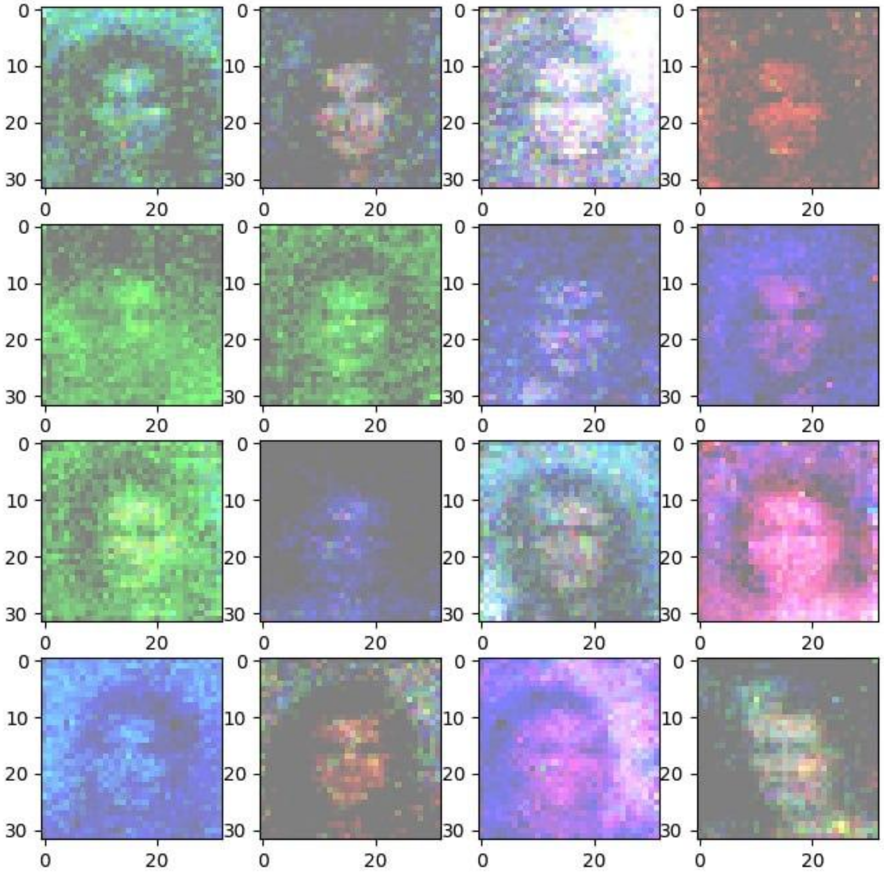
1-step generation after distillation



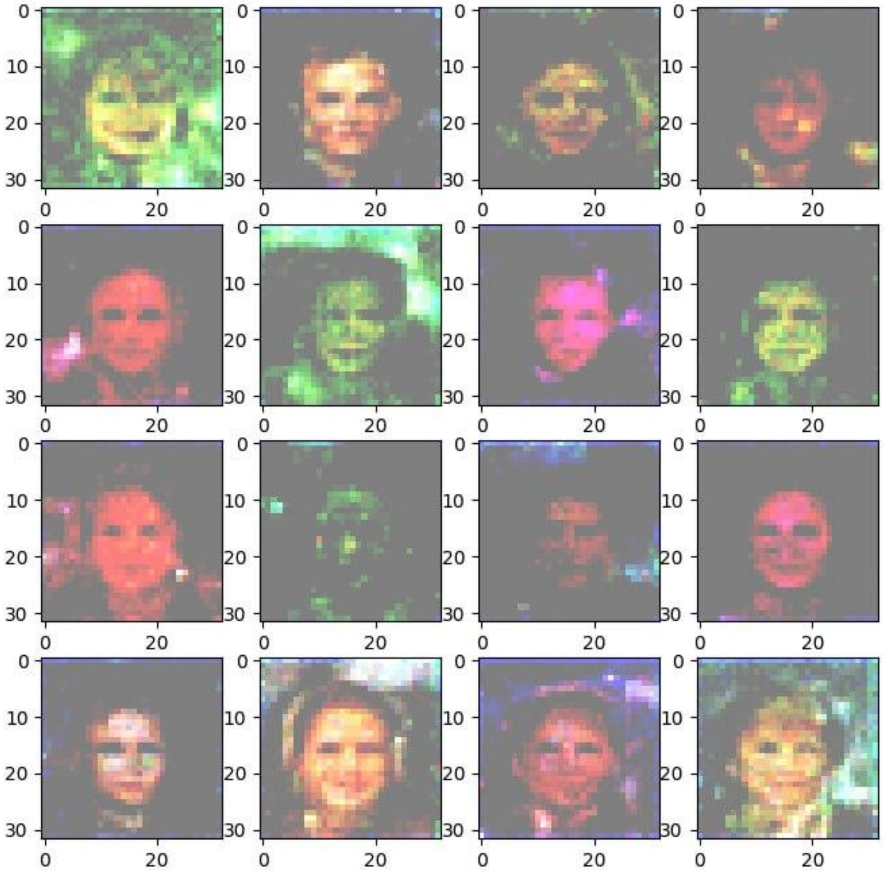
Usual 1000-step generation

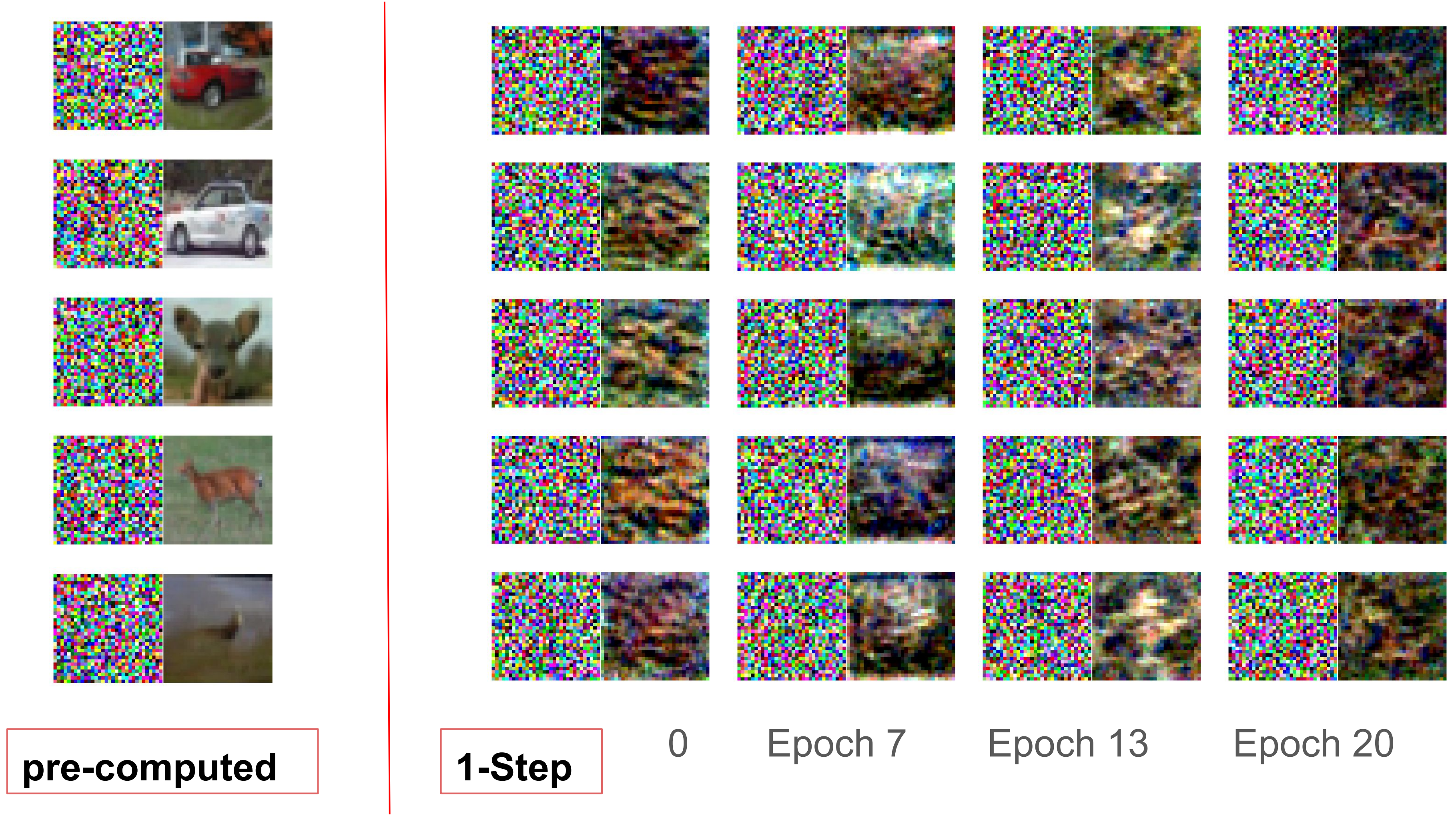


before_distillation

