Carnegie Mellon University

Lecture 7: Guidance & Controllable Generation

Yutong (Kelly) He

10-799 Diffusion & Flow Matching, Jan 29th, 2026

Modal aws

Quiz time!

10 minutes

Closed-book

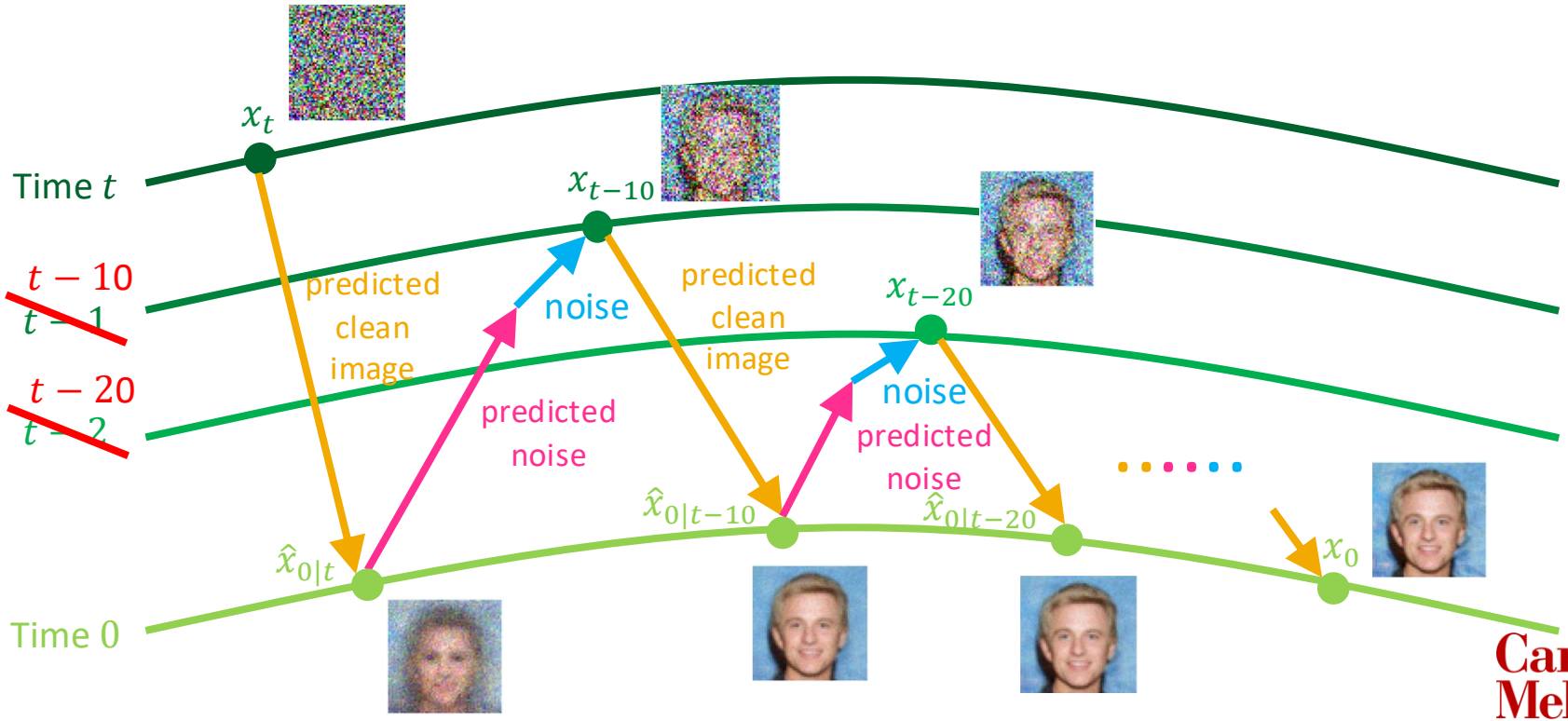
On Gradescope!!!

If you don’t want to stay for the lecture, feel free to leave after submitting your quiz!

Housekeeping Announcements

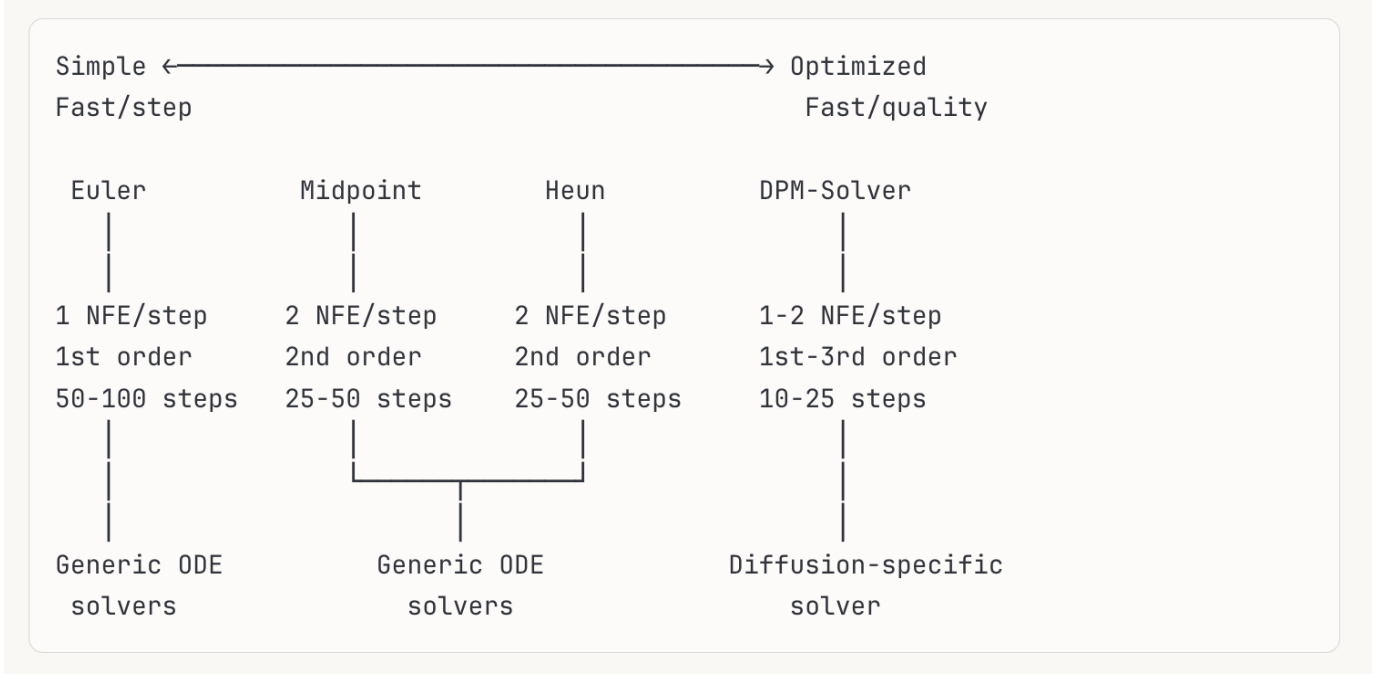
- Homework 2 is out! https://kellyyutonghe.github.io/10799S26/homework/
    - Due date: 2/3 Tue, Late Due date: 2/5 Thur
    - Training models takes time! Start early!
    - Maybe change in due date? Vote in Discord!
- **Next class** we will have **Max Simchowitz** come for a guest lecture/panel! Come to class **in person** if you want to ask him questions about diffusion/flow for robotics, control & decision making

Last time we learned DDIM

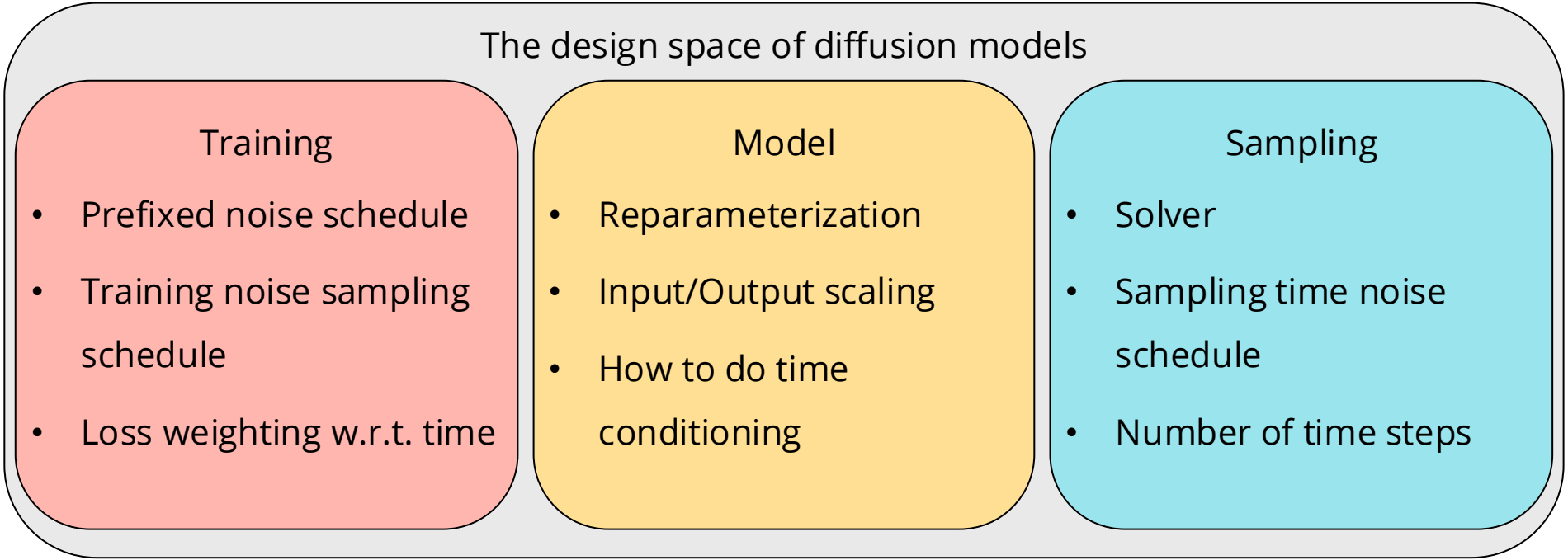


Carnegie Mellon University

Solver comparison (from Claude)

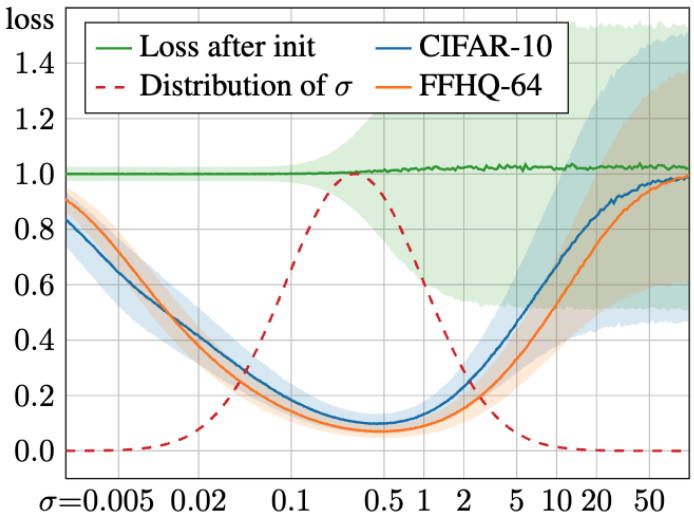


The design space of diffusion models



Clarification on training noise distribution

Just sampling the **time steps that are more effective** during training time!



