Denoising Diffusion Probabilistic Models

Jonathan Ho, Ajay Jain, and Pieter Abbeel

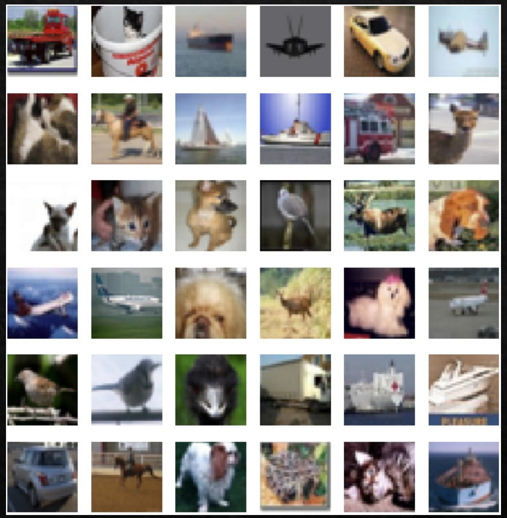
Presented by: Xuansheng Wu & Daniel Redder

Contents

Ho, Jonathan et. al., DDPM, NIPS 2022.

Background : Unconditional Image Synthesis

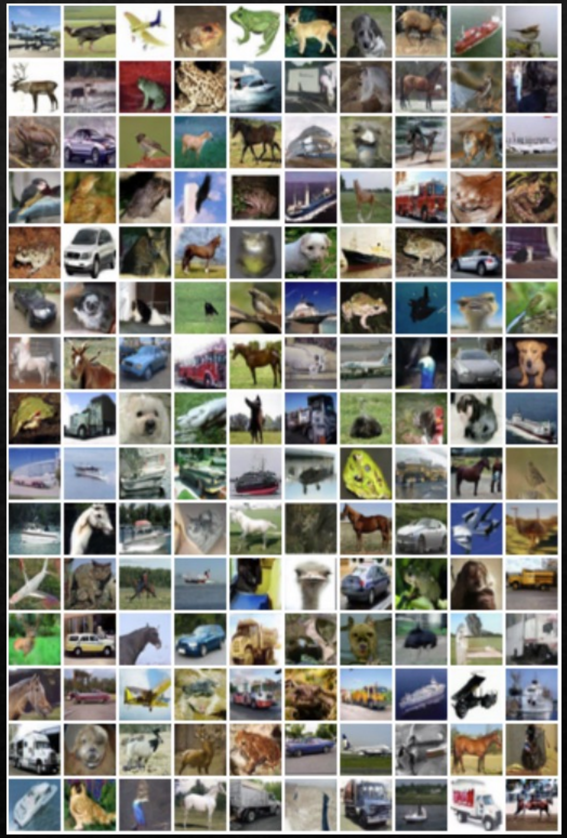
Modeling P(X), an image distribution X



CIFAR 10

Generator

- Variable Auto Encoders
- Generative Adversarial Networks
- Normalizing Flows



Generated Samples

Ho, Jonathan et. al., DDPM, NIPS 2022.

Background : **Diffusion Process and Time Reversal**

Diffusion destroys structures, and reverts things to a "stable state"

Data Distribution

Time Flies →

← Just go back in time

Uniform Distribution

Ho, Jonathan et. al., DDPM, NIPS 2022.

High Level Diffusion Explained: Markov Chains

- Model physical diffusion as a markov chain

Let the "most stable state" be gaussian noise

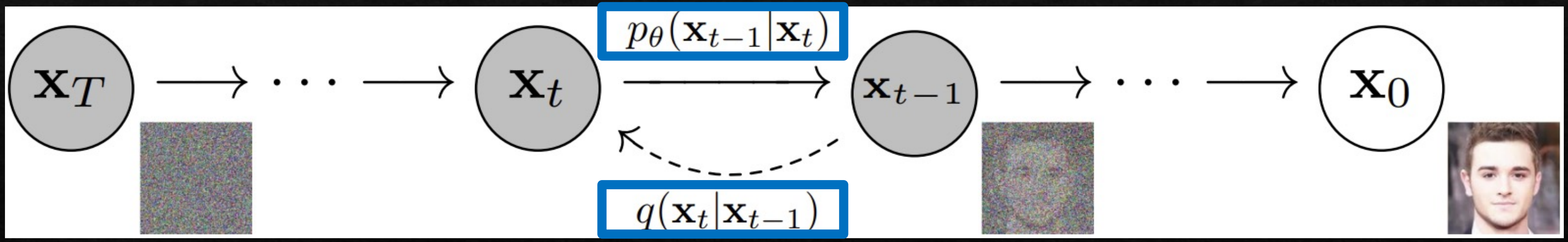
- x_0 is the original image
- x_t is the image after t additions of noise
- x_T the image of pure noise $\approx$ 1000

Forward Diffusion Adds Noise

Backward Diffusion Removes Noise!

