Sequence Generative Modeling

Prof. Vincent Sitzmann

Sequences are everywhere…

“A busy street in Europe.
A small crowd of people are
walking…”

“A Gibson Les Paul Electric
Guitar in a case…”

“A pair of birds built from Lego…”

Diffusion Probabilistic Modeling for Video Generation, Yang et al. 2022
Video Diffusion Models, Ho et al. 2022

Sequences are everywhere…

“A busy street in Europe.
A small crowd of people are
walking…”

“A Gibson Les Paul Electric
Guitar in a case…”

“A pair of birds built from Lego…”

Diffusion Probabilistic Modeling for Video Generation, Yang et al. 2022
Video Diffusion Models, Ho et al. 2022

Video Generative Modeling is an Essential Vision Problem

Optical flow tells us about 3D

Content of image from one moment to the next tells us about semantics (what things co-occur)

We can learn about processes (such as cooking) from video

Good candidate for representation learning

Video Generative Modeling is an Essential Vision Problem

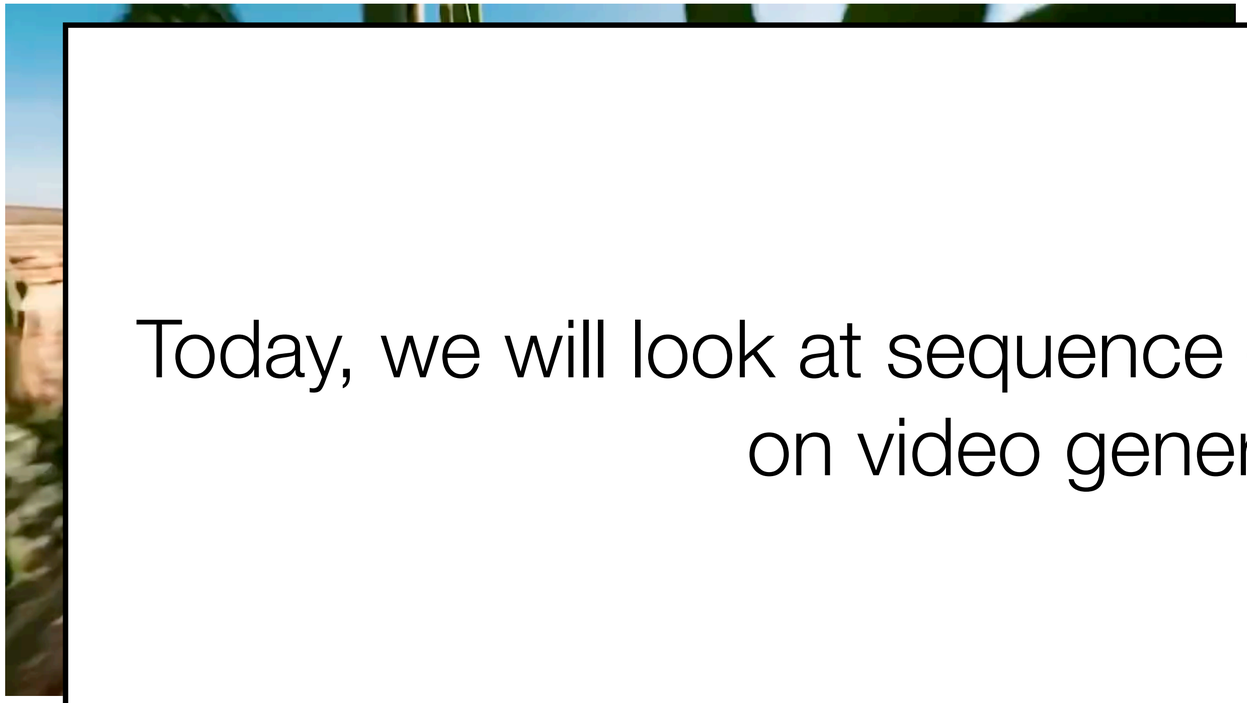
Optical flow tells us about 3D

Content of image from one moment to the next tells us about semantics (what things co-occur)

We can learn about processes (such as cooking) from video

Good candidate for representation learning

Video Generative Modeling is an Essential Vision Problem



Optical flow tells us about 3D

Today, we will look at sequence generation with a particular focus on video generative modeling.

learning

Preliminaries: Distributions of Sequences

Preliminaries: Distributions of Sequences

As the name implies, we are modeling distributions of sequences, denoted as

$$p(x_1, x_2, x_3, \dots)$$

Causal Factorization of the Joint

Causal Factorization of the Joint

We can always model the joint auto-regressively:

$$p(x_1, x_2, x_3, \dots) = \prod_{i=1}^{N} p(x_i \mid x_{<i})$$

That is, the future only depends on the past, not on the future.

Note that this is just a re-writing of the joint -
this is a mathematical fact. Any distribution can be factorized in this way, even ones that aren't causal to begin with.

The Markov Property

The Markov Property

Some sequences can be factorized as:

$$p(x_1, x_2, x_3, \dots) = p(x_1) \prod_{i=1}^{N} p(x_{i+1} \mid x_i)$$

“Given the present, the future is independent of the past.”

This is *not* a re-writing of the joint, b/c we dropped a bunch of conditionals! I.e. if a sequence is not Markov to begin with, we can’t appropriately model it as such.

The Markov Property

The Markov Property

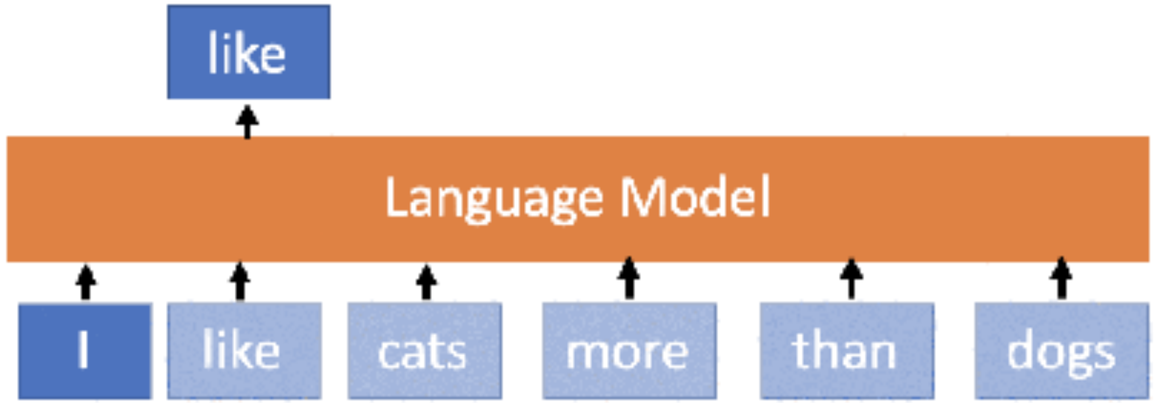
Varying degrees of “Markovianity” are possible:

$$p(x_1, x_2, x_3, \dots) = p(x_1) \prod_{i=1}^{N} p(x_{i+1} \mid x_i, x_{i-1})$$

Consider a video of a pendulum. Given just one frame, you cannot accurately predict the next - it could be swinging in either direction! Given two frames, you can.

Two major paradigms for sequence generation

Video generation generally trained with **Full-Sequence Diffusion**

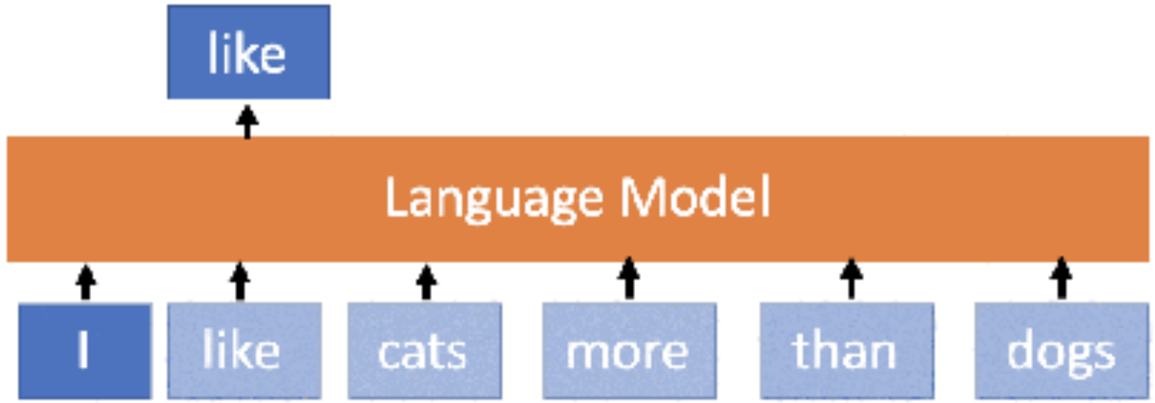


LLMs generally trained with **Next-Token Prediction**

Diffusion Probabilistic Modeling for Video Generation, Yang et al. 2022
Video Diffusion Models, Ho et al. 2022

Two major paradigms for sequence generation

Video generation generally trained with **Full-Sequence Diffusion**



LLMs generally trained with **Next-Token Prediction**

Diffusion Probabilistic Modeling for Video Generation, Yang et al. 2022
Video Diffusion Models, Ho et al. 2022