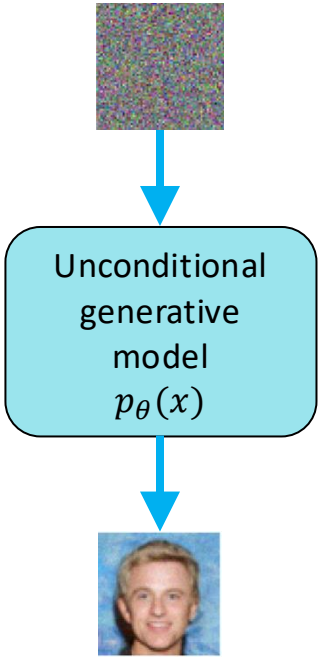
Carnegie Mellon University

Karras et al. "Elucidating the Design Space of Diffusion-Based Generative Models". NeurIPS 2022. https://arxiv.org/pdf/2206.00364

Are diffusion models perfect now?

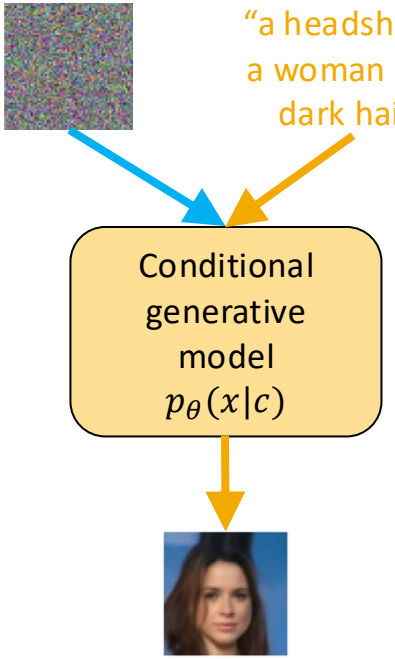
How do normal people do conditional generation

Unconditional generation:



But the unconditional model already has a lot of good knowledge on how to generate human faces in general…

Conditional generation:



You'll have to **train a new model** for each type of condition and can't reuse the unconditional model!

Carnegie Mellon University

Remember the No.1 rule in probability theory

We know that diffusion models model the score function

$$\nabla_{x_t} \log p_\theta(x_t, t)$$

By Bayes' Theorem, we know that

$$p(x|c) = \frac{p(x,c)}{p(c)} = \frac{p(c|x)p(x)}{p(c)} \propto p(c|x)p(x)$$

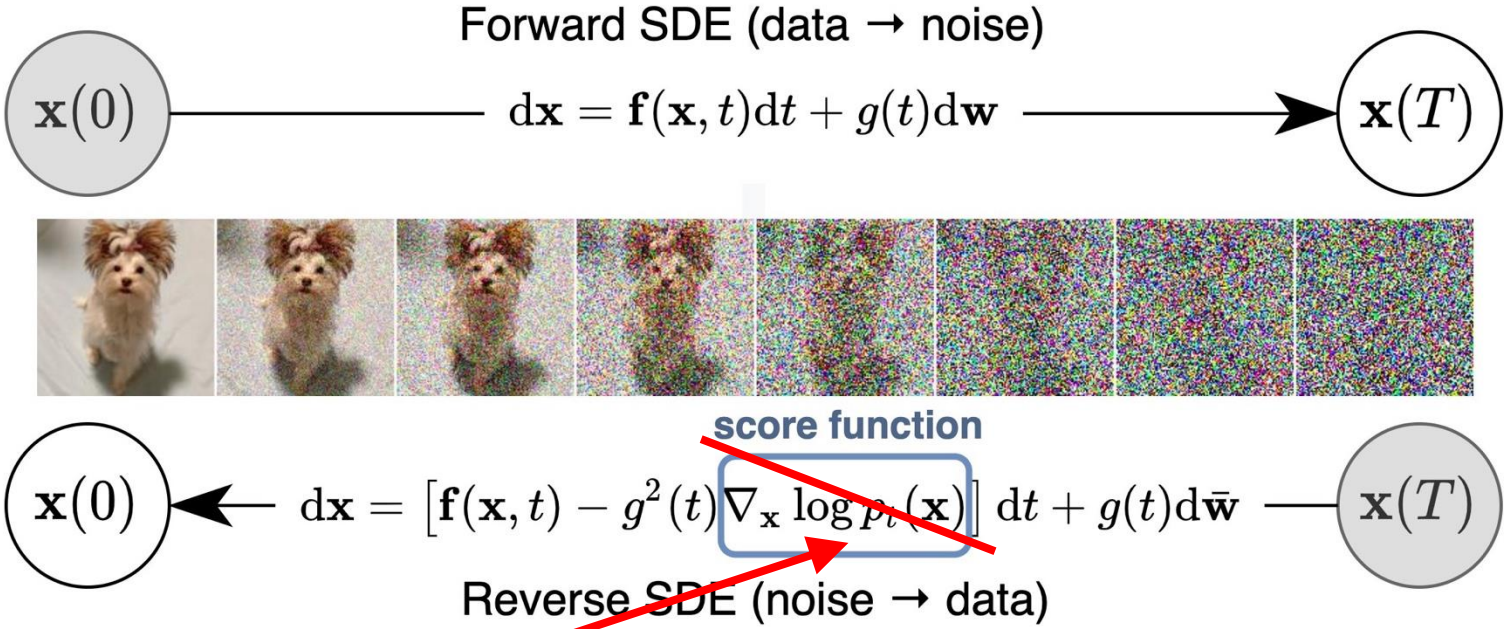
$$\nabla_{x_t} \log p_\theta(x_t, t|c) = \nabla_{x_t} \log p_\theta(c|x_t, t) + \nabla_{x_t} \log p_\theta(x_t, t)$$

Pre-trained unconditional diffusion model

A discriminative model!
(e.g. a classifier, a CLIP model, etc)

Carnegie Mellon University

Classifier guidance diffusion



Conditional score:

$$\nabla_{x_t} \log p_\theta(x_t, t|c) = \nabla_{x_t} \log p_\theta(c|x_t, t) + \nabla_{x_t} \log p_\theta(x_t, t)$$

Dhariwal & Nichol. "Diffusion Models Beat GANs on Image Synthesis". NeurIPS 2021. https://arxiv.org/pdf/2105.05233

Song et al. "Score-Based Generative Modeling through Stochastic Differential Equations". ICLR 2021. https://openreview.net/pdf?id=PxTIG12RRHS

Classifier guidance diffusion v.s. normies

Unconditional diffusion:

Unconditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t)$

Conditional diffusion:

"a headshot of a woman with dark hair"

Conditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t|c)$

Classifier guidance diffusion:

"a headshot of a woman with dark hair"

Unconditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t)$

Classifier $p_\theta(c|x_t, t)$

Carnegie Mellon University

Classifier guidance diffusion/Conditional score SDE

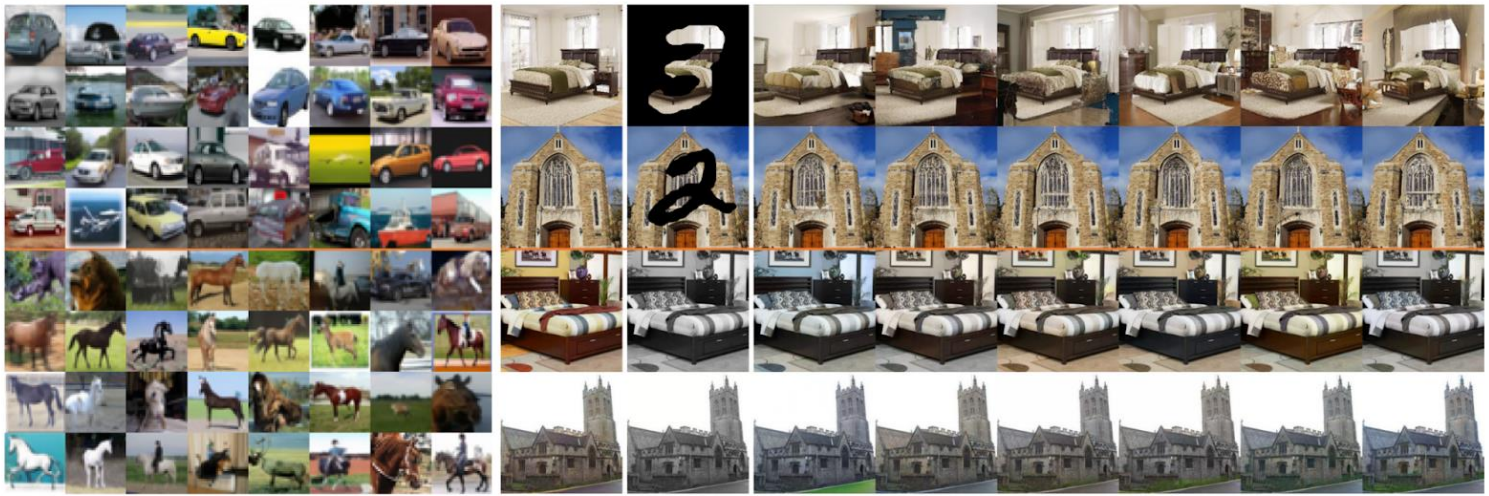


Figure 4: *Left*: Class-conditional samples on 32 × 32 CIFAR-10. Top four rows are automobiles and bottom four rows are horses. *Right:* Inpainting (top two rows) and colorization (bottom two rows) results on 256 × 256 LSUN. First column is the original image, second column is the masked/gray-scale image, remaining columns are sampled image completions or colorizations.

Carnegie Mellon University

Song et al. "Score-Based Generative Modeling through Stochastic Differential Equations". ICLR 2021. https://openreview.net/pdf?id=PxTIG12RRHS

Remember the No.1 rule in probability theory

We know that diffusion models model the score function

$$\nabla_{x_t} \log p_\theta(x_t, t)$$

By Bayes' Theorem, we know that

$$p(x|c) = \frac{p(x,c)}{p(c)} = \frac{p(c|x)p(x)}{p(c)} \propto p(c|x)p(x)$$

$$\nabla_{x_t} \log p_\theta(x_t, t|c) = \nabla_{x_t} \log p_\theta(c|x_t, t) + \nabla_{x_t} \log p_\theta(x_t, t)$$

Pre-trained unconditional diffusion model

A discriminative model!
(e.g. a classifier, a CLIP model, etc)

The classifier needs to be able to process noisy images!

Carnegie Mellon University

We need to train a noisy classifier for classifier guidance diffusion!

Unconditional diffusion:

Unconditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t)$

Conditional diffusion:

"a headshot of a woman with dark hair"

Conditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t|c)$

Classifier guidance diffusion:

"a headshot of a woman with dark hair"

Unconditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t)$

Noisy Classifier $p_\theta(c|x_t, t)$

The noisy classifier needs to be specifically trained to pair with the unconditional diffusion model

Carnegie Mellon University

Is it possible to use an off-the-shelf classifier?