First Order Markov Assumption: the current time step is only dependent on the previous

Backward Diffusion



Forward Diffusion

Ho, Jonathan et. al., DDPM, NIPS 2022.

High Level Diffusion Explained: Model Passes

- Forward noising pass (training)
  - Training involves predicting $\epsilon \mid x_t, t$ notated $\epsilon_\theta(x_t, t)$
  - x_t is created by manual noise application

Slows Inference

- Backward denoising pass (From T → 0)

$$p_\theta(x_{t-1}|x_t) = N\left(X_{t-1};\ \mu_\theta(x_t, \mathrm{t}),\ \textstyle\sum_\theta (x_t, \mathrm{t})\right)$$

- Calculated with β_t & ϵ_θ
- Formerly predicted

- Replaced by a constant in DbG & DPM (below)
- Separately trainable – (importance decreases with T)

Ho, Jonathan et. al., DDPM, NIPS 2022.

O. Avrahami, D. Lischinski, and O. Fried, "Blended diffusion for text-driven editing of natural images," 2021.

J. Ho, C. Saharia, W. Chan, D. J. Fleet, M. Norouzi, and T. Salimans, "Cascaded diffusion models for high fidelity image generation," 2021.

Diffusion Forward Pass

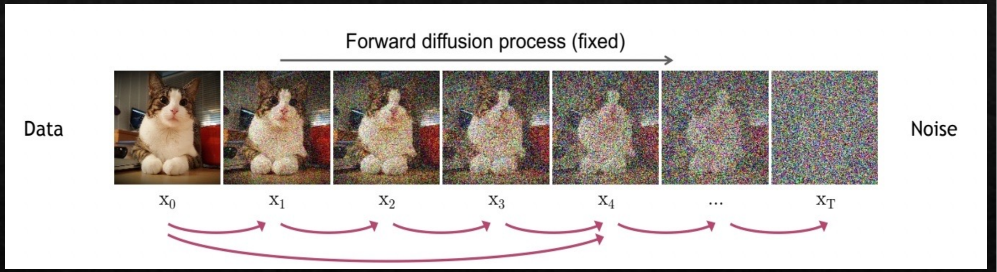
- Given a data point $\mathbf{x}_0$ at the time step $t = 0$, the diffusion process at each step t can be formalized as:

$$q(x_t|x_{t-1}) = \mathcal{N}(\sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I}),$$

where $\{\beta_t\}_{t=1}^{T}$ is the variance of each step.

- Reparameterization Trick: Let $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$, we have:

$$\begin{aligned}
\mathbf{x}_t &= \sqrt{\alpha_t}\mathbf{x}_{t-1} + \sqrt{1-\alpha_t}\epsilon_{t-1} \\
&= \sqrt{\alpha_t}(\sqrt{\alpha_{t-1}}\mathbf{x}_{t-2} + \sqrt{1-\alpha_{t-1}}\epsilon_{t-2}) + \sqrt{1-\alpha_t}\epsilon_{t-1} \\
&= \sqrt{\alpha_t\alpha_{t-1}}\mathbf{x}_{t-2} + \sqrt{\sqrt{\alpha_t - \alpha_t\alpha_{t-1}}^2 + \sqrt{1-\alpha_t}^2}\,\bar{\epsilon}_{t-2} \\
&= \sqrt{\alpha_t\alpha_{t-1}}\mathbf{x}_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\,\bar{\epsilon}_{t-2} \\
&= \dots \\
&= \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon \text{ ;where } \epsilon_{t-1}, \epsilon_{t-2}, \dots \sim \mathcal{N}(\mathbf{0}, \mathbf{I})
\end{aligned}$$



- Consider the entire diffusion process as a Markov Chain, we have:

$$q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}).$$

Ho, Jonathan et. al., DDPM, NIPS 2022.