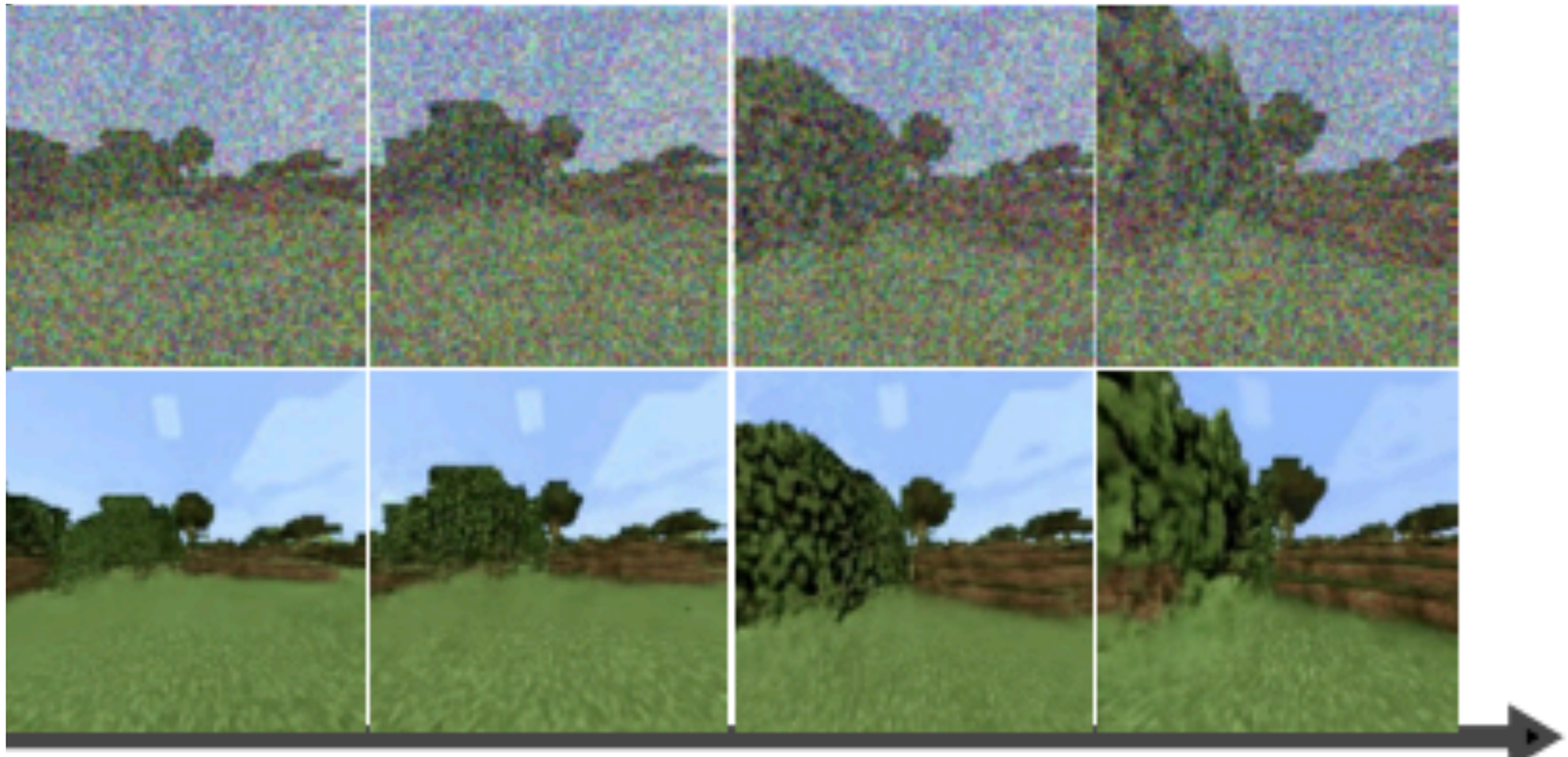
Preliminaries: Noise as Masking

Unmasked

Preliminaries: Noise as Masking

Slightly Masked

Unmasked

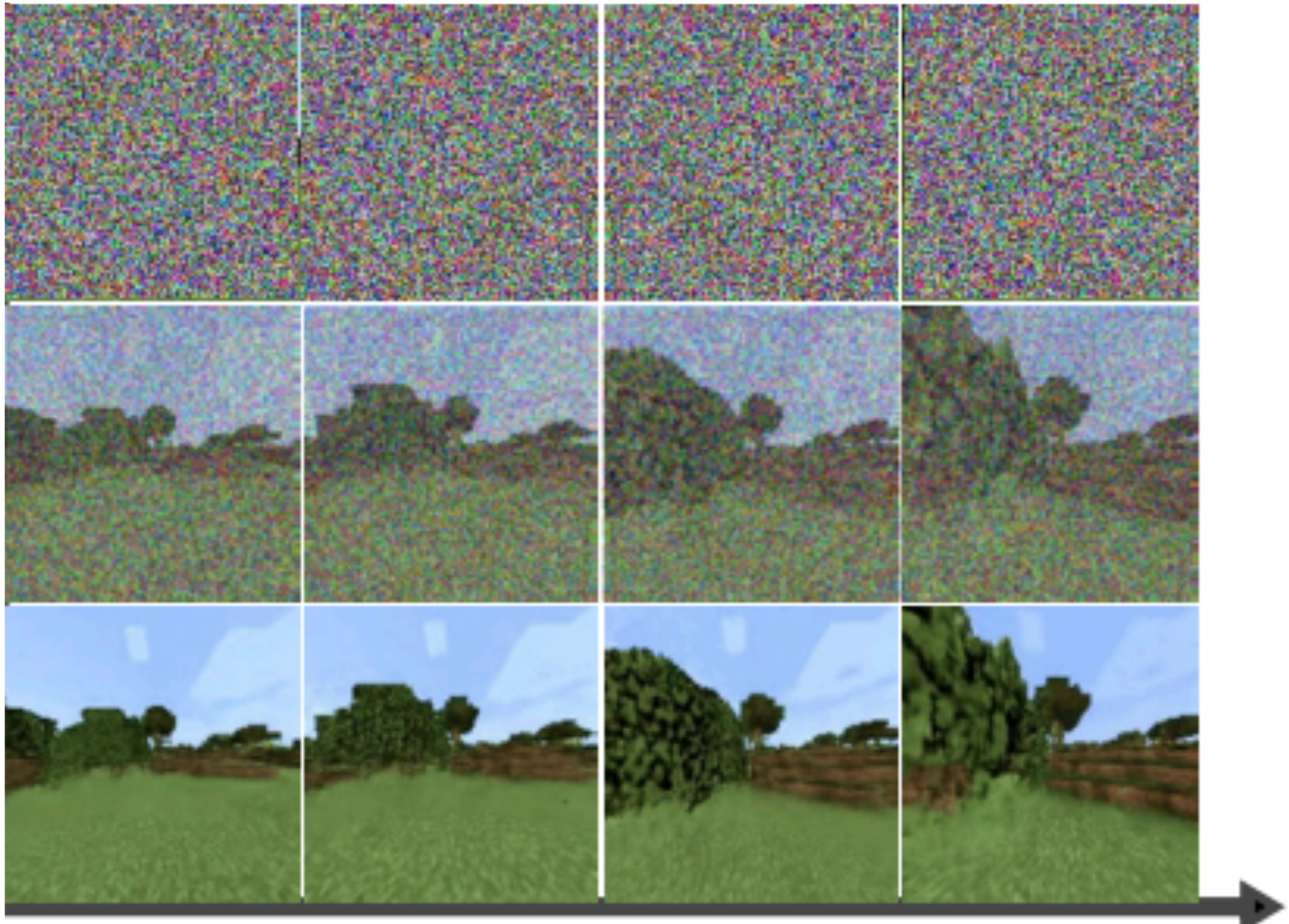


Preliminaries: Noise as Masking

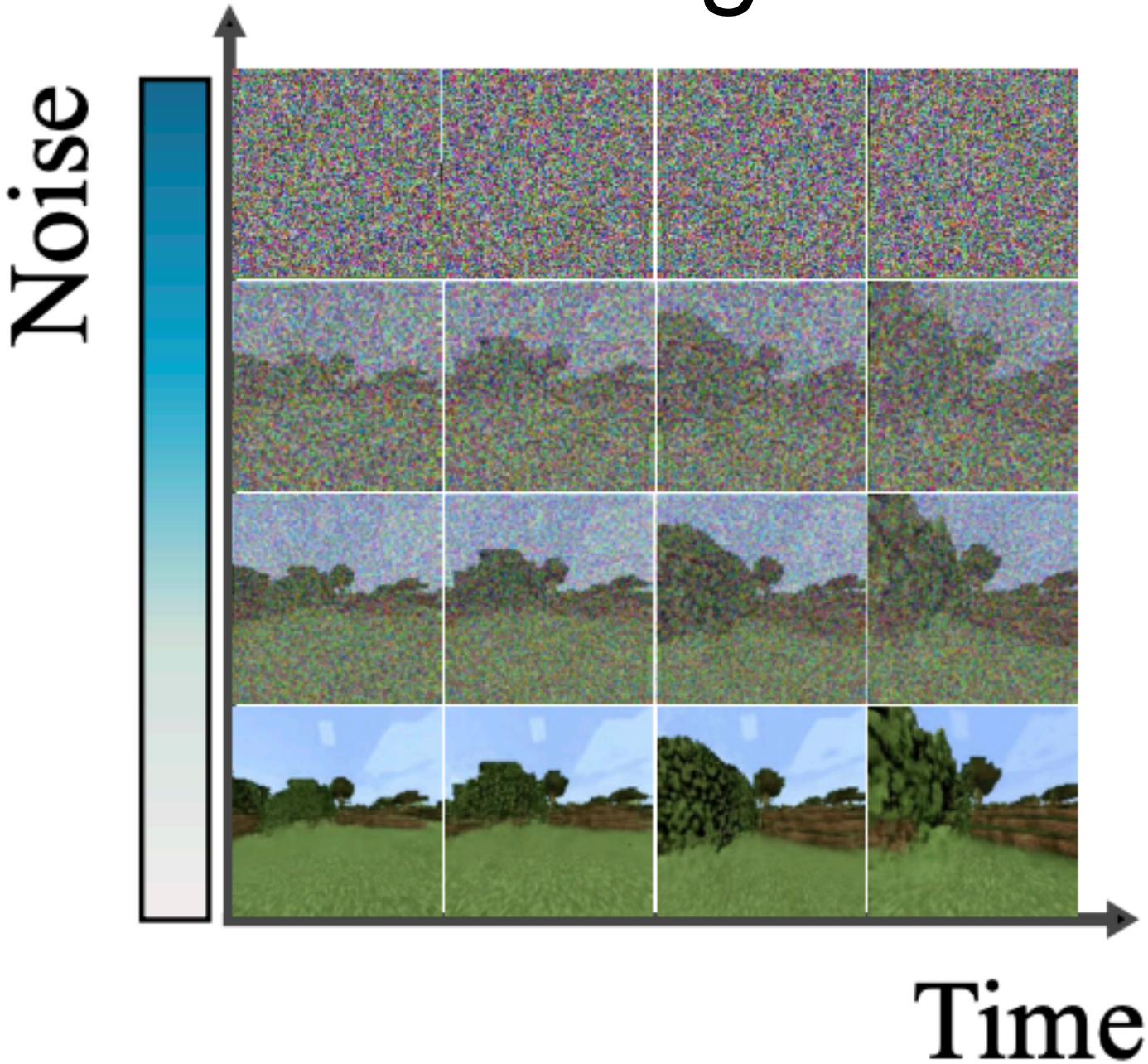
Fully Masked

Slightly Masked

Unmasked

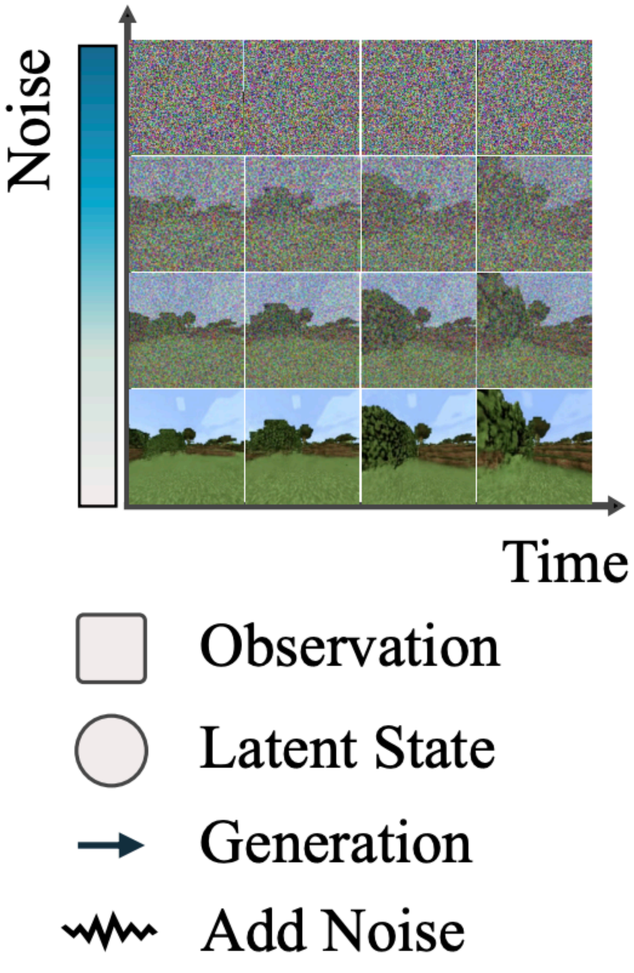


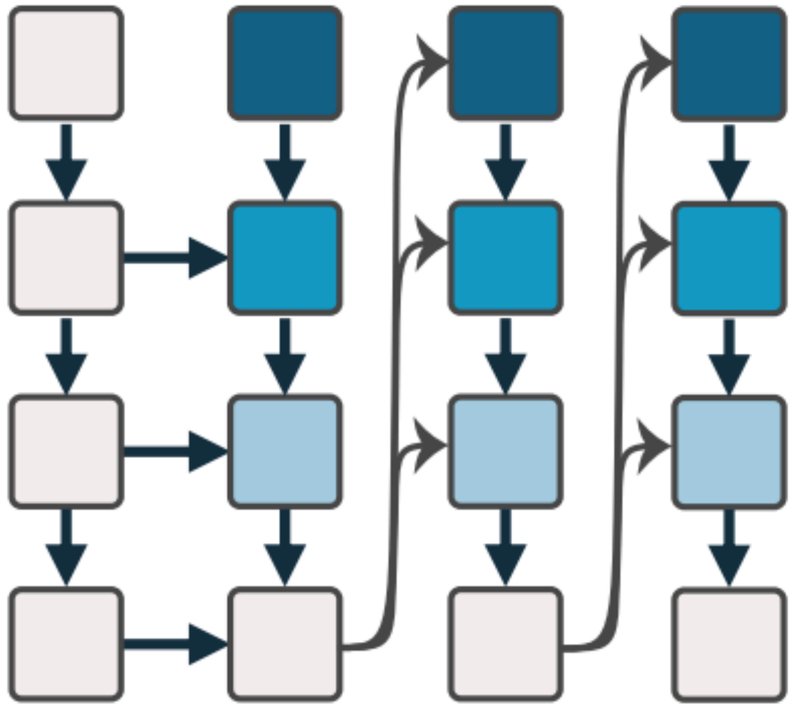
Preliminaries: Noise as Masking



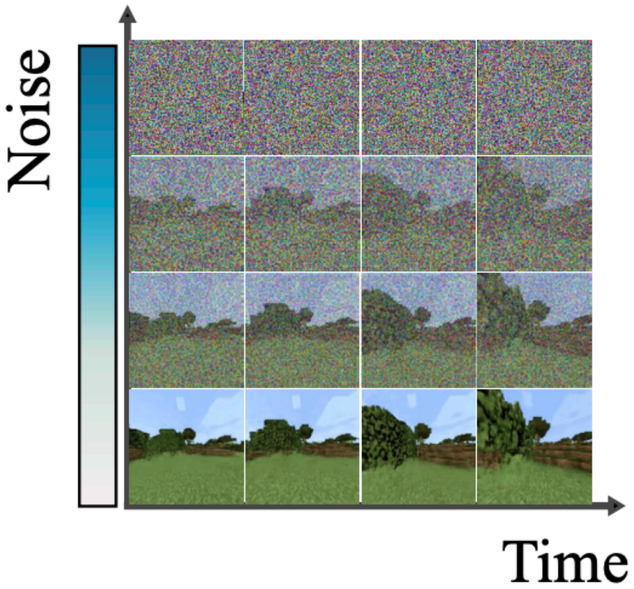
Two axes of
sequence generation

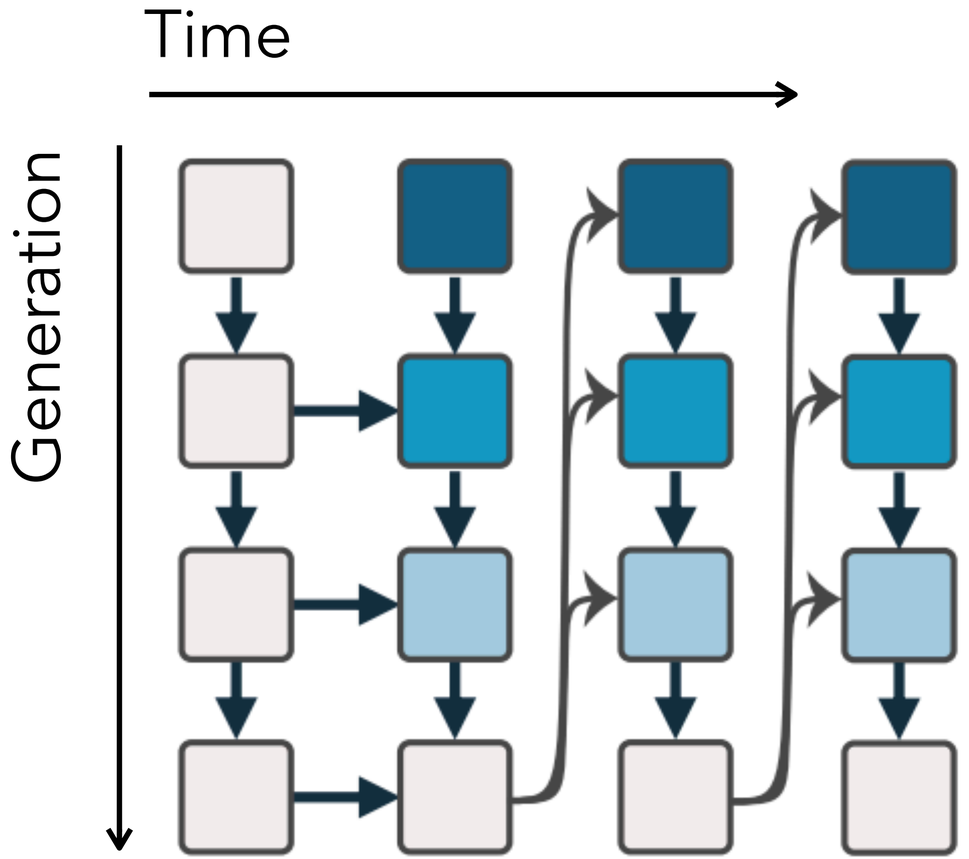