Unconditional diffusion:

Unconditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t)$

Conditional diffusion:

"a headshot of a woman with dark hair"

Conditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t|c)$

Classifier guidance diffusion:

"a headshot of a woman with dark hair"

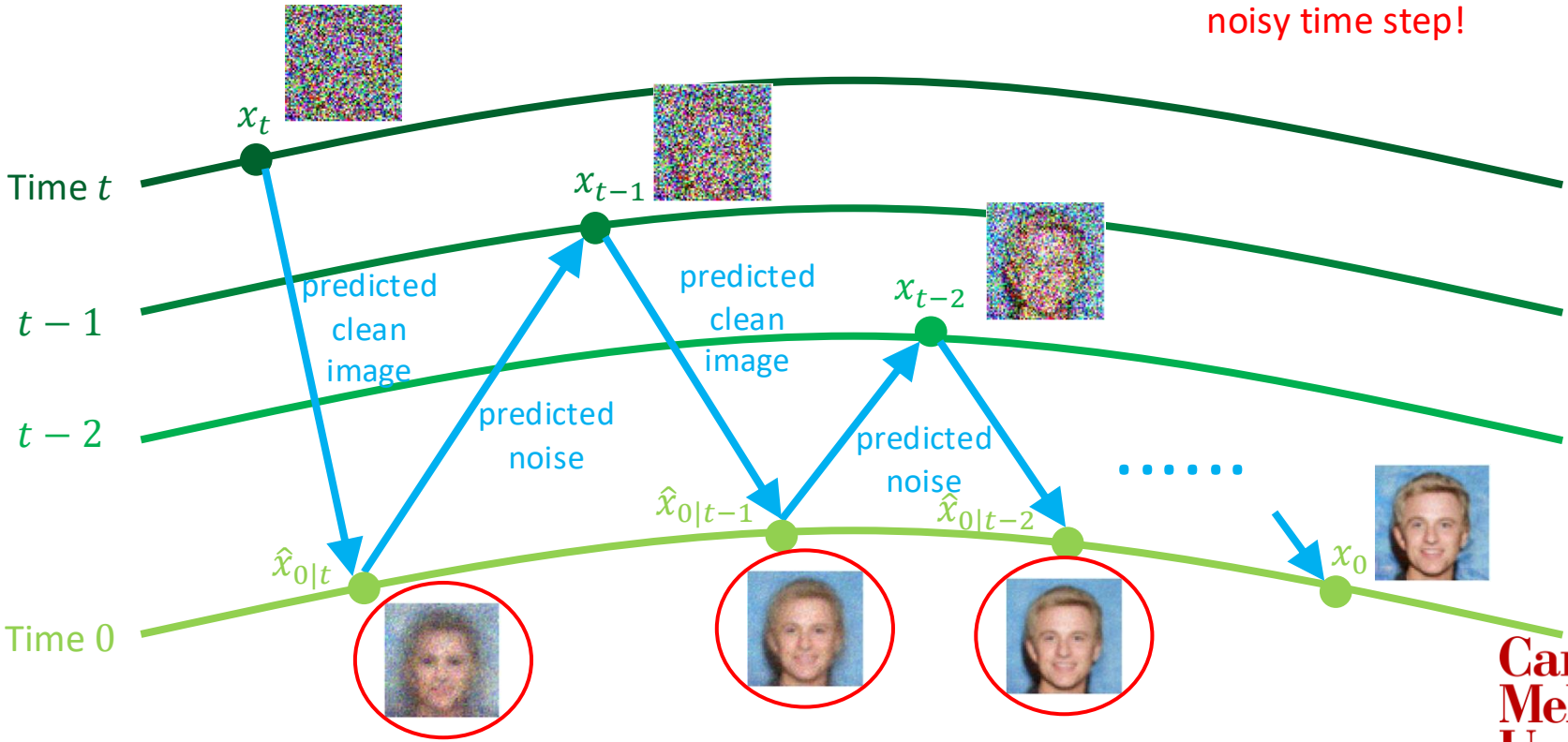
Unconditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t)$

Off-the-shelf Classifier $p_\phi(c|x_0)$

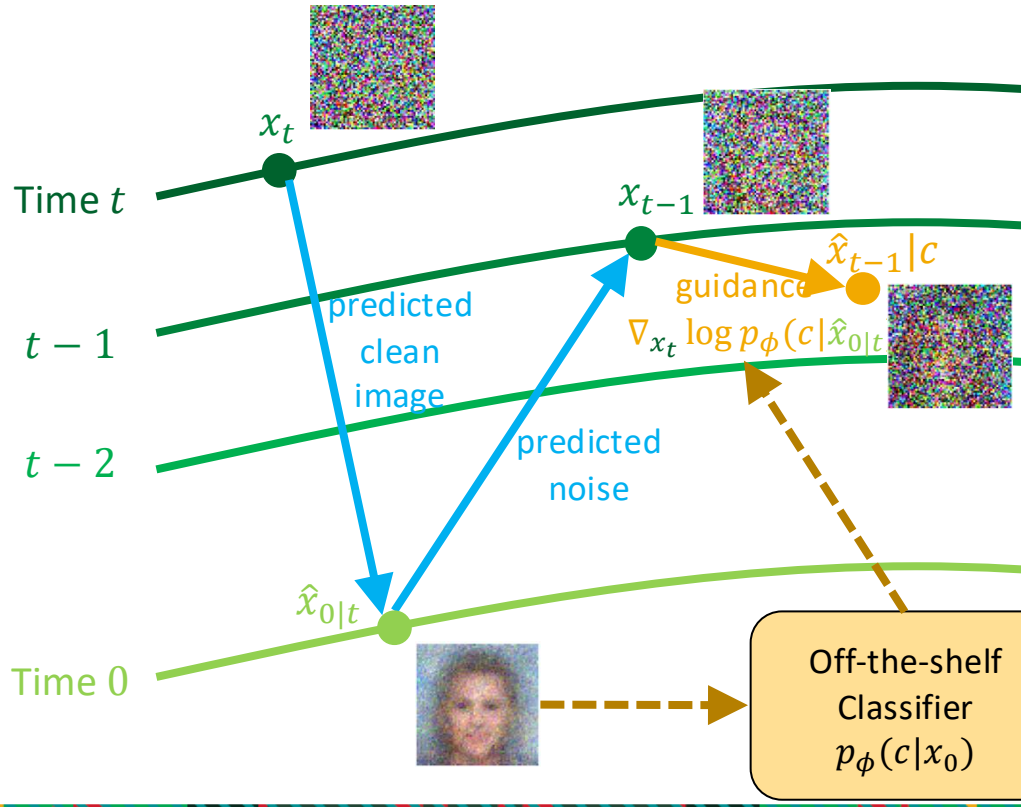
Carnegie Mellon University

Remember how we do DDIM

We do have some clean image estimates from each noisy time step!



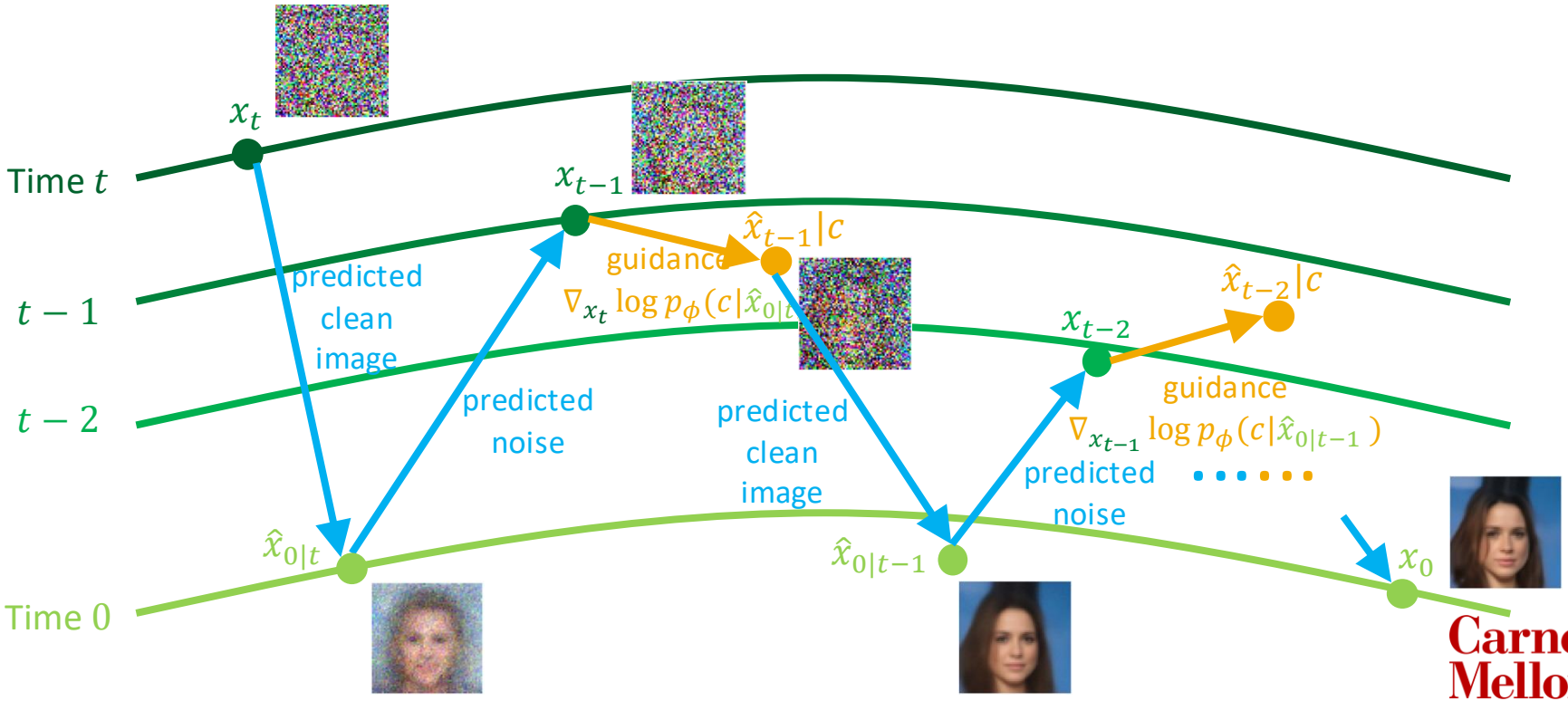
Using off-the-shelf classifiers for diffusion guidance



This is called diffusion posterior sampling

Carnegie Mellon University

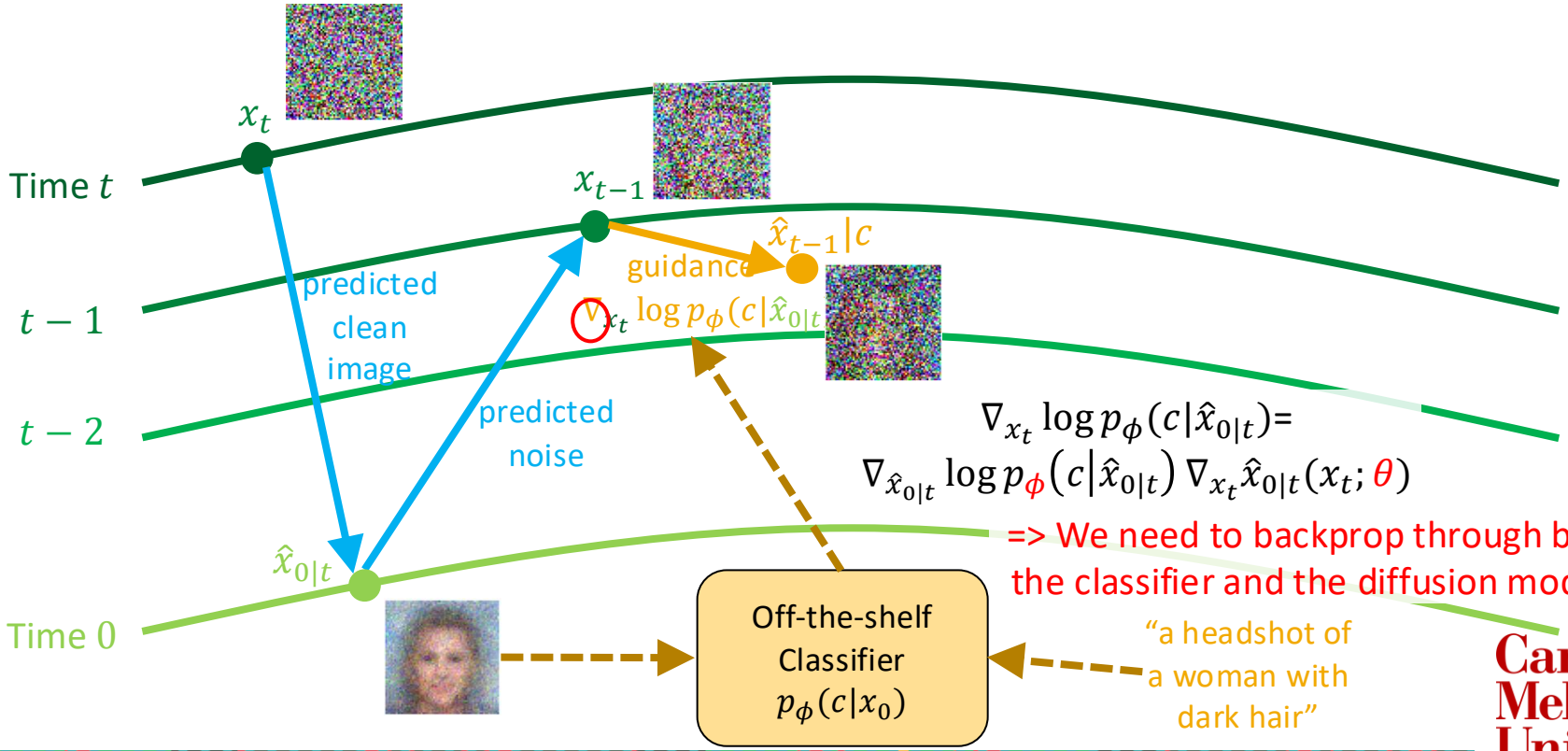
Diffusion posterior sampling (DPS)