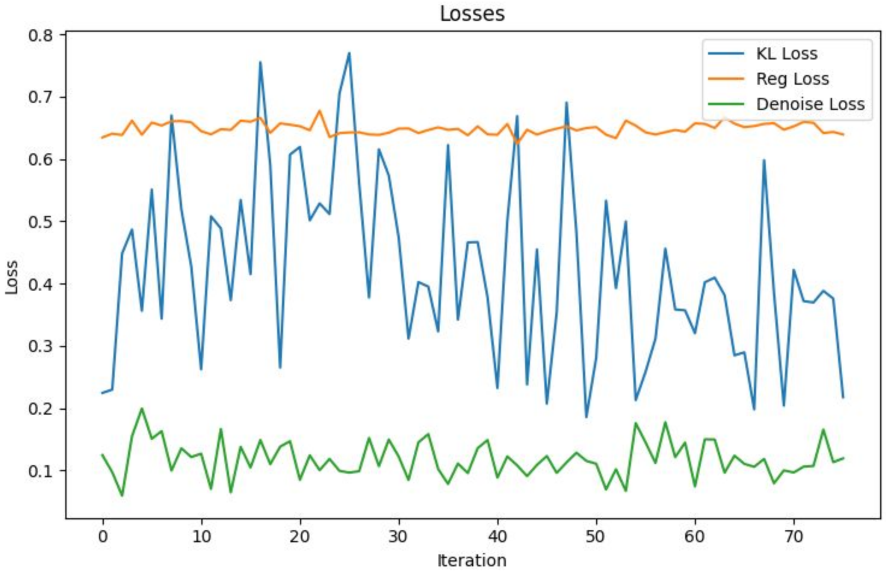


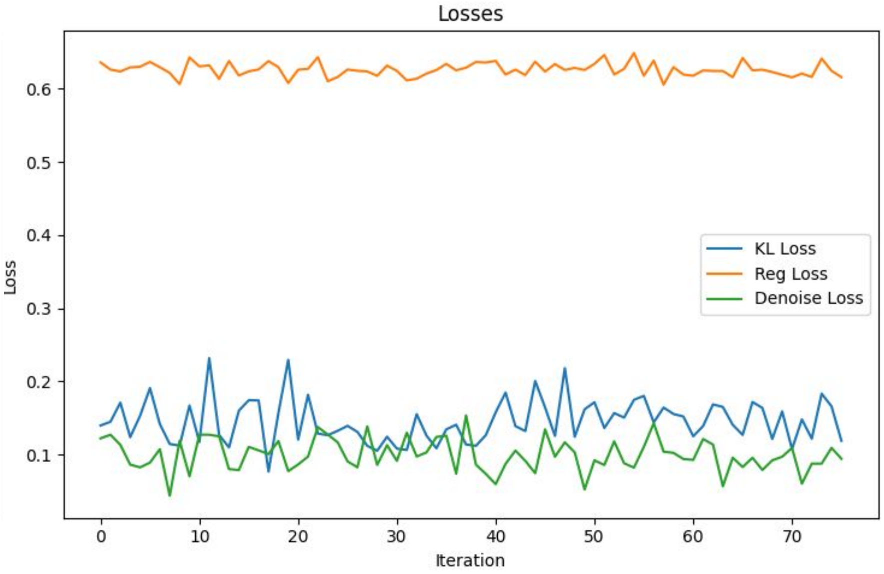
after_distillation




pre-computed
1-Step
0
Epoch 7
Epoch 13
Epoch 20

Losses on Epoch 0 vs Epoch 13

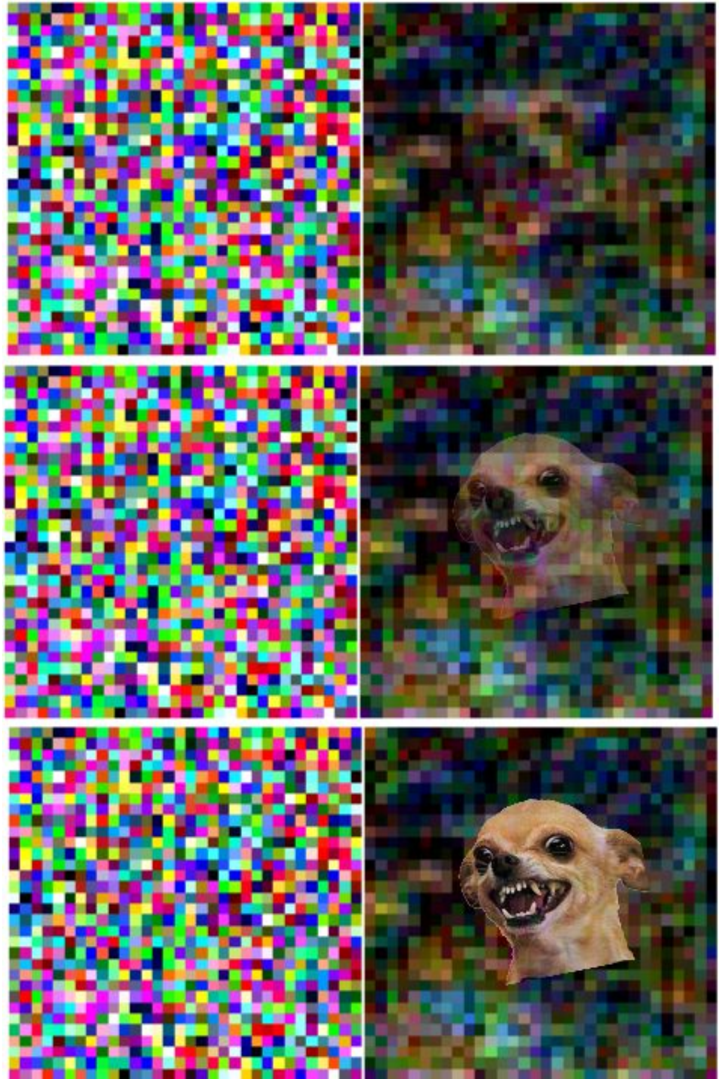




CIFAR10

Parameter	Value
Total parameters (μbase)	35746307
Pretrained model source	google/ddpm-cifar10-32
Number pre-computed pairs	4864
Learning Rate μreal	1e-6
Learning Rate μfake	1e-6
Batch Size	64
Number of Epochs	20
Number of steps (total)	1520

Thanks for your attention!



Sources

Yin, T., Gharbi, M., Zhang, R., Shechtman, E., Durand, F., Freeman, W. T., & Park, T. (2023). One-step Diffusion with Distribution Matching Distillation. https://arxiv.org/pdf/2311.18828.pdf