Next-Token Prediction





Next-Token Prediction





Observation

Latent State

Generation

Add Noise

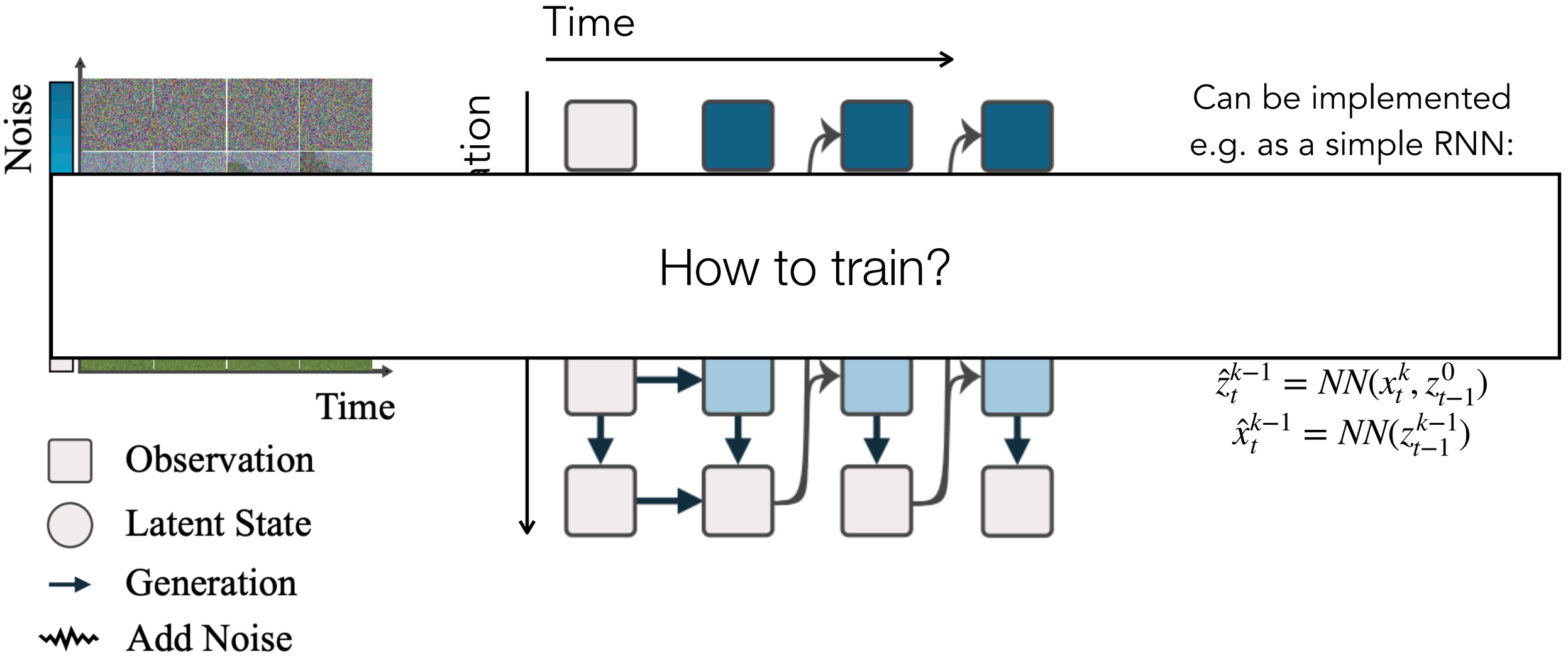
Can be implemented e.g. as a simple RNN:

$$\hat{x}_t^{k-1} = NN(x_t^k, x_{t-1}^0)$$

Or with latent states:

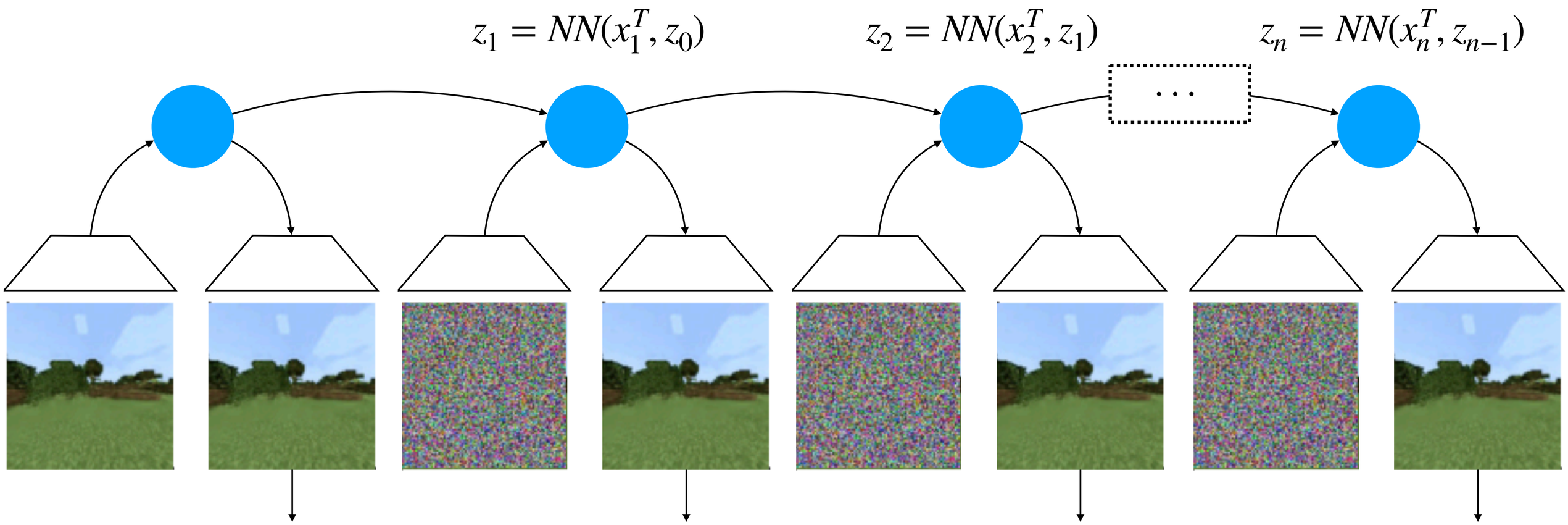
$$\hat{z}_t^{k-1} = NN(x_t^k, z_{t-1}^0)$$
$$\hat{x}_t^{k-1} = NN(z_{t-1}^{k-1})$$

Next-Token Prediction



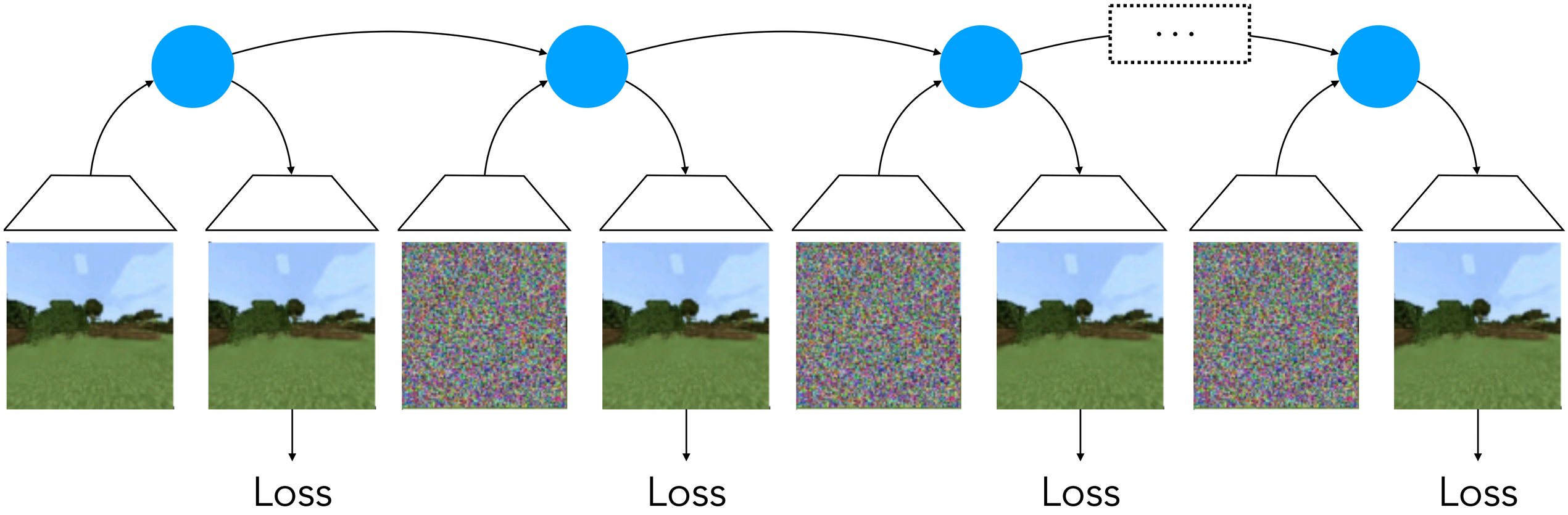
$$\hat{z}_t^{k-1} = NN(x_t^k, z_{t-1}^0)$$
$$\hat{x}_t^{k-1} = NN(z_{t-1}^{k-1})$$