Carnegie Mellon University

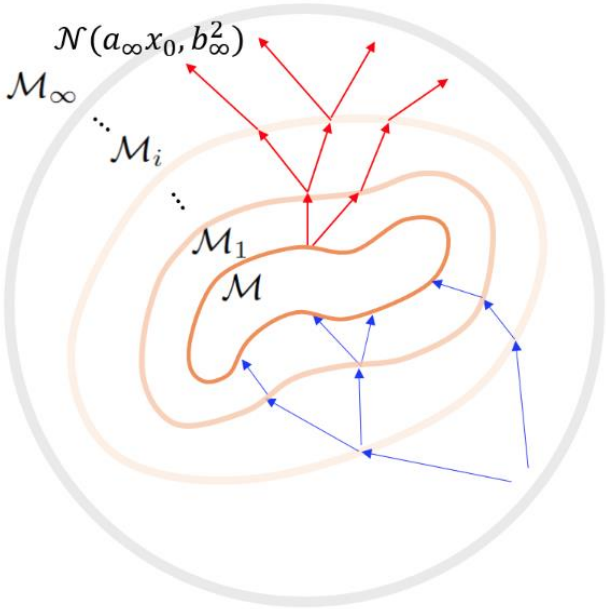
Chung et al. "Diffusion Posterior Sampling for General Noisy Inverse Problems". ICLR 2023. https://arxiv.org/pdf/2209.14687

DPS is great, but



Carnegie Mellon University

Diffusion manifolds are layered bubble shells



(a) Geometry of diffusion model

Why?
Because high dimensional Gaussians are soap bubbles!

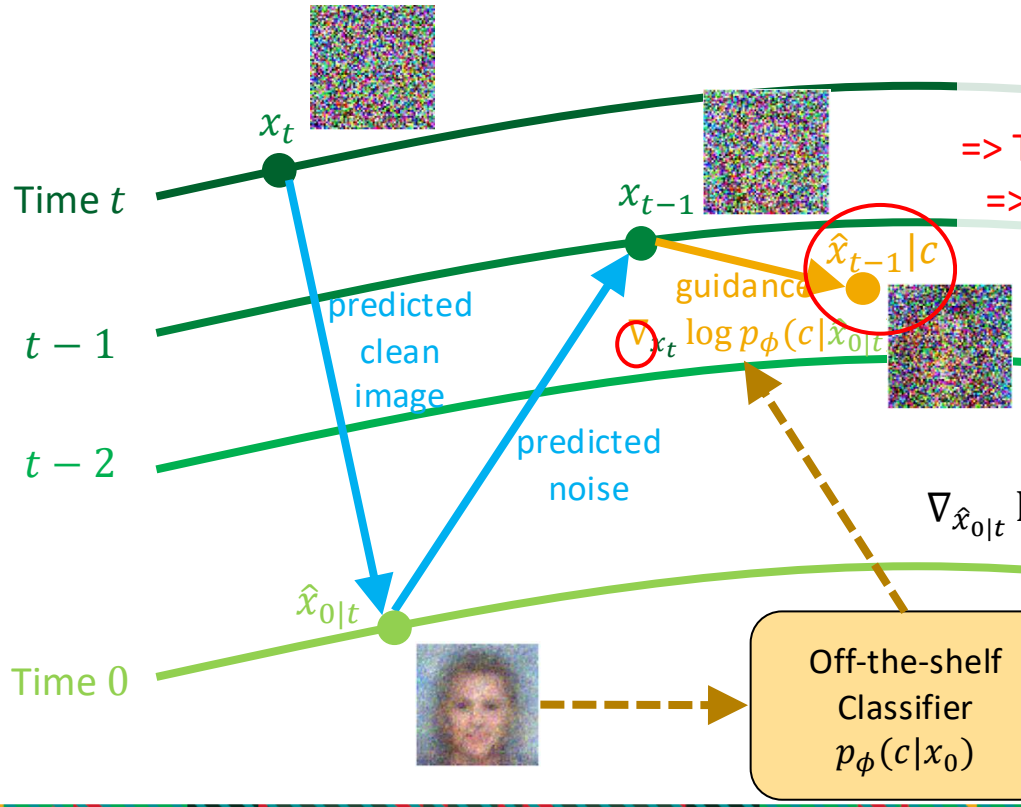
Proof hint:
Try to see where is the majority of the density concentrated at in high dimensional Gaussians

Carnegie Mellon University

Chung et al. "Improving Diffusion Models for Inverse Problems using Manifold Constraints". NeurIPS 2022. https://arxiv.org/pdf/2206.00941

DPS is great, but

DPS is very very slow!



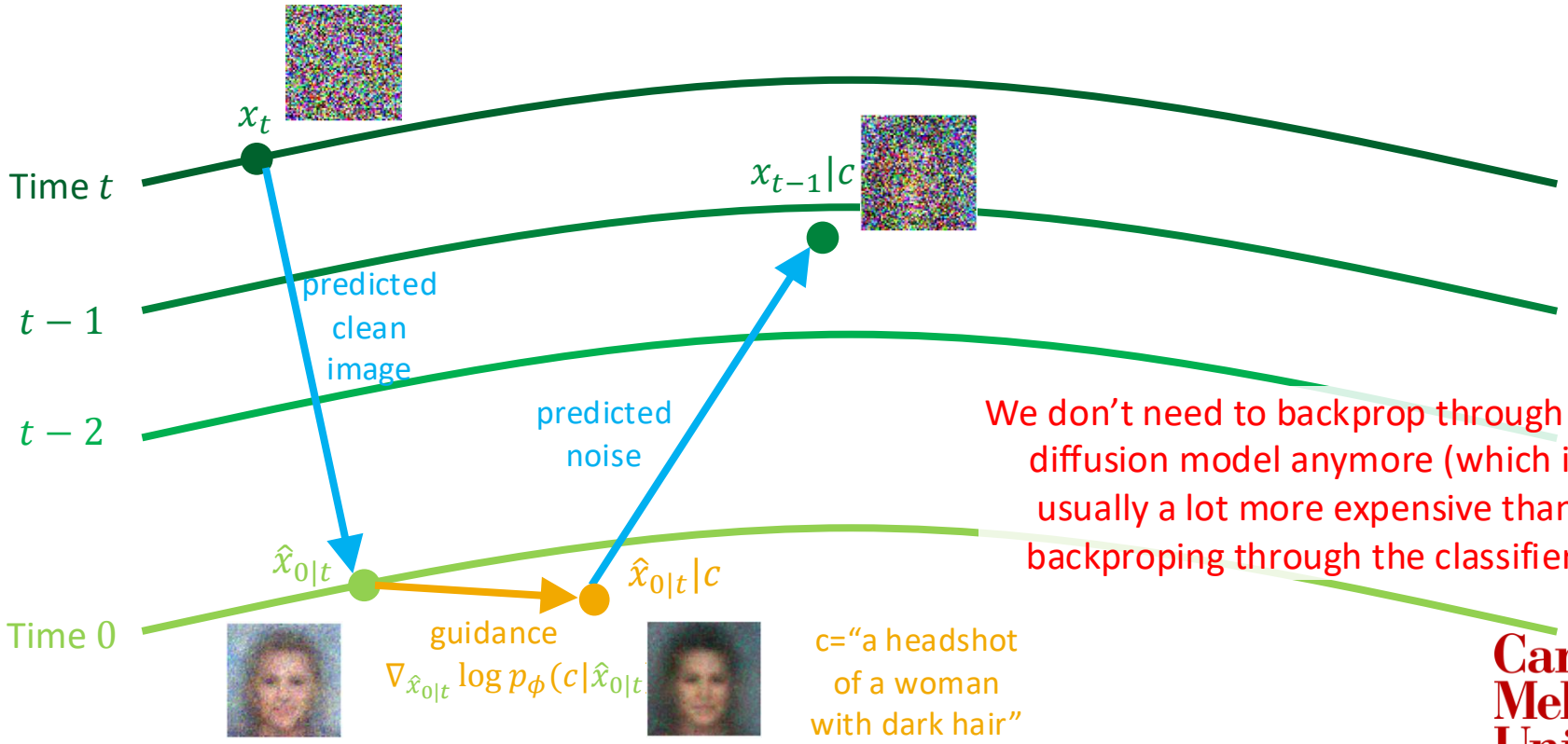
The guidance is unconstrained!
=> The guided sample can go off the manifold!
=> Need very small guidance steps and/or a lot of diffusion steps!

$$\nabla_{x_t} \log p_\phi(c|\hat{x}_{0|t}) = \nabla_{\hat{x}_{0|t}} \log p_\phi(c|\hat{x}_{0|t}) \nabla_{x_t} \hat{x}_{0|t}(x_t;\theta)$$

=> We need to backprop through both the classifier and the diffusion model!

Carnegie Mellon University

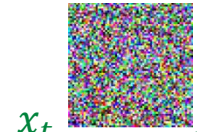
Remedy 1: Take gradient w.r.t. the clean image instead



We don't need to backprop through the diffusion model anymore (which is usually a lot more expensive than backproping through the classifier)

Carnegie Mellon University

Remedy 2: Project the guided samples to the manifold



x_t

Time t

$t-1$

$t-2$

Time 0

$x^*_{t-1}|c$

Tangent space of the noisy manifold

$x_{t-1}|c$

predicted clean image

predicted noise

But we don't have access to those manifolds

$\hat{x}_{0|t}$

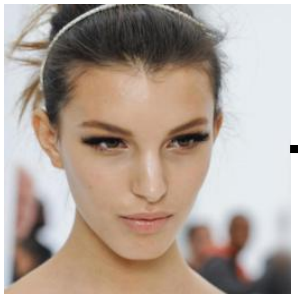
$\hat{x}_{0|t}|c$

guidance $\nabla_{\hat{x}_{0|t}} \log p_\phi(c|\hat{x}_{0|t})$

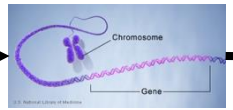
c="a headshot of a woman with dark hair"