Diffusion Backward Pass

- By conditioning on $\mathbf{x}_0$, the real distribution $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ can be written as:

$$q(x_{t-1}|x_t, x_0) = \mathcal{N}(\tilde{\mu}(x_t, x_0), \tilde{\beta}_t \mathbf{I}).$$

- Using Bayes' rule, we have:

$$\begin{aligned}
q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) &= q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{\boxed{q(\mathbf{x}_t|\mathbf{x}_0)}} \\
&\propto \exp\Big(-\frac{1}{2}\big(\frac{(\mathbf{x}_t - \sqrt{\alpha_t}\mathbf{x}_{t-1})^2}{\beta_t} + \frac{(\mathbf{x}_{t-1} - \sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0)^2}{1-\bar{\alpha}_{t-1}} - \frac{(\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0)^2}{1-\bar{\alpha}_t}\big)\Big) \\
&= \exp\Big(-\frac{1}{2}\big(\frac{\mathbf{x}_t^2 - 2\sqrt{\alpha_t}\mathbf{x}_t\mathbf{x}_{t-1} + \alpha_t\mathbf{x}_{t-1}^2}{\beta_t} + \frac{\mathbf{x}_{t-1}^2 - 2\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0\mathbf{x}_{t-1} + \bar{\alpha}_{t-1}\mathbf{x}_0^2}{1-\bar{\alpha}_{t-1}} - \frac{(\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0)^2}{1-\bar{\alpha}_t}\big)\Big) \\
&= \exp\Big(-\frac{1}{2}\big((\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha}_{t-1}})\mathbf{x}_{t-1}^2 - (\frac{2\sqrt{\alpha_t}}{\beta_t}\mathbf{x}_t + \frac{2\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}\mathbf{x}_0)\mathbf{x}_{t-1} + \boxed{C(\mathbf{x}_t, \mathbf{x}_0)}\big)\Big)
\end{aligned}$$

where $C(\mathbf{x}_t, \mathbf{x}_0)$ is some function not involving $\mathbf{x}_{t-1}$

- Following the standard Gaussian density function, we have:

$$\begin{aligned}
\tilde{\beta}_t &= 1/(\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha}_{t-1}}) = 1/(\frac{\alpha_t - \bar{\alpha}_t + \beta_t}{\beta_t(1-\bar{\alpha}_{t-1})}) = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t} \cdot \beta_t \\
\tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0) &= (\frac{\sqrt{\alpha_t}}{\beta_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}\mathbf{x}_0)/(\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha}_{t-1}}) \\
&= (\frac{\sqrt{\alpha_t}}{\beta_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}\mathbf{x}_0)\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t} \cdot \beta_t \\
&= \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0 = \frac{1}{\sqrt{\alpha_t}} * (\mathrm{x}_t - \frac{1-\alpha_t}{\sqrt{1-\alpha_t}} * \epsilon_t)
\end{aligned}$$

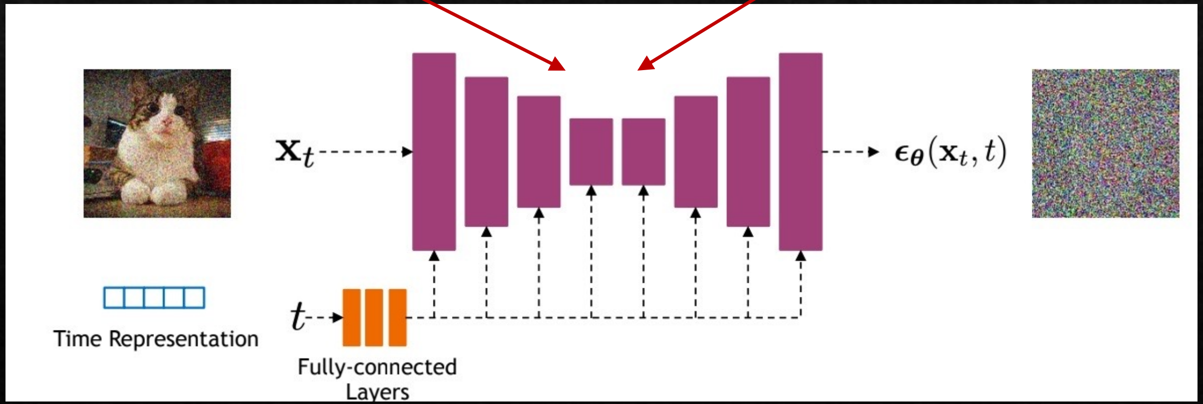
The Diffusion Algorithm & Structure

Algorithm 1 Training

1: **repeat**
2: $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
3: $t \sim \mathrm{Uniform}(\{1, \dots, T\})$
4: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
5: Take gradient descent step on
$$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$$
6: **until** converged

Algorithm 2 Sampling

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = T, \dots, 1$ **do**
3: $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
4: $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right) + \sigma_t \mathbf{z}$
5: **end for**
6: **return** $\mathbf{x}_0$



Ho, Jonathan et. al., DDPM, NIPS 2022.