“Free Running” Training

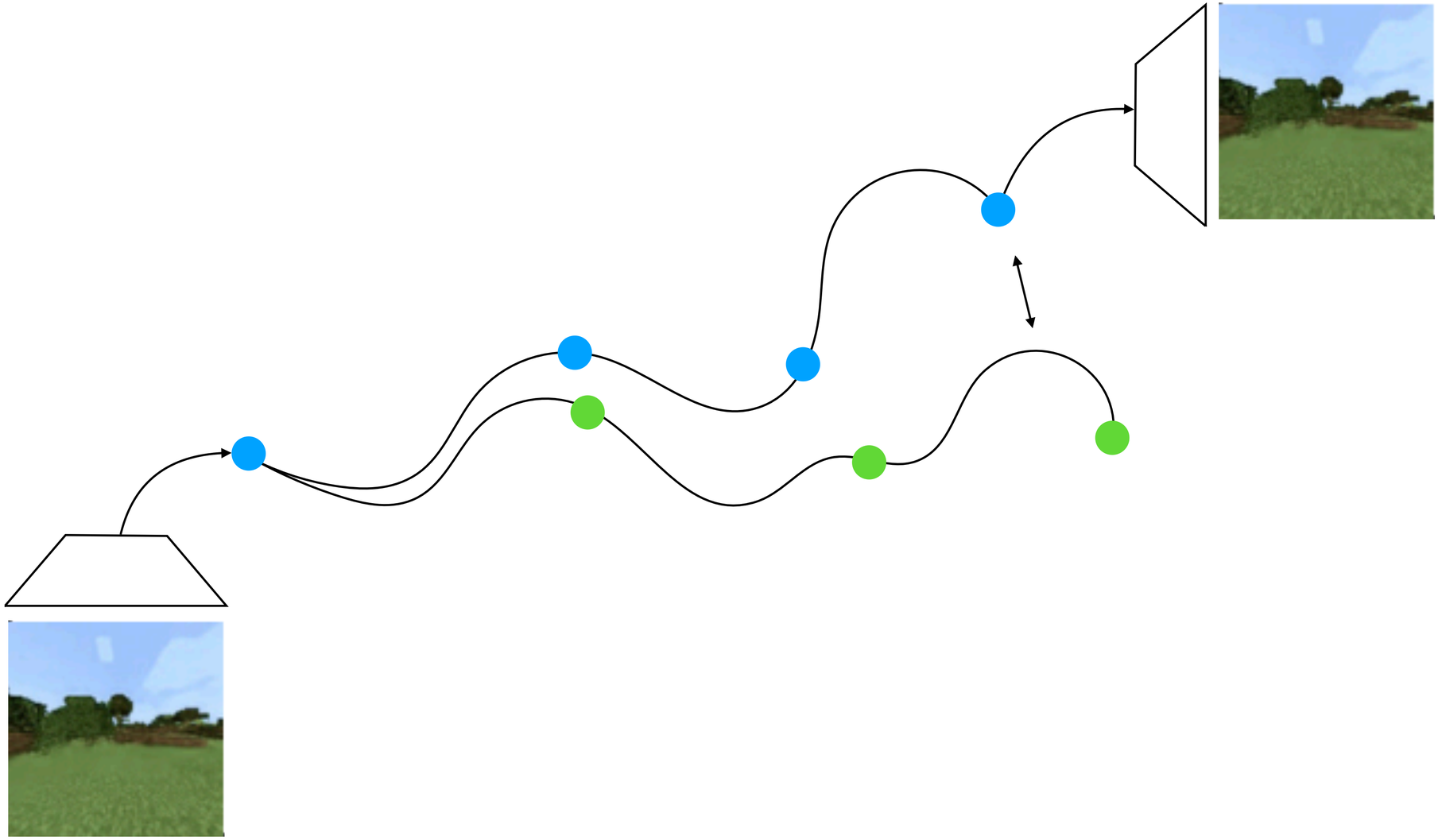


“Free Running” Training

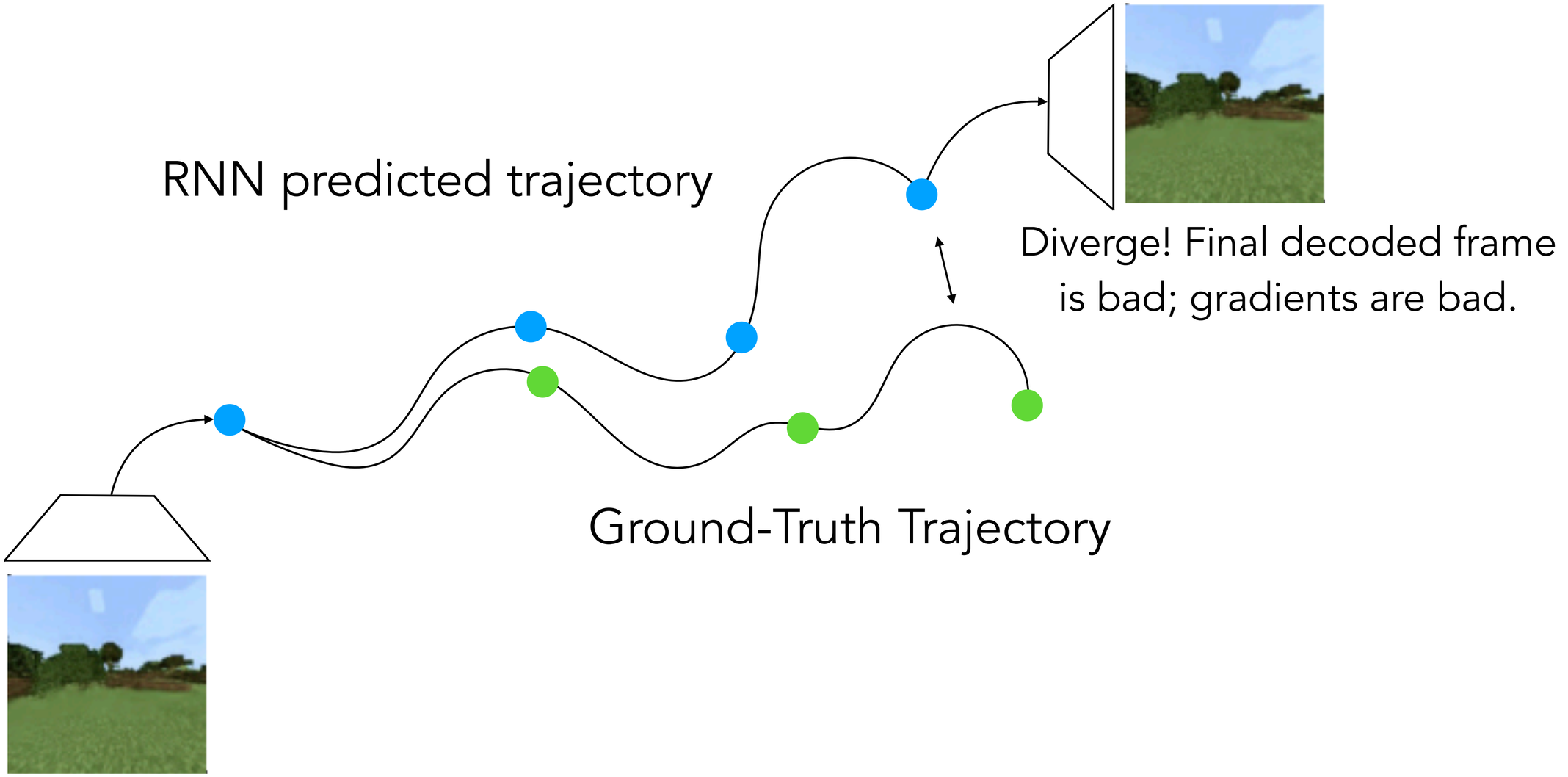
$z_0 = NN(x_0^0)$ $z_1 = NN(x_1^T, z_0)$ $z_2 = NN(x_2^T, z_1)$ $z_n = NN(x_n^T, z_{n-1})$



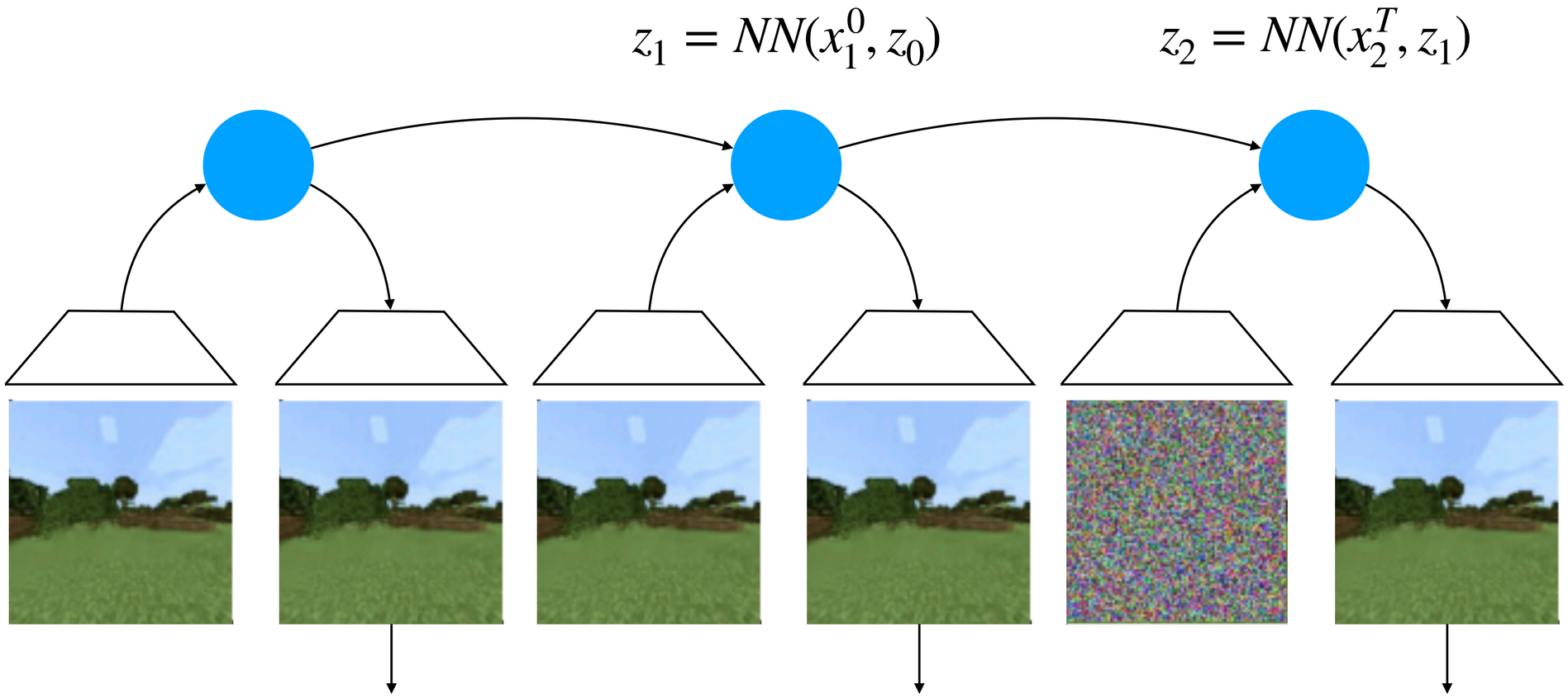
The Problem with Free Running



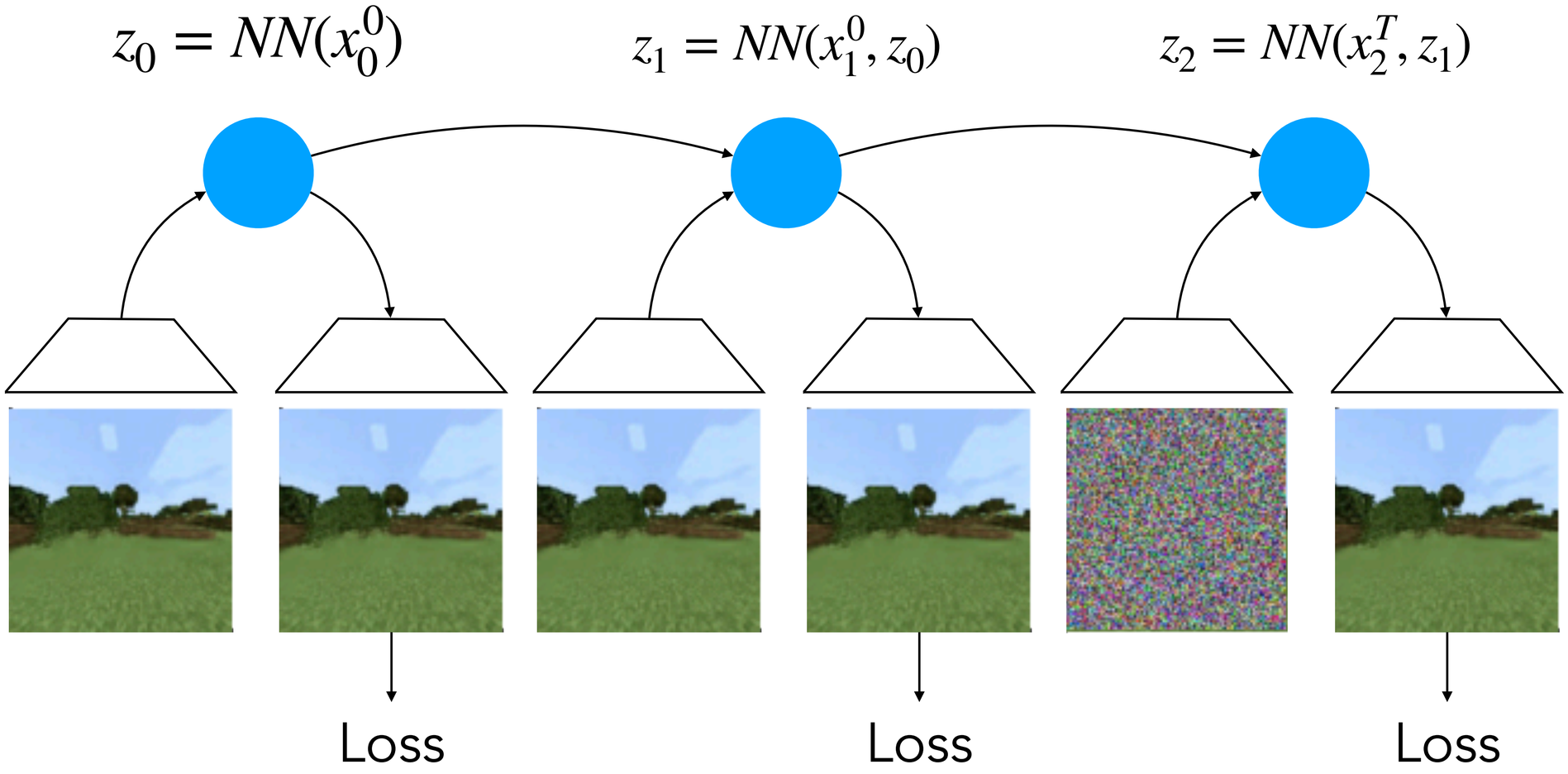
The Problem with Free Running



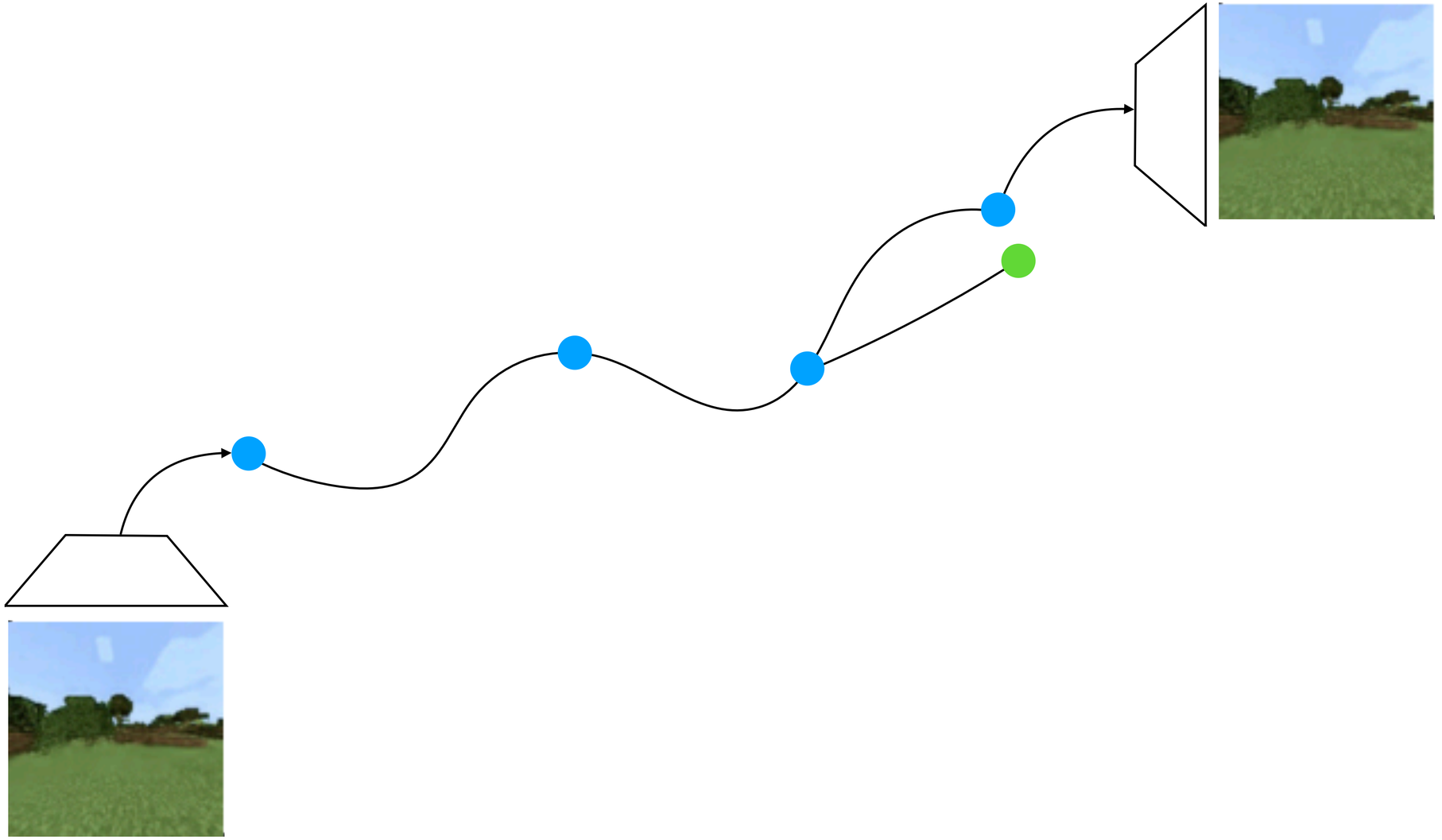
Teacher Forcing Training: Only Final Frame is masked, all intermediates are ground-truth