Carnegie Mellon University

Getting access to data manifolds via the decoder of an autoencoder

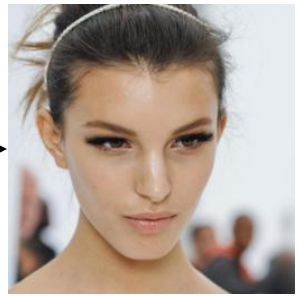


Encoder
E_ϕ

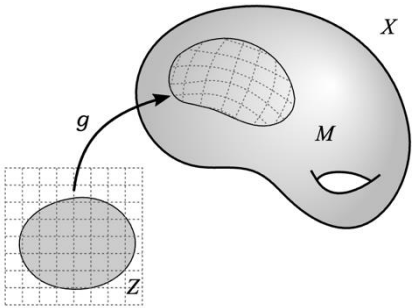


Z

Decoder
D_θ



Reconstruction



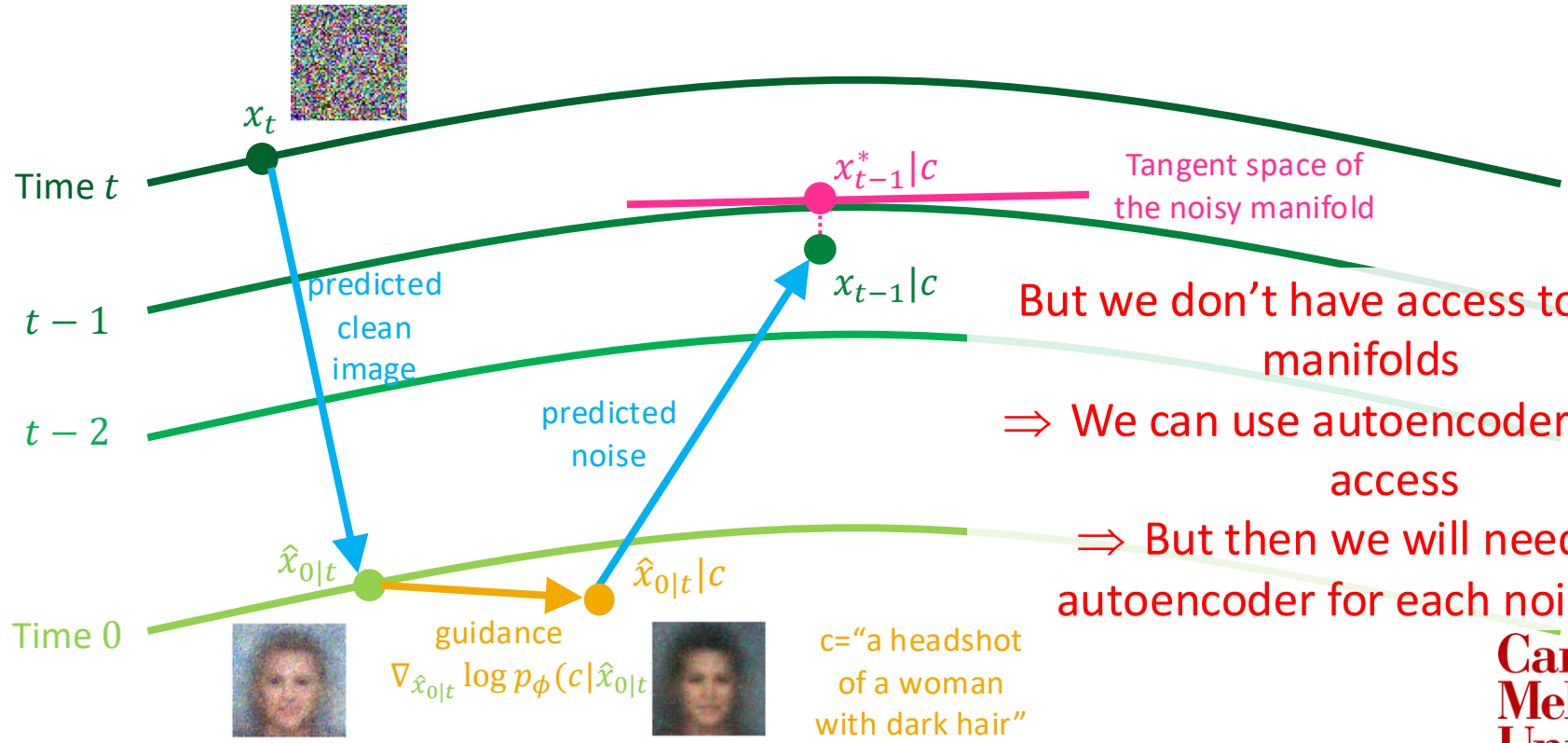
Low dimensional representation of the data

The Jacobian of the decoder gives us access to the tangent space of the data manifold

Carnegie Mellon University

Shao et al. "The Riemannian Geometry of Deep Generative Models". CVPR 2018 workshop. https://arxiv.org/pdf/2311.16424zd

Remedy 2: Project the guided samples to the manifold



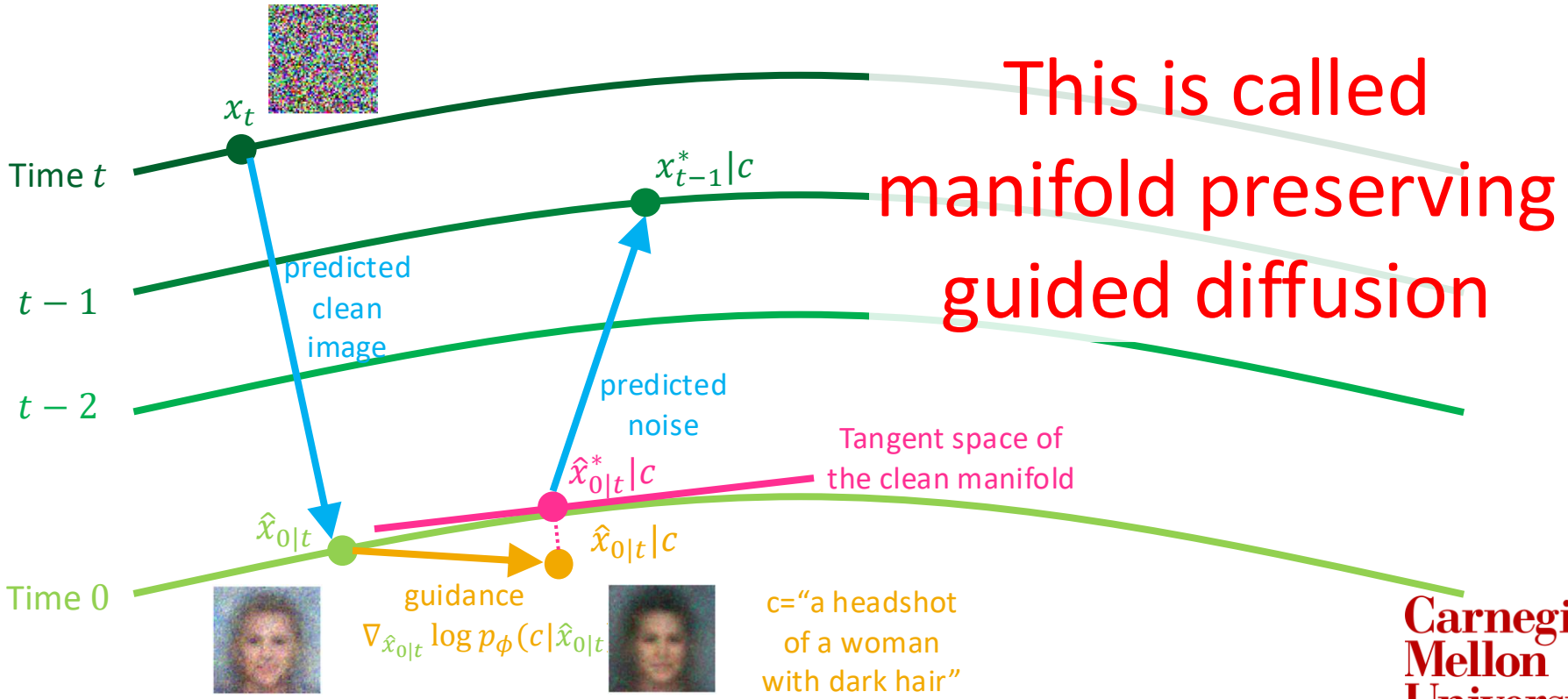
But we don't have access to those manifolds

$\Rightarrow$ We can use autoencoders to get access

$\Rightarrow$ But then we will need an autoencoder for each noise level!

Carnegie Mellon University

Remedy 2.5: Project the clean data estimation to the clean data manifold



Carnegie Mellon University

He et al. "Manifold Preserving Guided Diffusion". ICLR 2024. https://arxiv.org/pdf/2311.16424

Manifold preserving guided diffusion (MPGD) is much faster (and better) than DPS

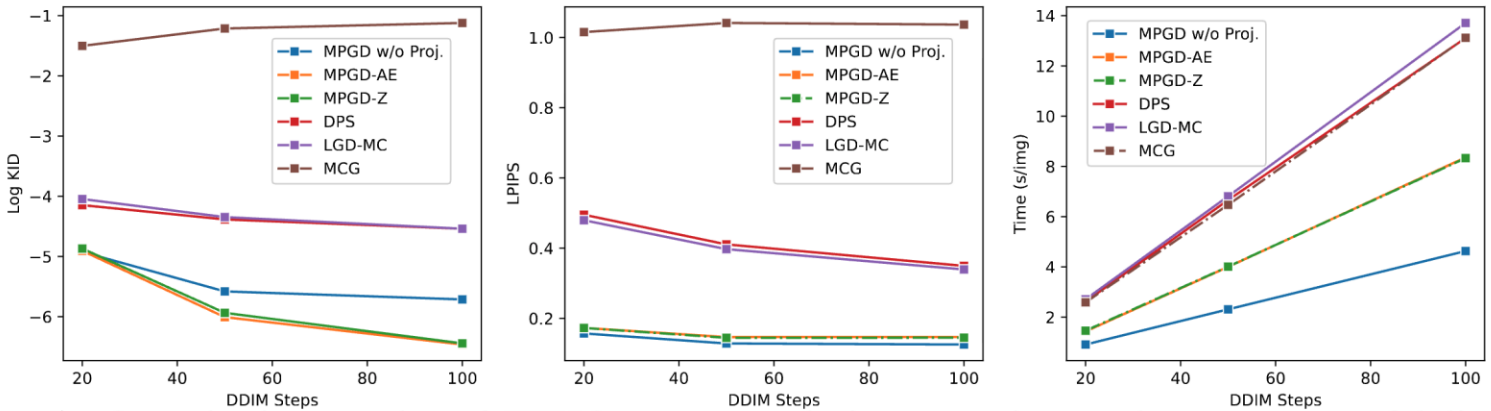
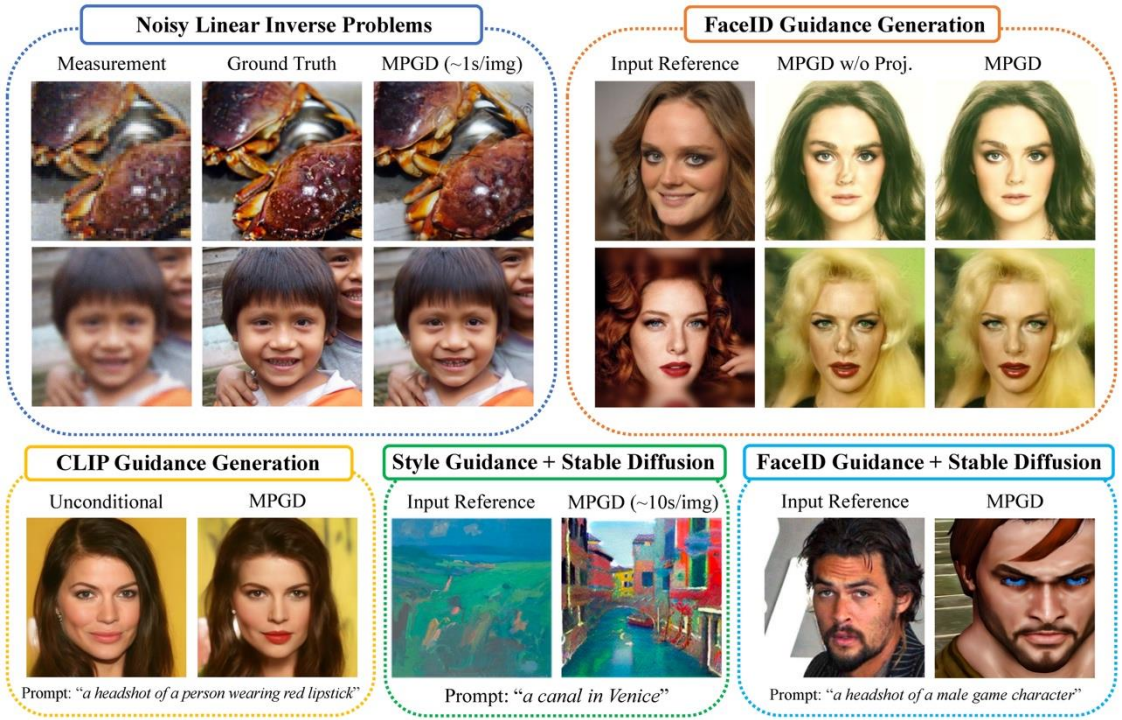


Figure 5: Quantitative results of FFHQ super-resolution experiment that compares fidelity (log KID), guidance quality (LPIPS) and inference time across different numbers of DDIM steps.