Metrics: Cifar-10

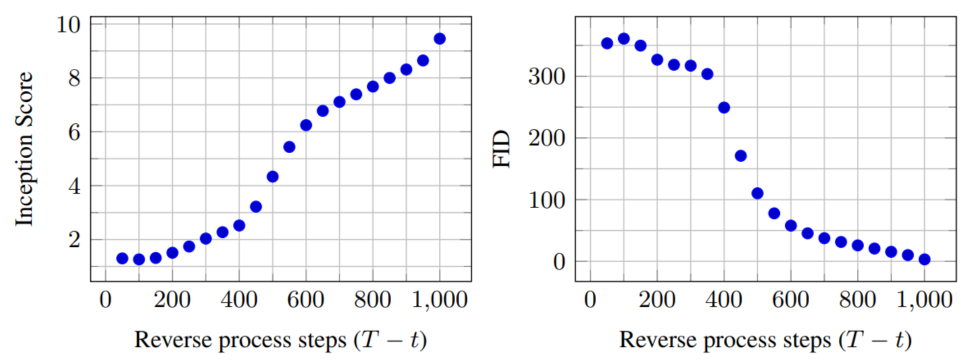


Figure 10: Unconditional CIFAR10 progressive sampling quality over time

Progressive Image Generation
(conditional)

Table 1: CIFAR10 results. NLL measured in bits/dim.

Model	IS	FID	NLL Test (Train)
Conditional			
EBM [11]	8.30	37.9	
JEM [17]	8.76	38.4	
BigGAN [3]	9.22	14.73	
StyleGAN2 + ADA (v1) [29]	**10.06**	**2.67**	
Unconditional			
Diffusion (original) [53]			≤ 5.40
Gated PixelCNN [59]	4.60	65.93	3.03 (2.90)
Sparse Transformer [7]			**2.80**
PixelIQN [43]	5.29	49.46	
EBM [11]	6.78	38.2	
NCSNv2 [56]		31.75	
NCSN [55]	8.87 ± 0.12	25.32	
SNGAN [39]	8.22 ± 0.05	21.7	
SNGAN-DDLS [4]	9.09 ± 0.10	15.42	
StyleGAN2 + ADA (v1) [29]	**9.74** $\pm$ 0.05	3.26	
Ours (L_{simple})	9.46 ± 0.11	**3.17**	≤ 3.75 (3.72)

Ho, Jonathan et. al., DDPM, NIPS 2022.

Metrics: LSUN

LSUN: dataset of classes of room images

Table 3: FID scores for LSUN 256 × 256 datasets

Model	LSUN Bedroom	LSUN Church	LSUN Cat
ProgressiveGAN [27]	8.34	6.42	37.52
StyleGAN [28]	**2.65**	4.21*	8.53*
StyleGAN2 [30]	-	**3.86**	**6.93**
Ours (L_{simple})	6.36	7.89	19.75
Ours (L_{simple}, large)	4.90	-	-

Unconditioned Image Generation

Figure 17: LSUN Bedroom generated samples, large model. FID=4.90

Ho, Jonathan et. al., DDPM, NIPS 2022.

“Metric”: Latent Mixing

What if we mix our inputted images?

- Right source is mixed at a ratio of λ
- Rec. is unclear (they do not define it)
- They do not add noise to these images. It is unclear if they used the CelebA model here