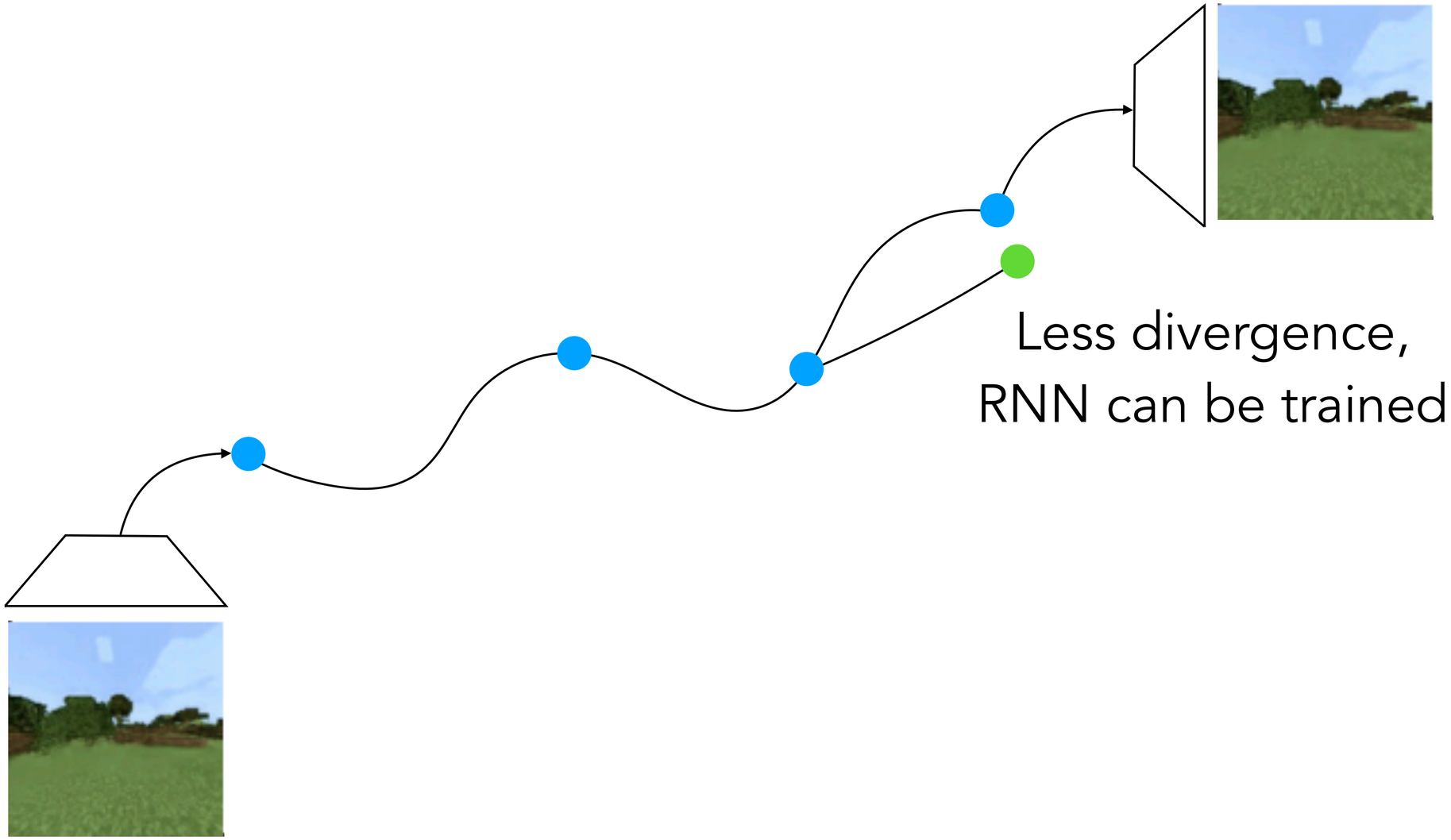
Teacher Forcing Training: Only Final Frame is masked, all intermediates are ground-truth



Teacher Forcing



Teacher Forcing



Problem with Teacher Forcing: Test-Time is OOD!

Next-token prediction

compounding error

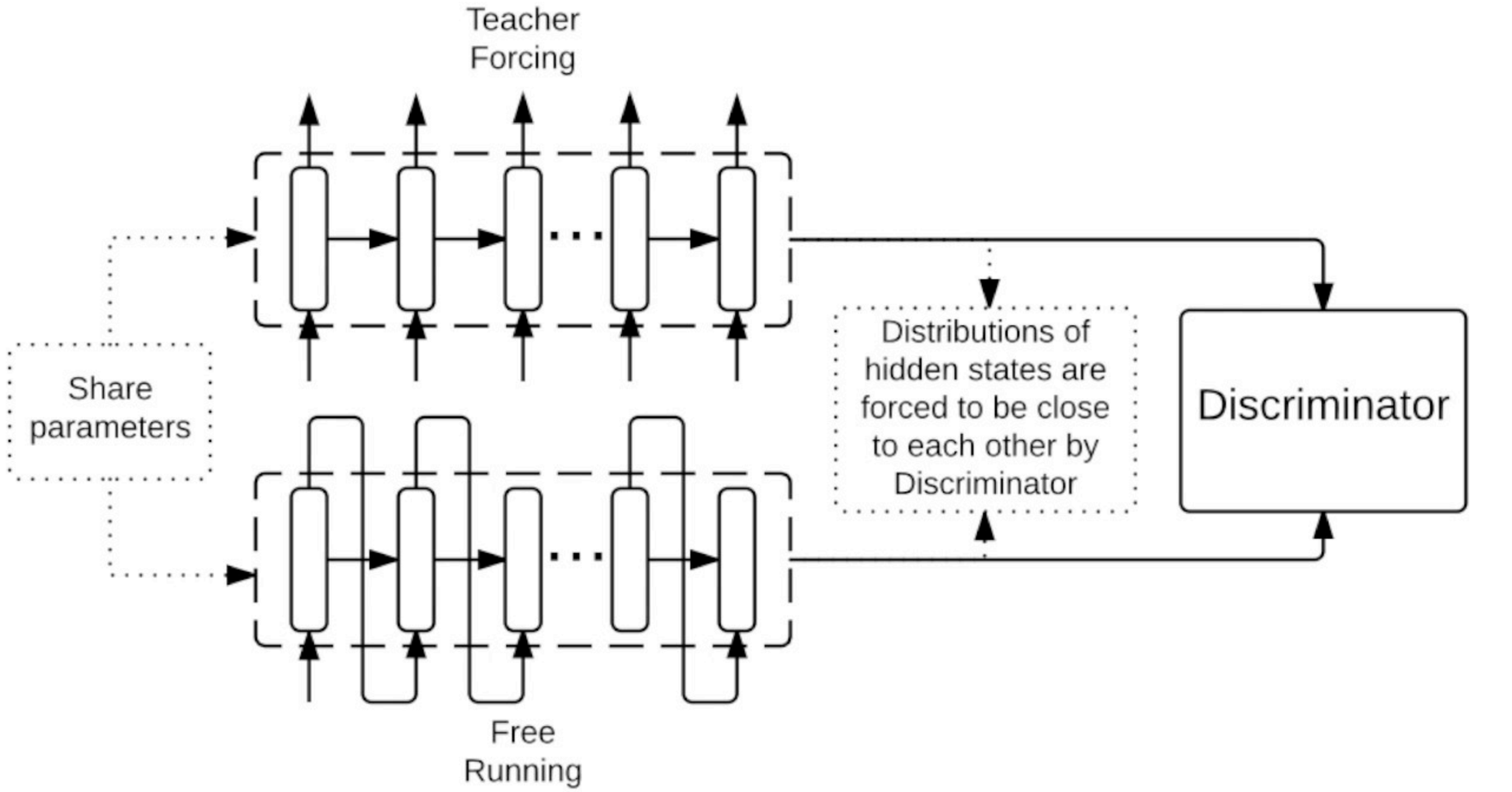
Problem with Teacher Forcing: Test-Time is OOD!

Next-token prediction

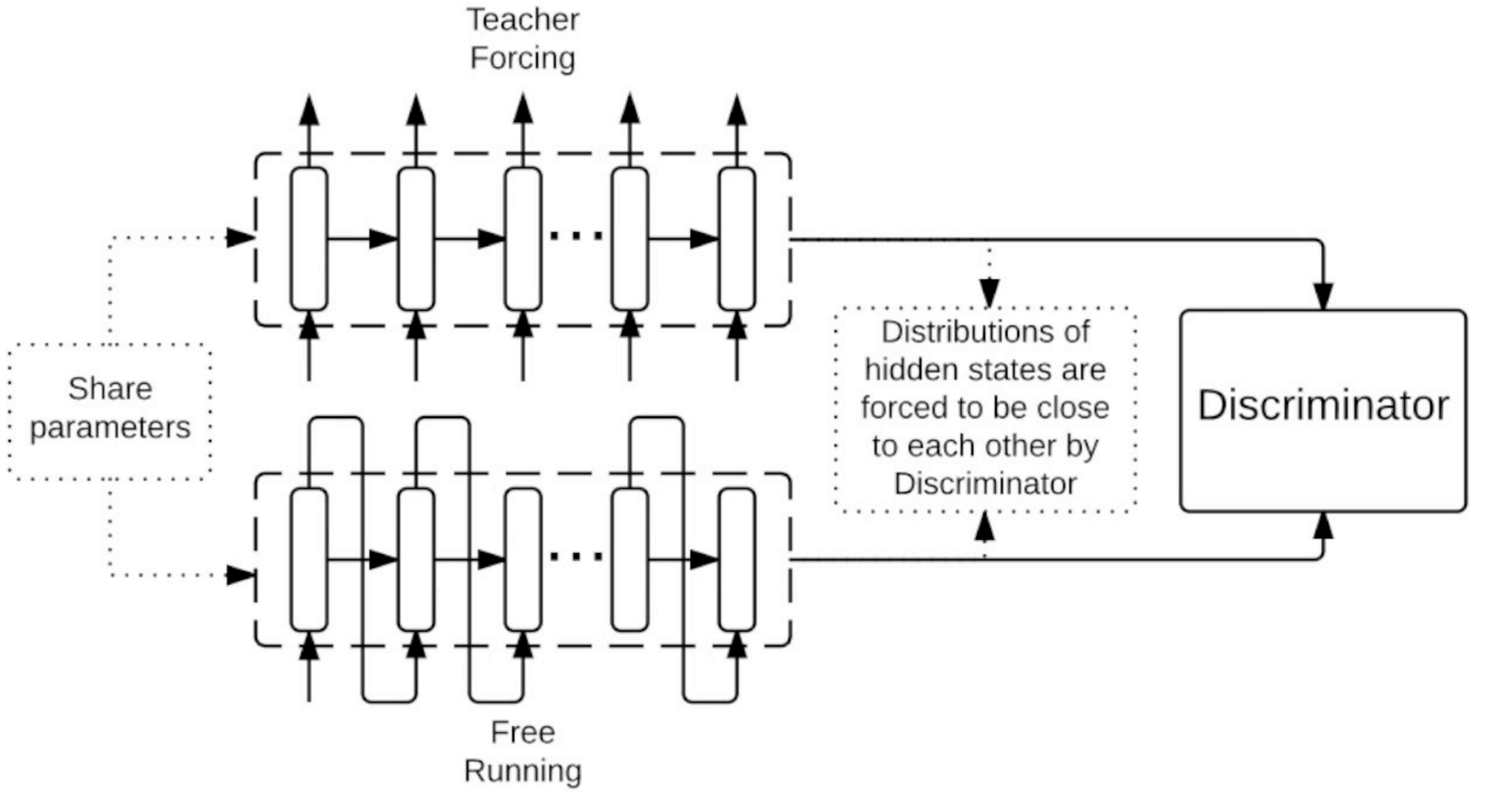
compounding error

Problem: at test time, *have* to use RNN predictions! Quickly diverges.

Prior attempts at fixing this

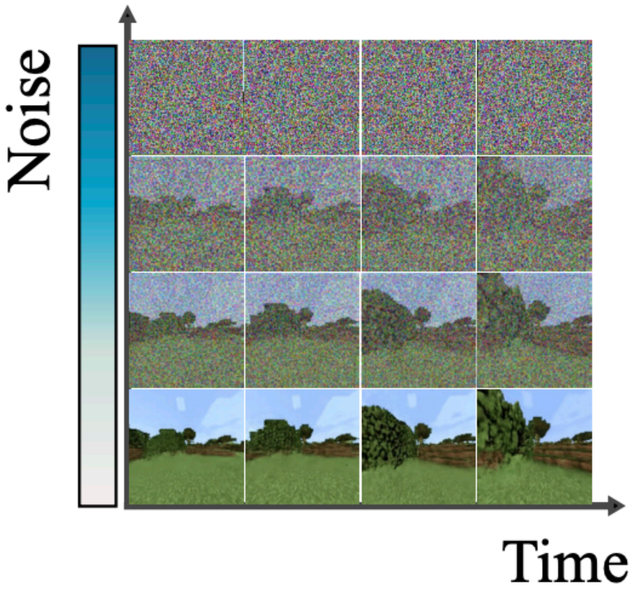


Prior attempts at fixing this



[Professor Forcing, Lamb et al. 2016]

Auto-Regressive Generation (Next-Token Prediction)



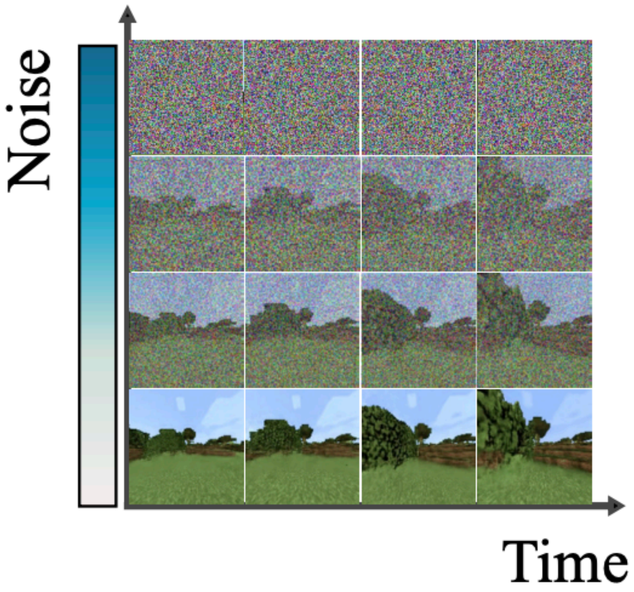
Observation

Latent State

Generation

Add Noise

Auto-Regressive Generation (Next-Token Prediction)