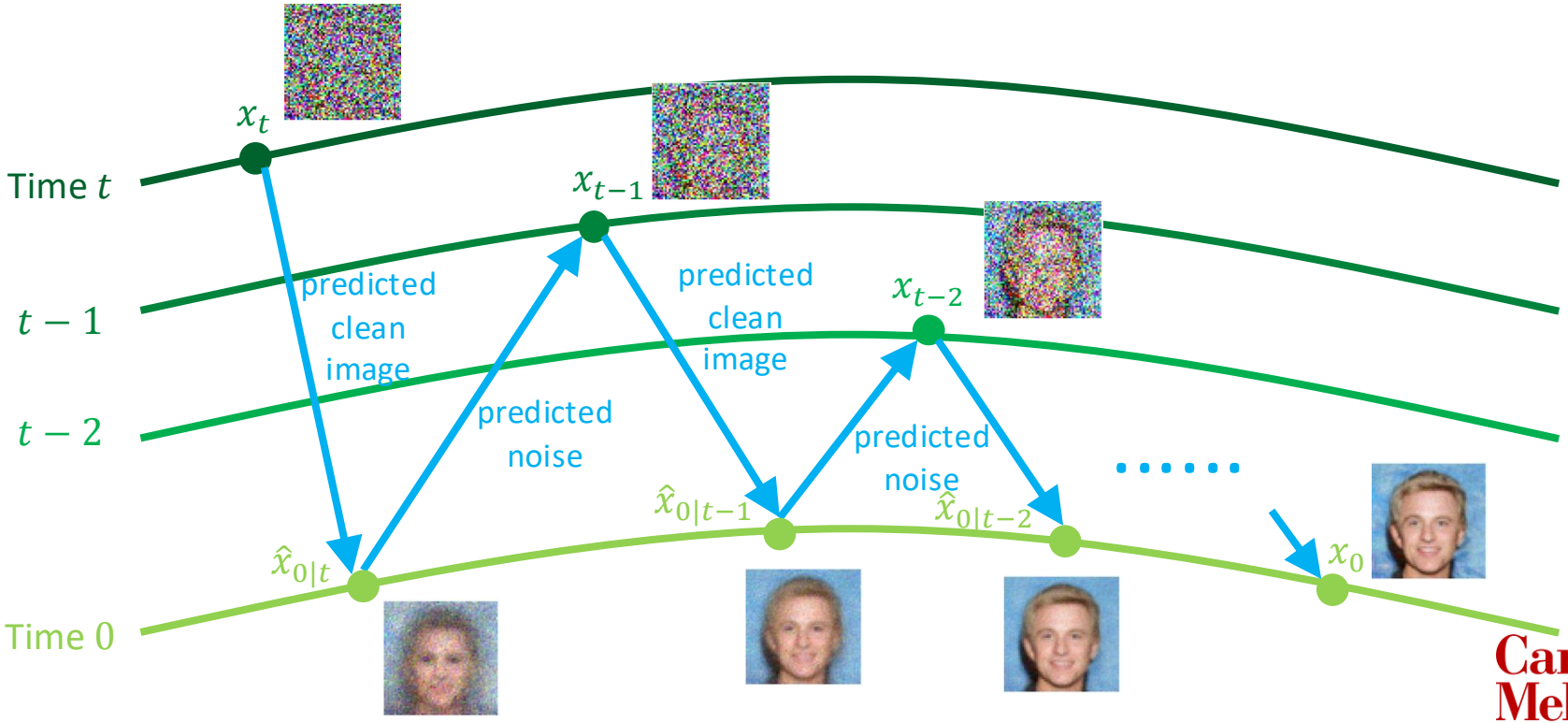
Carnegie Mellon University

MPGD is training-free and applicable to all differentiable guidance/classifier/reward



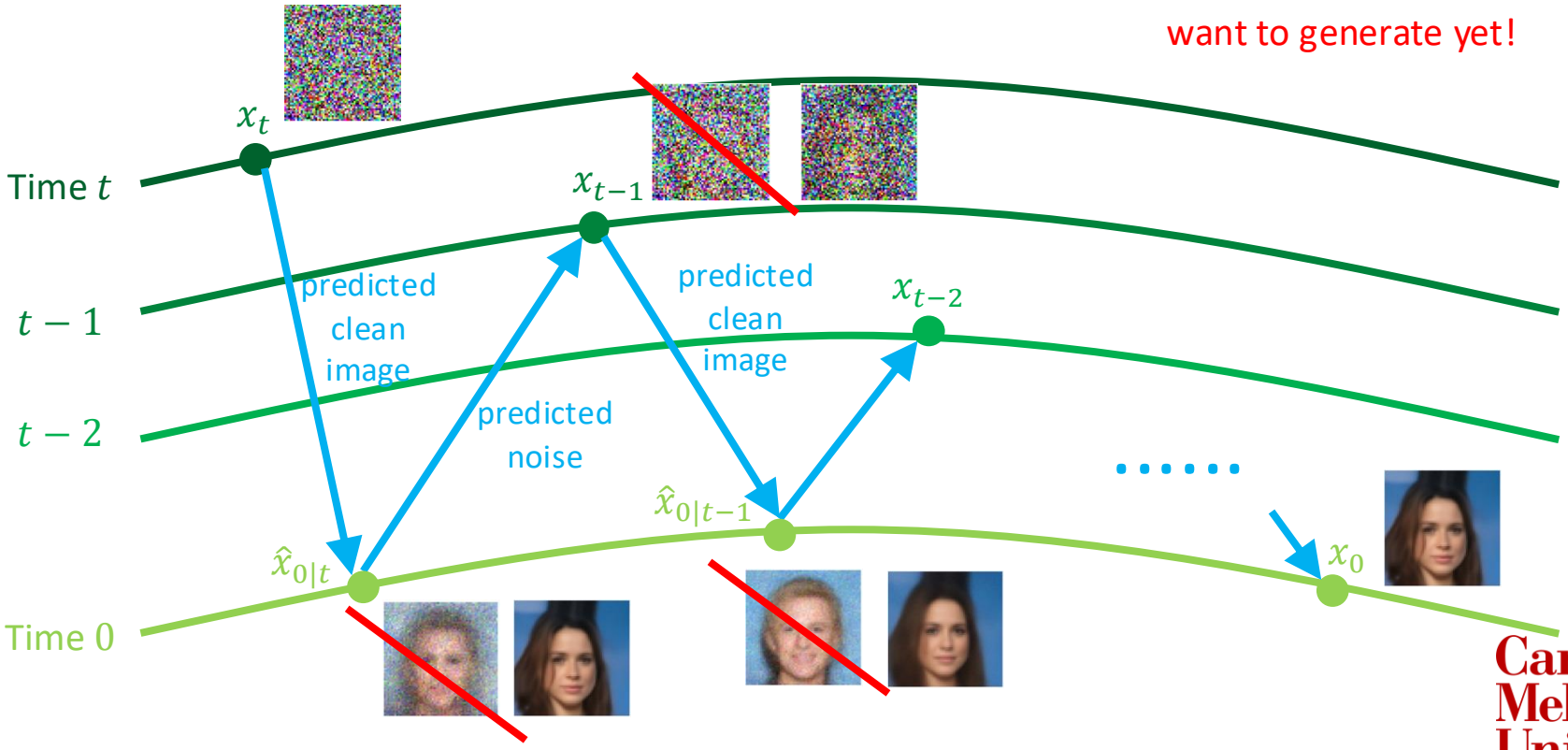
Any other training-free ways to inject conditions into diffusion models?

Remember how we do DDIM



If we already know what we want to generate

But we don't have what we want to generate yet!



But do we really need to know EXACTLY what we need to generate?

We only need a rough draft/preliminary version of what we want to generate!

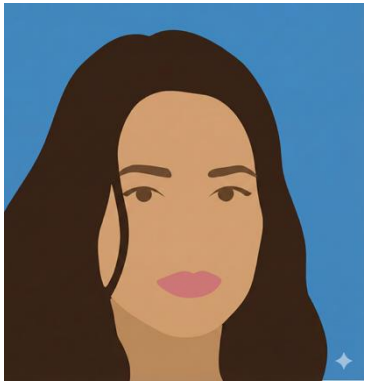
But we also don’t have a draft!

We only need a rough draft/preliminary version of what we want to generate!

Or do we?

How about just draw one lol

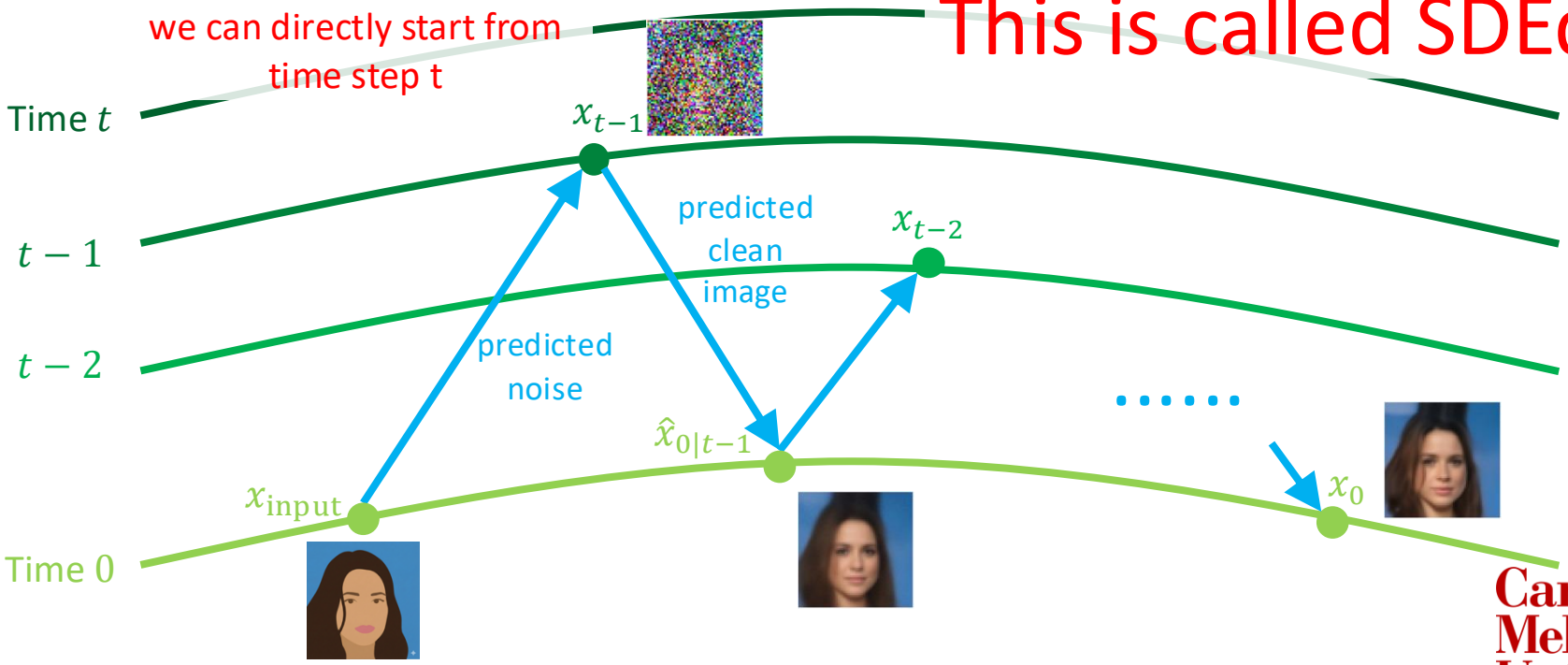


With this rough painting, we can do

Notice that we don't need to do any steps before t now, we can directly start from time step t