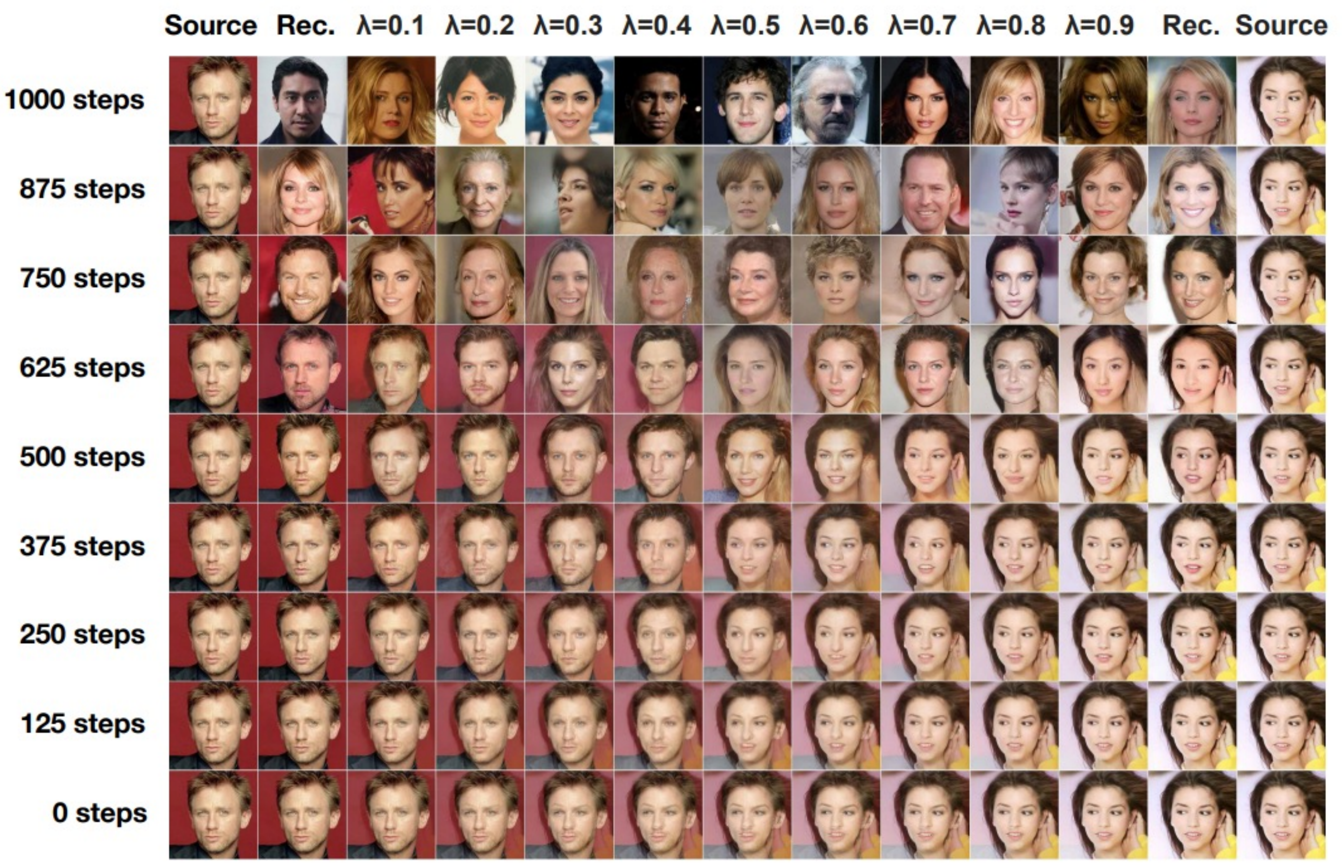


Figure 9: Coarse-to-fine interpolations that vary the number of diffusion steps prior to latent mixing.

Ho, Jonathan et. al., DDPM, NIPS 2022.

Some Extensions

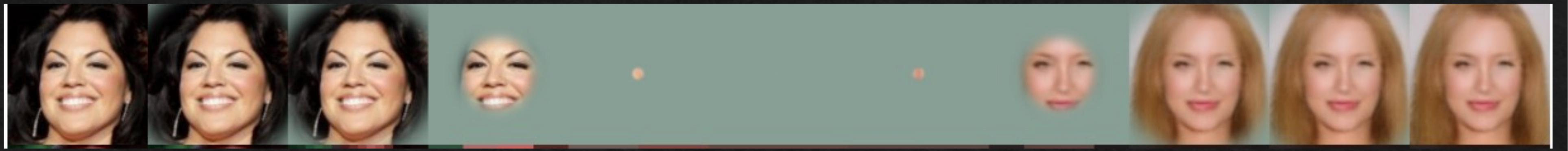
O. Avrahami, D. Lischinski, and O. Fried, "Blended diffusion for text-driven editing of natural images," 2021.

J. Ho, C. Saharia, W. Chan, D. J. Fleet, M. Norouzi, and T. Salimans, "Cascaded diffusion models for high fidelity image generation," 2021.

A. Bansal, E. Borgnia, H.-M. Chu, J. S. Li, H. Kazemi, F. Huang, M. Goldblum, J. Geiping, and T. Goldstein, "Cold diffusion: Inverting arbitrary image transforms without noise," 2022.

E. Aiello, D. Valsesia, and E. Magli, "Fast inference in denoising diffusion models via mmd finetuning," *arXiv preprint arXiv:2301.07969*, 2023.

Cold Diffusion: Non gaussian noise does not affect inference



Blended Diffusion: Text conditioned inpainting

input+mask "big mountain" "big wall" "New York City"

CONfusion: explainable AI through confidence intervals

Input Lower Bound Upper Bound Ground Truth Error Interval Sizes

MMD Finetuning: Fast Diffusion Inference through Approximation

Conclusion