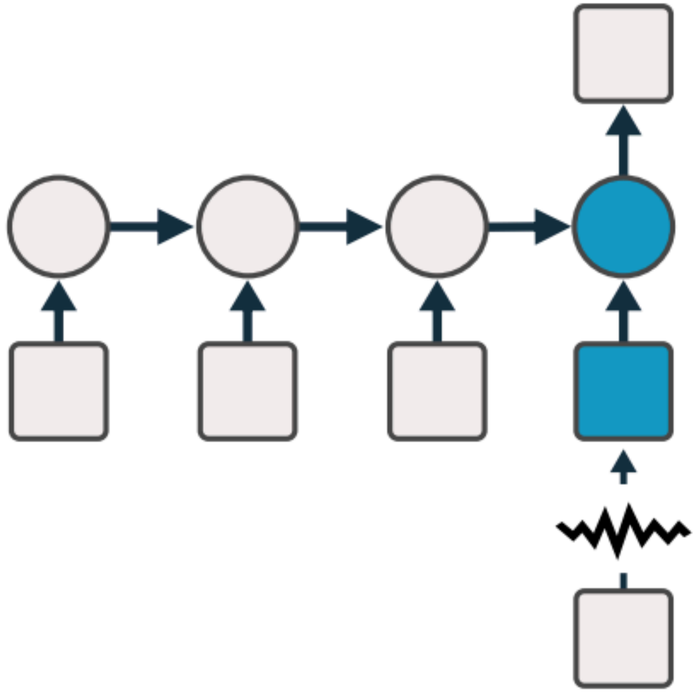
Observation

Latent State

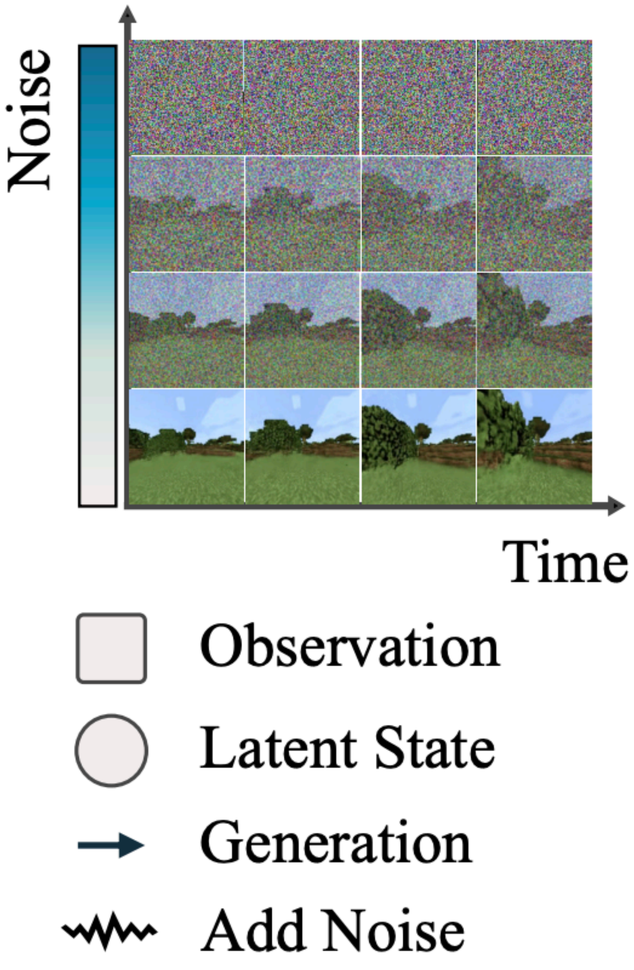
Generation

Add Noise

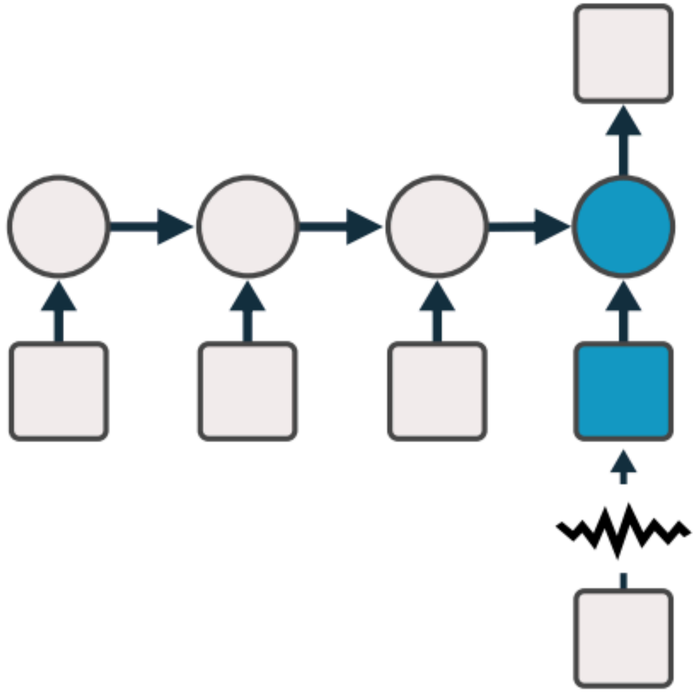
Training



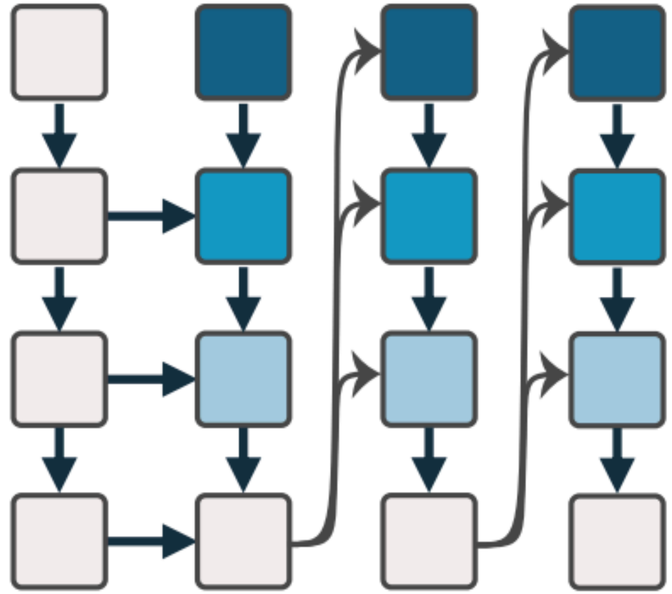
Auto-Regressive Generation (Next-Token Prediction)