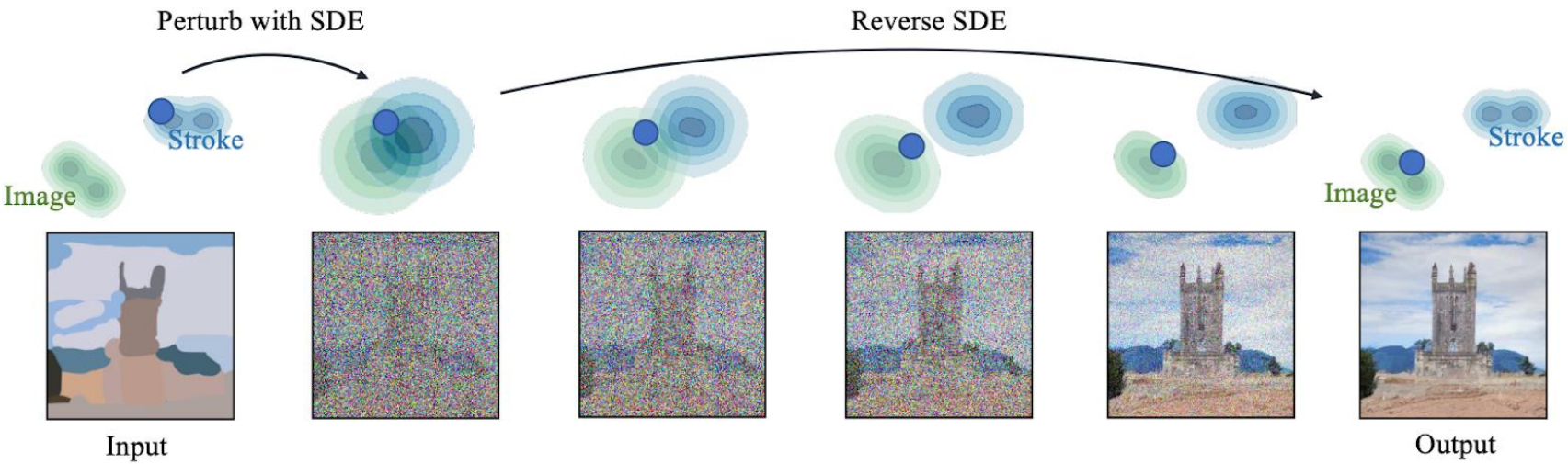
This is called SDEdit



Carnegie Mellon University

Meng et al. "SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations". ICLR 2022. https://arxiv.org/pdf/2108.01073.pdf

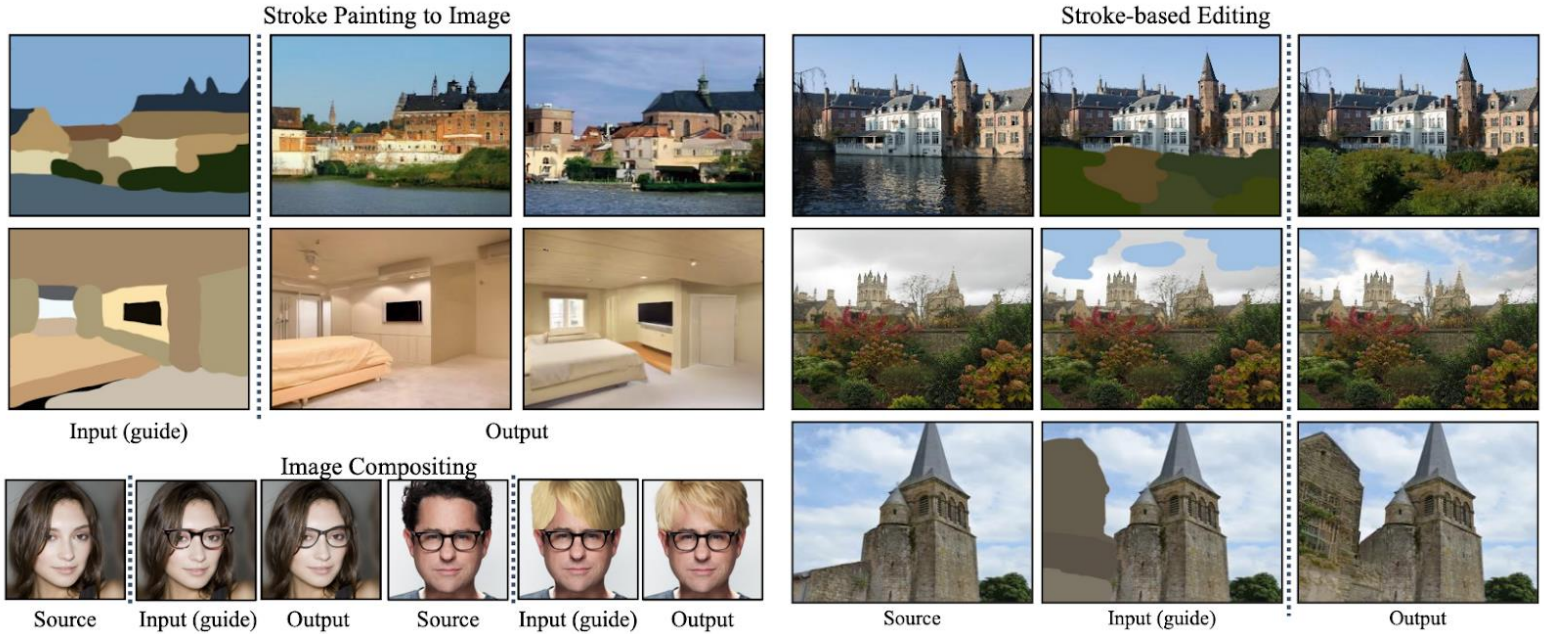
SDEdit hijacks the diffusion process with user inputs



Carnegie Mellon University

Meng et al. "SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations". ICLR 2022. https://arxiv.org/pdf/2108.01073.pdf

SDEdit can turn your preliminary paintings/edits into realistic looking images



Meng et al. "SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations". ICLR 2022. https://arxiv.org/pdf/2108.01073.pdf

Carnegie Mellon University

SDEdit can turn your preliminary paintings/edits into realistic looking images

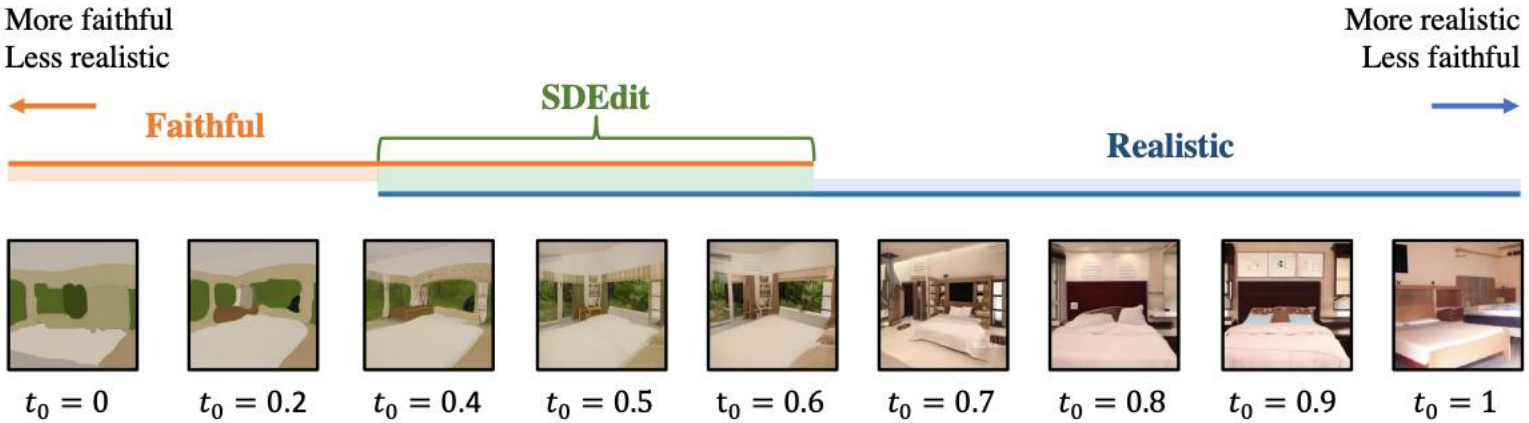
Original Image

User Input

SDEdit Output

Carnegie Mellon University

The controllability-fidelity tradeoff



Carnegie Mellon University

Meng et al. "SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations". ICLR 2022. https://arxiv.org/pdf/2108.01073.pdf

Guidance pro tips 1: You should guide more during the middle time steps (wisdom from EDM still holds)

Phase#1: **Chaotic** Stage
(#1000~#800 Timestep)
Almost Out of Control

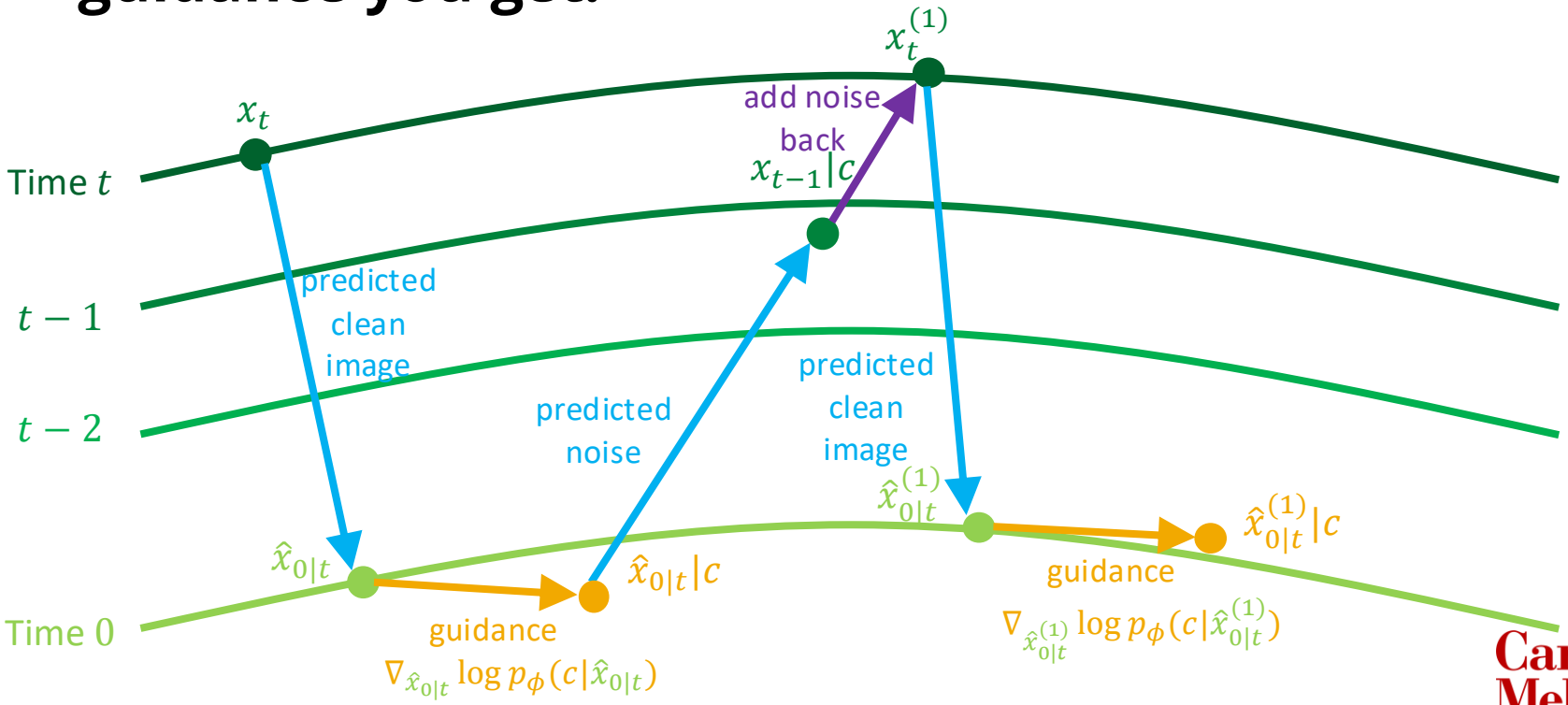
Phase#2: **Semantic** Stage
(#800~#500 Timestep)
Critcal for Conditional Generation

Phase#3: **Refinement** Stage
(#500~#1 Timestep)
Too Late for Full Control

Carnegie Mellon University

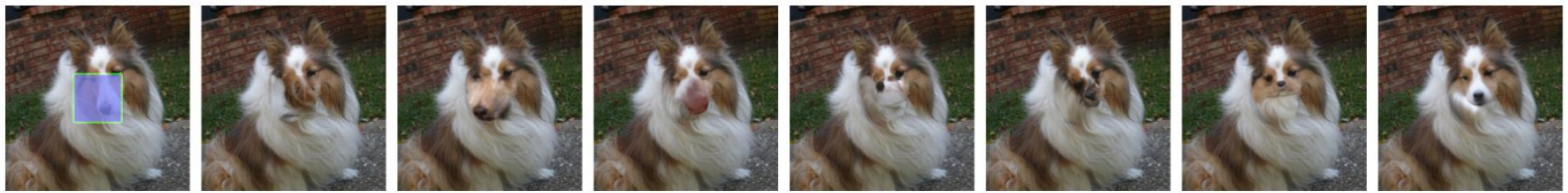
Yu et al. "FreeDoM: Training-Free Energy-Guided Conditional Diffusion Model". ICCV 2023. https://arxiv.org/pdf/2303.09833

Guidance pro tips 2: You can even time travel back to the previous time step if you want to refine the guidance you get!