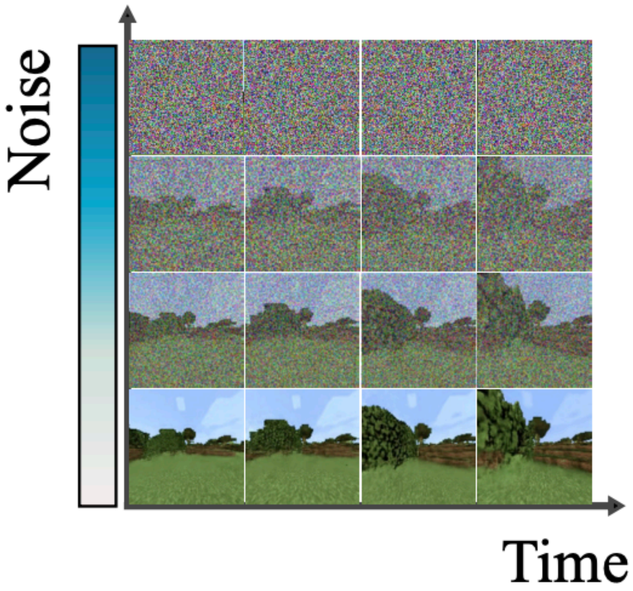
Training



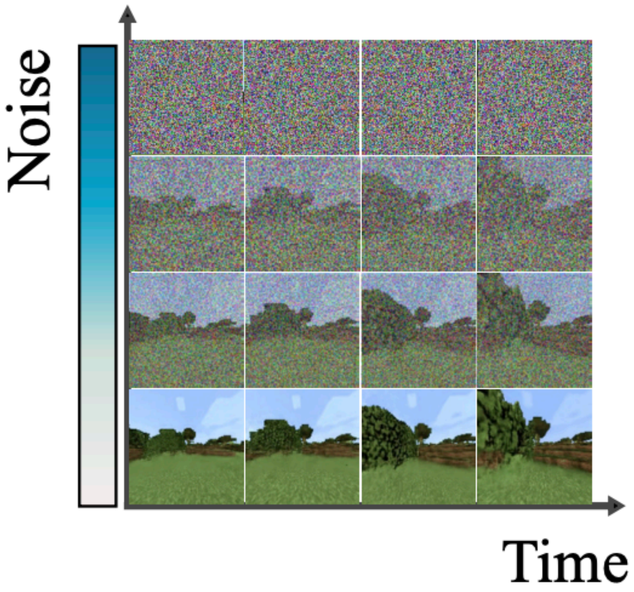
Inference



Full-Sequence Diffusion



Full-Sequence Diffusion



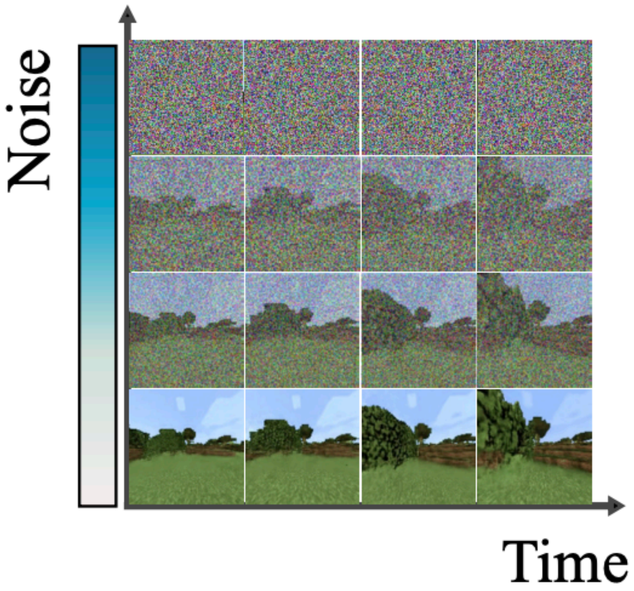
Observation

Latent State

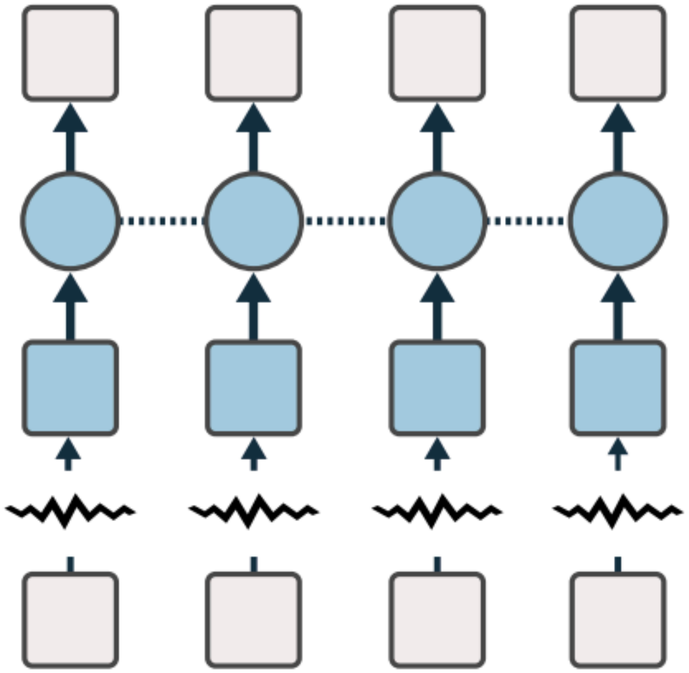
Generation

Add Noise

Full-Sequence Diffusion



Training



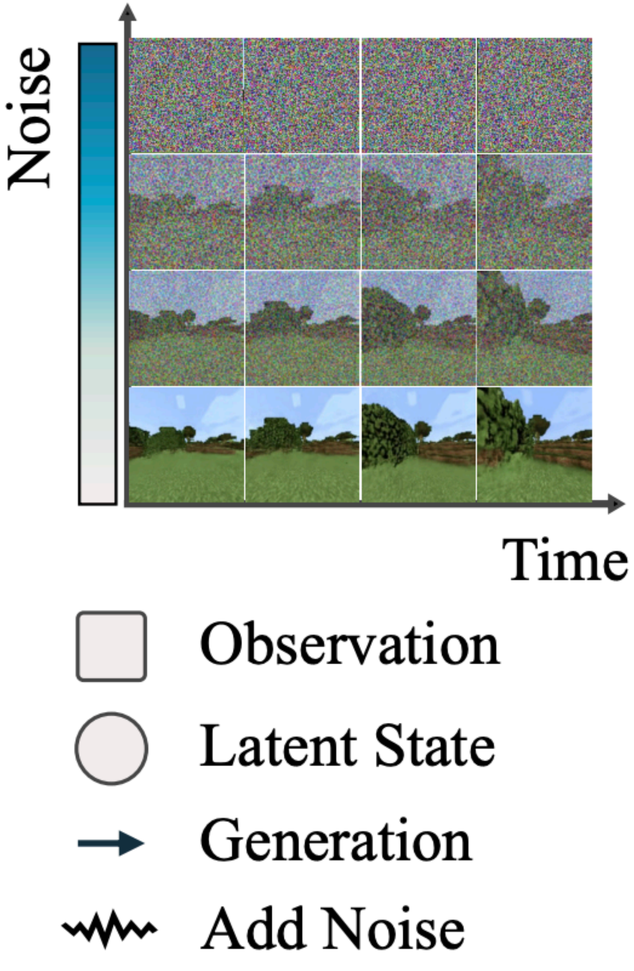
Observation

Latent State

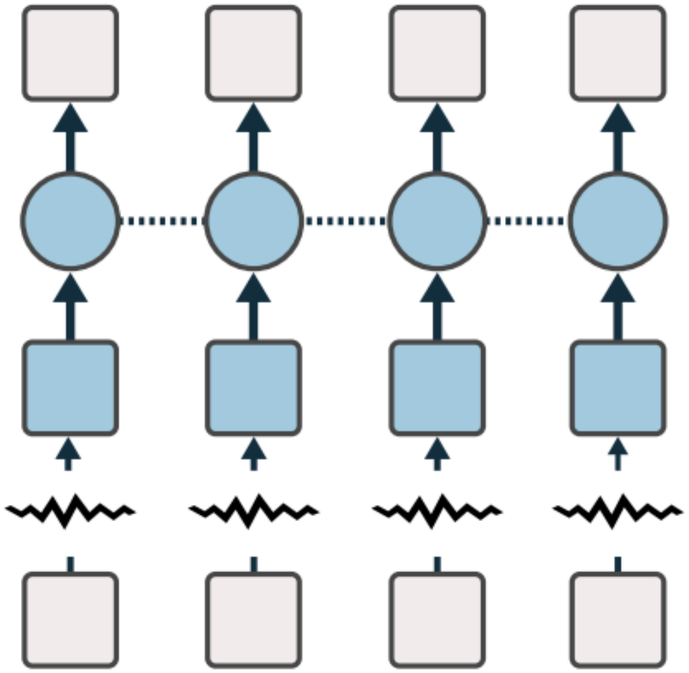
Generation

Add Noise

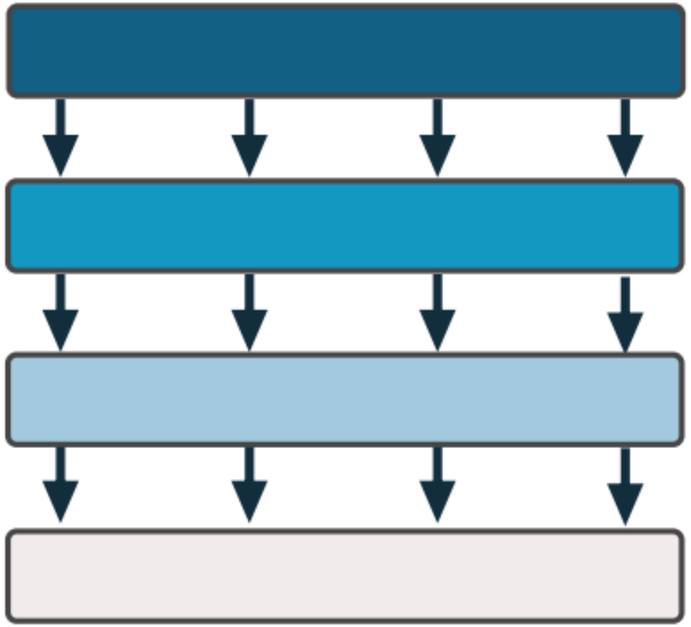
Full-Sequence Diffusion



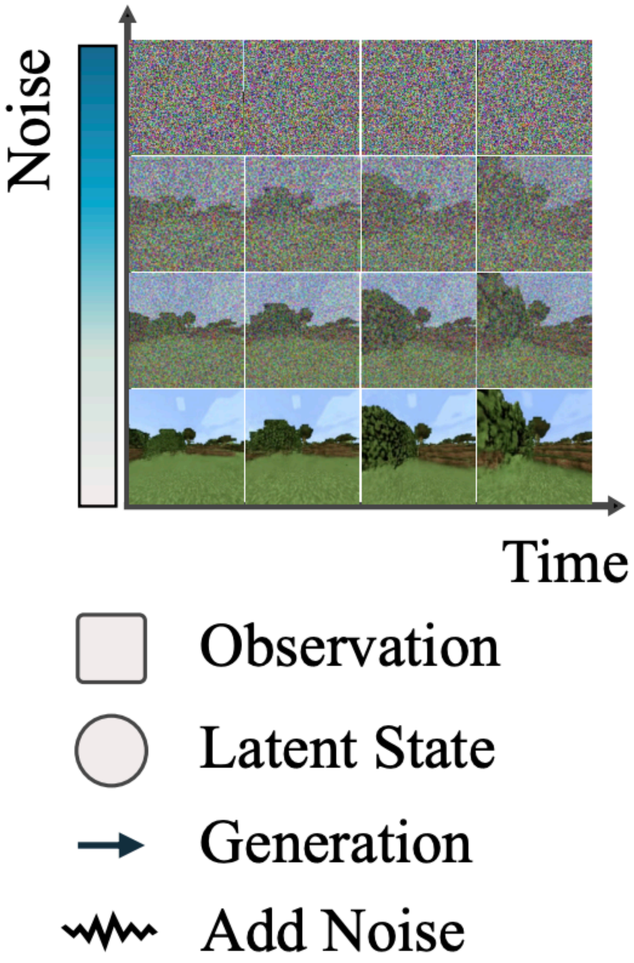
Training



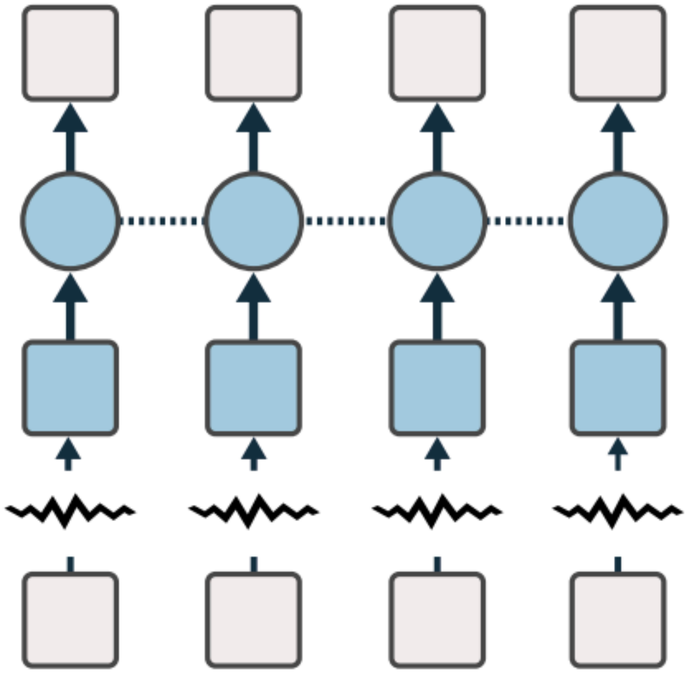
Inference



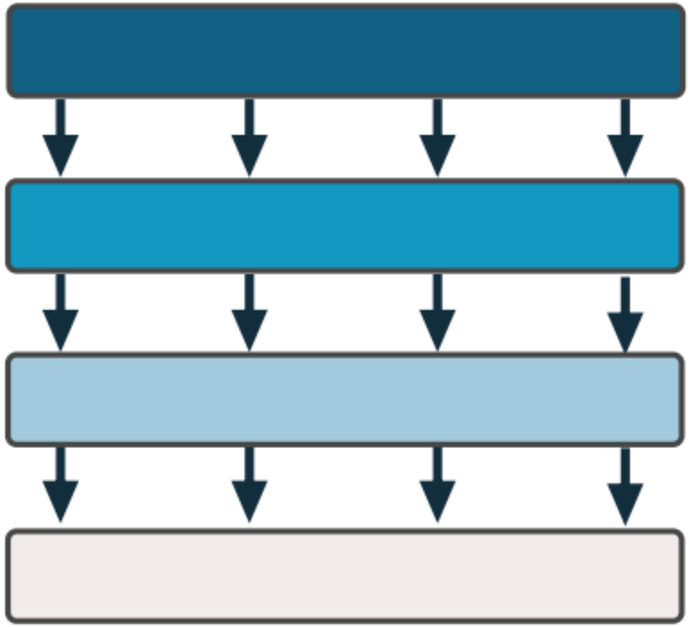
Full-Sequence Diffusion



Training



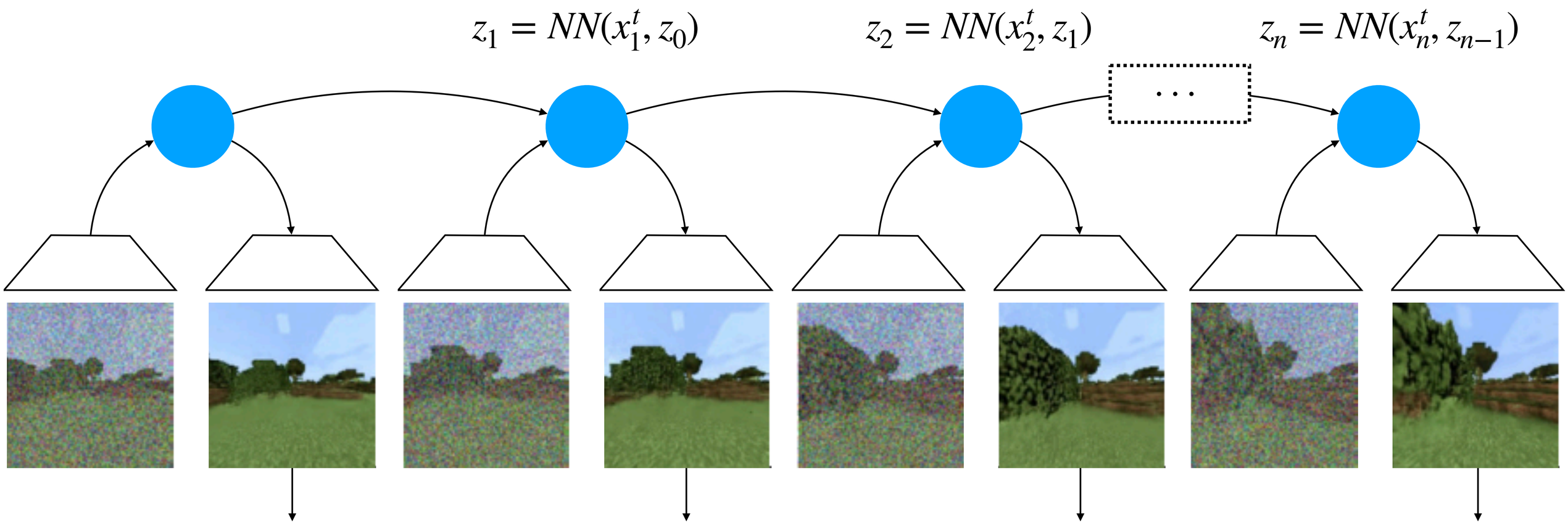
Inference



Examples: SORA, Meta Movie Gen, Diffusion Probabilistic Modeling for Video Generation, [Yang et al. 2022], Video Diffusion Models [Ho et al. 2022]