Carnegie Mellon University

Guidance pro tips 2: You can even time travel back to the previous time step if you want to refine the guidance you get!



Carnegie Mellon University

Lugmayr et al. “RePaint: Inpainting using Denoising Diffusion Probabilistic Models”. CVPR 2022. https://arxiv.org/pdf/2201.09865

What if we are normies

Unconditional diffusion:

Unconditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t)$

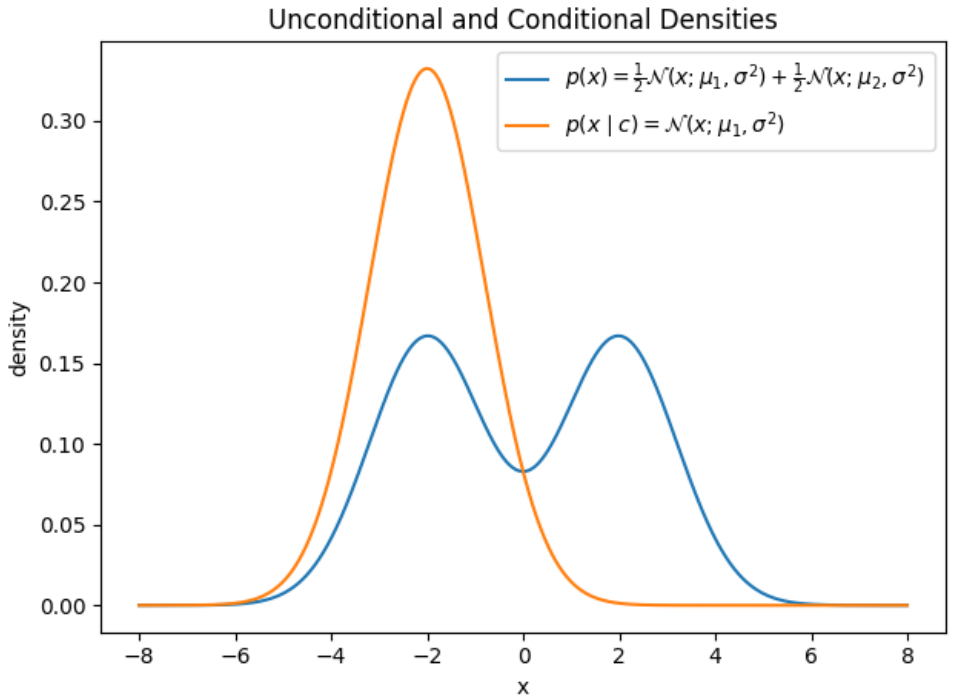
Conditional diffusion:

"a headshot of a woman with dark hair"

Conditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t|c)$

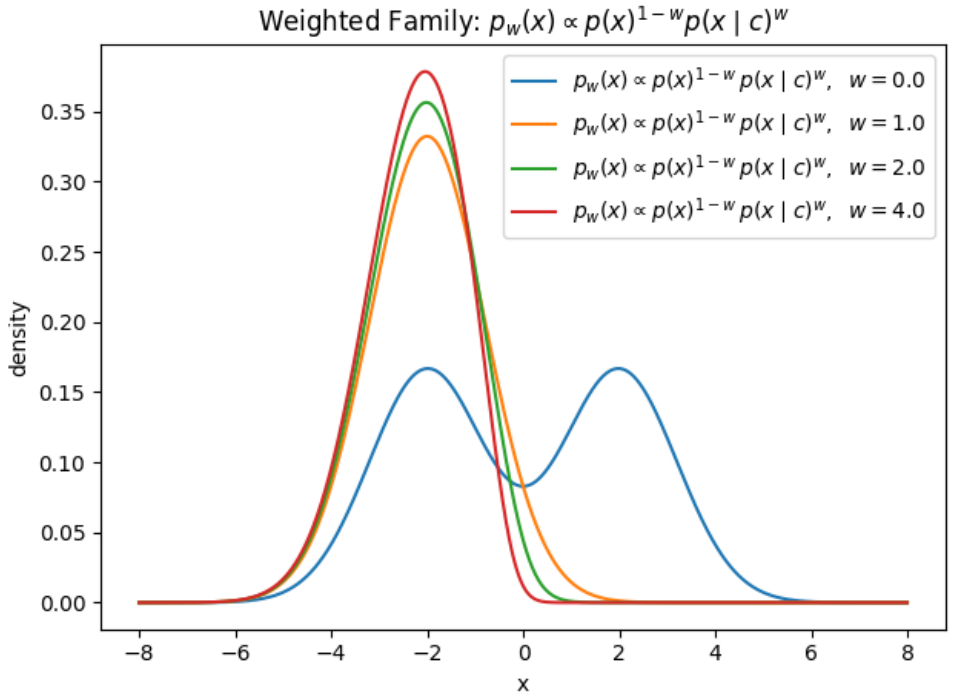
If we have already trained an unconditional diffusion model and a separate conditional diffusion model, can we somehow **use both** to make our conditional generation better?

Unconditional distribution v.s. Conditional distribution



Carnegie Mellon University

Conditional mixed with unconditional distributions



Carnegie Mellon University

Classifier-free guidance: Sample from the weighted mixture of unconditional + conditional score

$$\nabla_x \log p_\gamma(x \mid y) = (1-\gamma)\nabla_x \log p(x) + \gamma \nabla_x \log p(x \mid y).$$

Algorithm 1 Joint training a diffusion model with classifier-free guidance

Require: p_{uncond}: probability of unconditional training

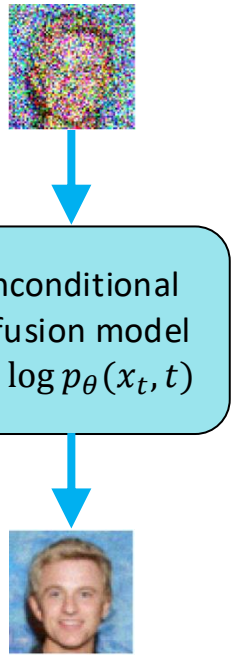
1: **repeat**
2: $(\mathbf{x}, \mathbf{c}) \sim p(\mathbf{x}, \mathbf{c})$ ▷ Sample data with conditioning from the dataset
3: $\mathbf{c} \leftarrow \varnothing$ with probability p_{uncond} ▷ Randomly discard conditioning to train unconditionally
4: $\lambda \sim p(\lambda)$ ▷ Sample log SNR value
5: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
6: $\mathbf{z}_\lambda = \alpha_\lambda \mathbf{x} + \sigma_\lambda \boldsymbol{\epsilon}$ ▷ Corrupt data to the sampled log SNR value
7: Take gradient step on $\nabla_\theta \left\| \boldsymbol{\epsilon}_\theta(\mathbf{z}_\lambda, \mathbf{c}) - \boldsymbol{\epsilon} \right\|^2$ ▷ Optimization of denoising model
8: **until** converged

Carnegie Mellon University

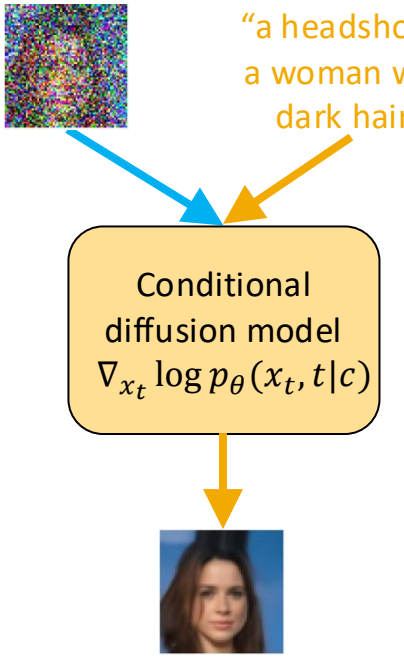
Ho & Salimans. "Classifier-Free Diffusion Guidance". NeurIPS 2021 Workshop. https://arxiv.org/pdf/2207.12598

Do we even need to train two separate models?

Unconditional diffusion:



Conditional diffusion:



Do we even need to train two separate models?

Unconditional diffusion:

Unconditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t)$

Conditional diffusion:

"a headshot of a woman with dark hair"

Conditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t|c)$

Same model (share parameters)

Pseudo unconditional diffusion:

""

Conditional diffusion model $\nabla_{x_t} \log p_\theta(x_t, t|\emptyset)$

Carnegie Mellon University

Classifier-free guidance: Sample from the weighted mixture of unconditional + conditional score

$$\nabla_x \log p_\gamma(x \mid y) = (1-\gamma)\nabla_x \log p(x) + \gamma \nabla_x \log p(x \mid y).$$

Two sets of samples from OpenAI's GLIDE model, for the prompt *'A stained glass window of a panda eating bamboo.'*, taken from their paper. Guidance scale 1 (no guidance) on the left, guidance scale 3 on the right.

Carnegie Mellon University

Images from Sander Dieleman https://sander.ai/2022/05/26/guidance.html

Now you know how to turn an unconditional diffusion into a conditional one!

Training-based methods:

- Classifier guidance diffusion
- Classifier free guidance (if you don’t already have a conditional model)

Training-free methods:

- DPS
- MPGD
- SDEdit
- Classifier free guidance (if you already have a conditional model)

In the next two weeks, we are going into the realm of the state-of-the-arts

- **2/3 (Tue):** How to use diffusion/flow models for robotics, control & decision making (**Max Simchowitz**, MLD Prof.)
- **2/5 (Thur):** Text-to-image models and SOTA techniques
- **2/10 (Tue):** How to sample from diffusion/flow models with a single step
    - Distillation
    - Consistency models
    - Flow maps
- **2/12 (Thur):** Real-time generation techniques for video (**Linqi (Alex) Zhou**, Luma AI)

In the next two weeks, we are going into the realm of the state-of-the-arts

- **2/3 (Tue):** How to use diffusion/flow models for robotics, control & decision making (**Max Simchowitz**, MLD Prof.) (Will be chill, please come in person!)
- **2/5 (Thur):** Text-to-image models and SOTA techniques
- **2/10 (Tue):** How to sample from diffusion/flow models with a single step
    - Distillation
    - Consistency models
    - Flow maps
- **2/12 (Thur):** Real-time generation techniques for video (**Linqi (Alex) Zhou**, Luma AI)