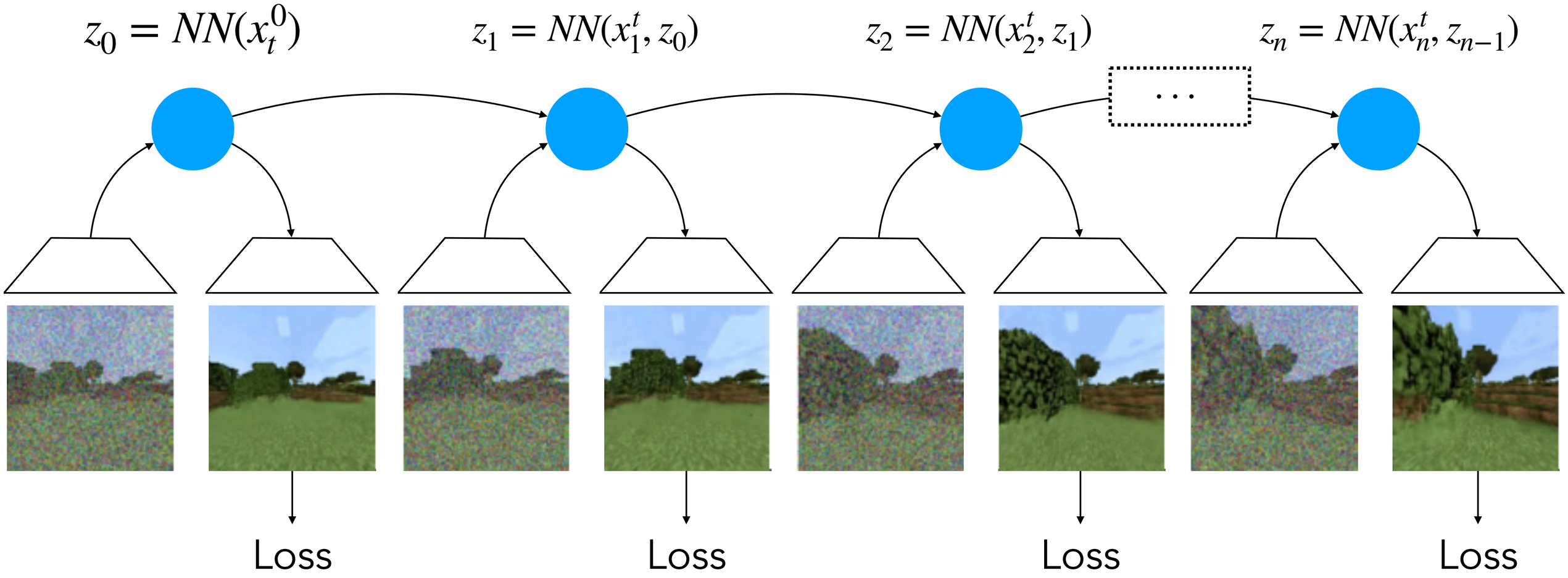
Full-Sequence Diffusion Training

With RNN



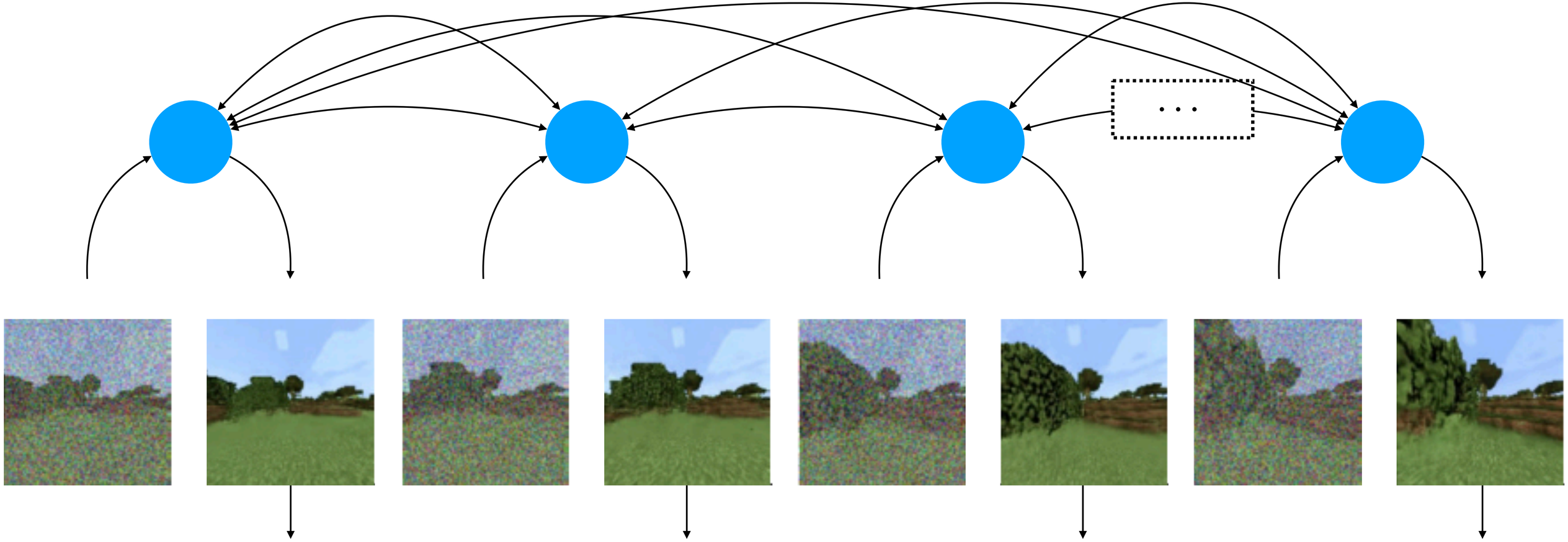
Full-Sequence Diffusion Training

With RNN



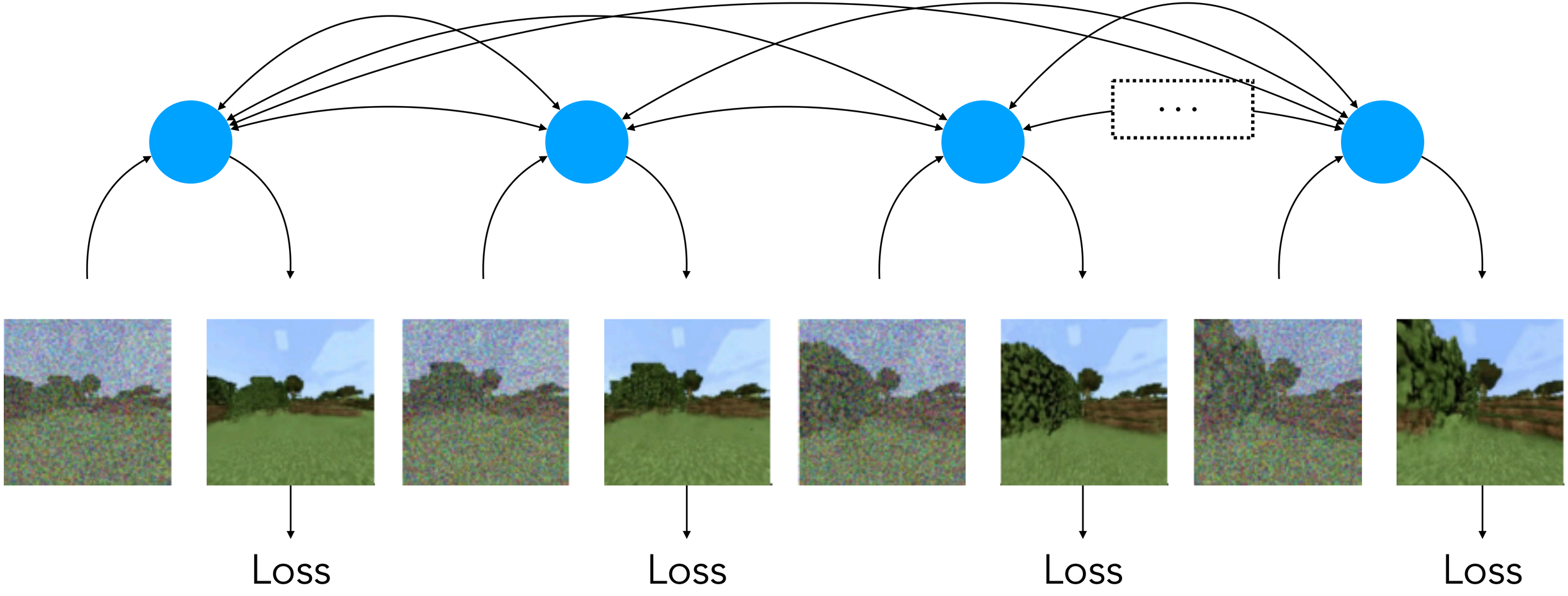
Full-Sequence Diffusion Training

With Transformers

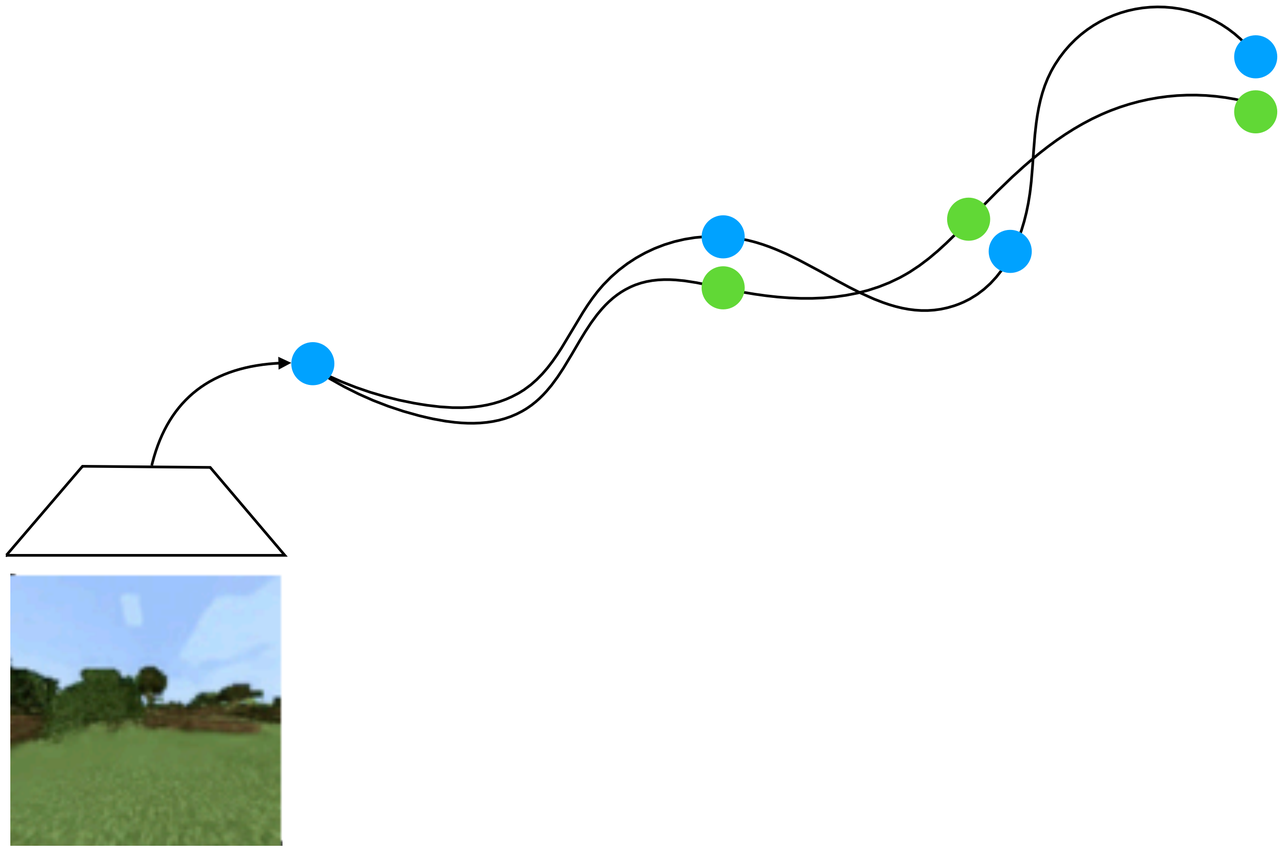


Full-Sequence Diffusion Training

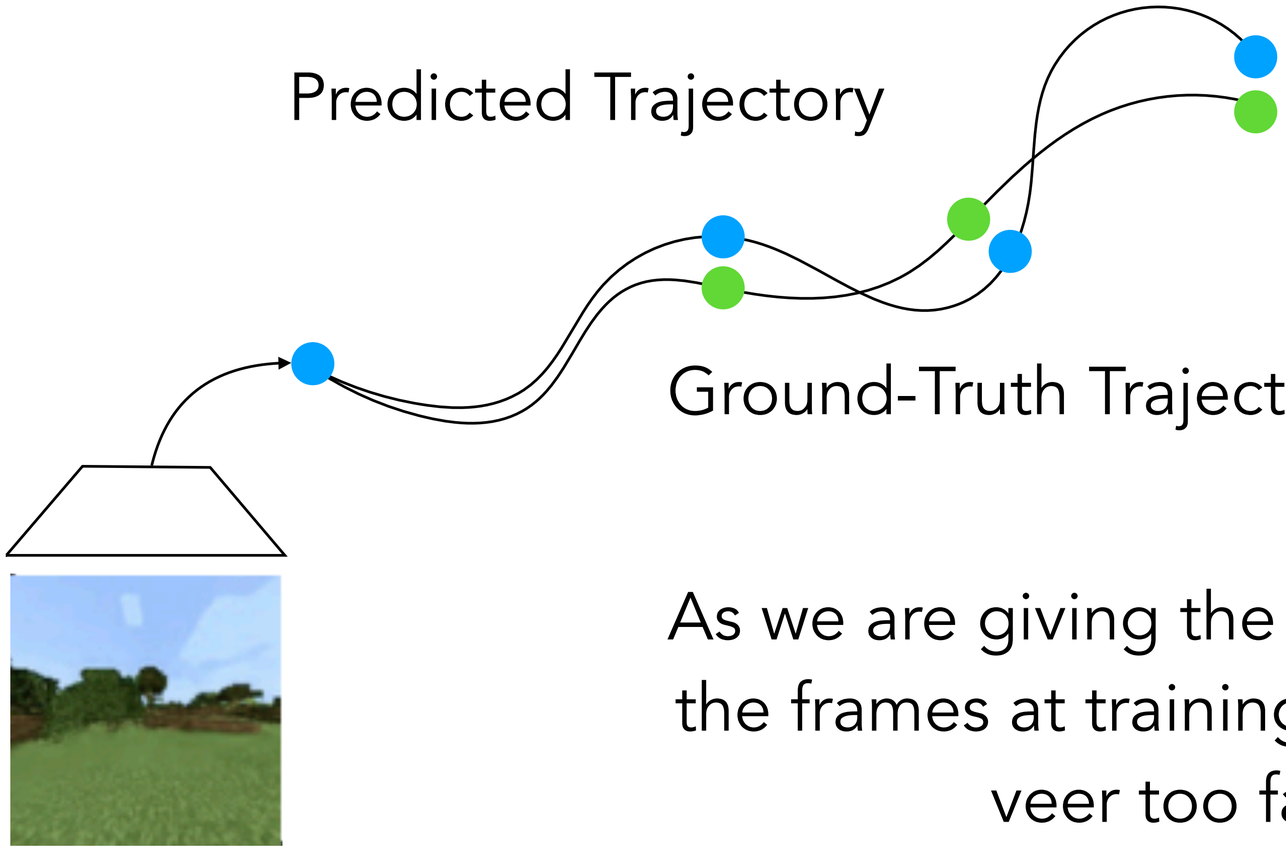
With Transformers



Full-Sequence Diffusion Training

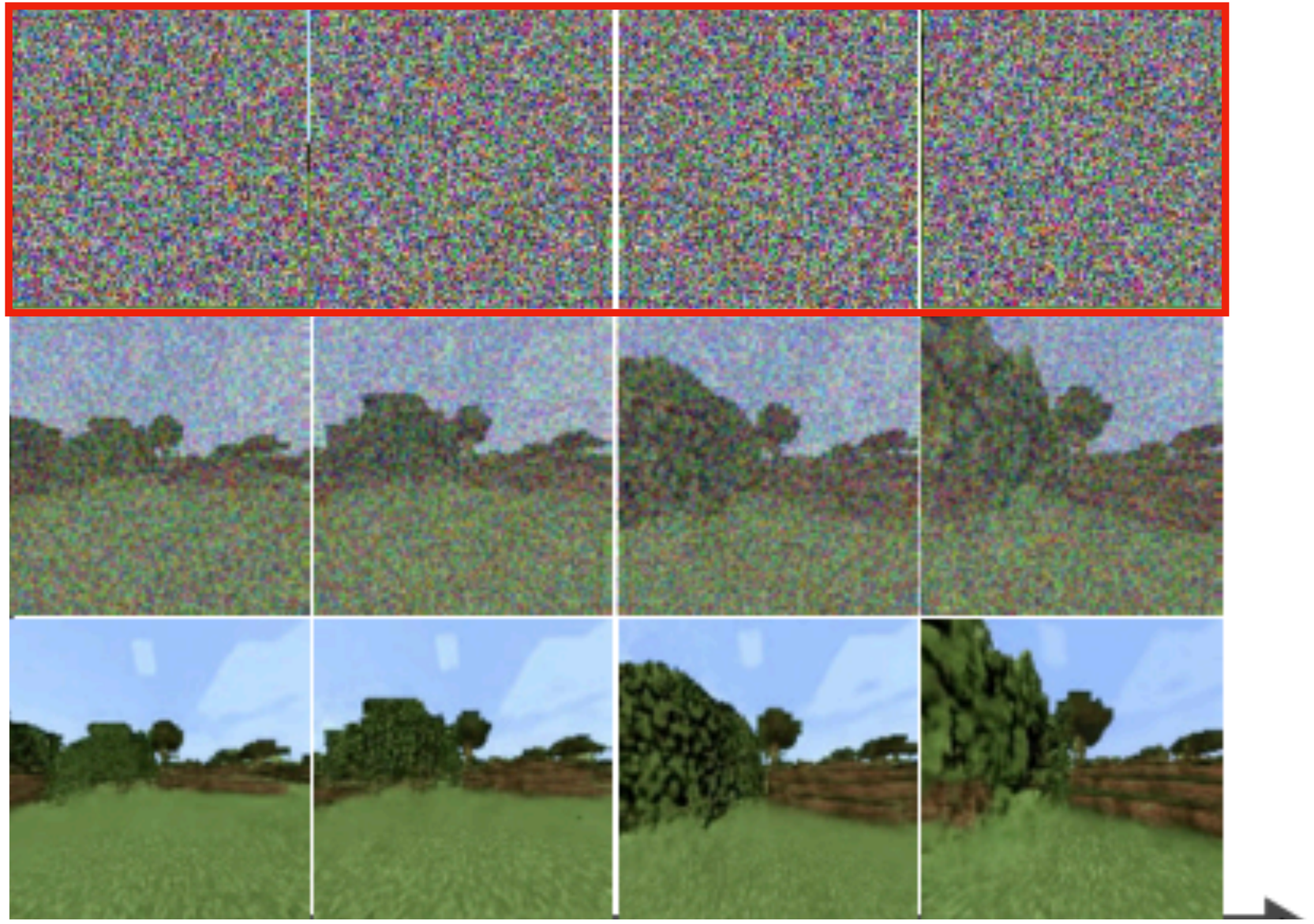


Full-Sequence Diffusion Training

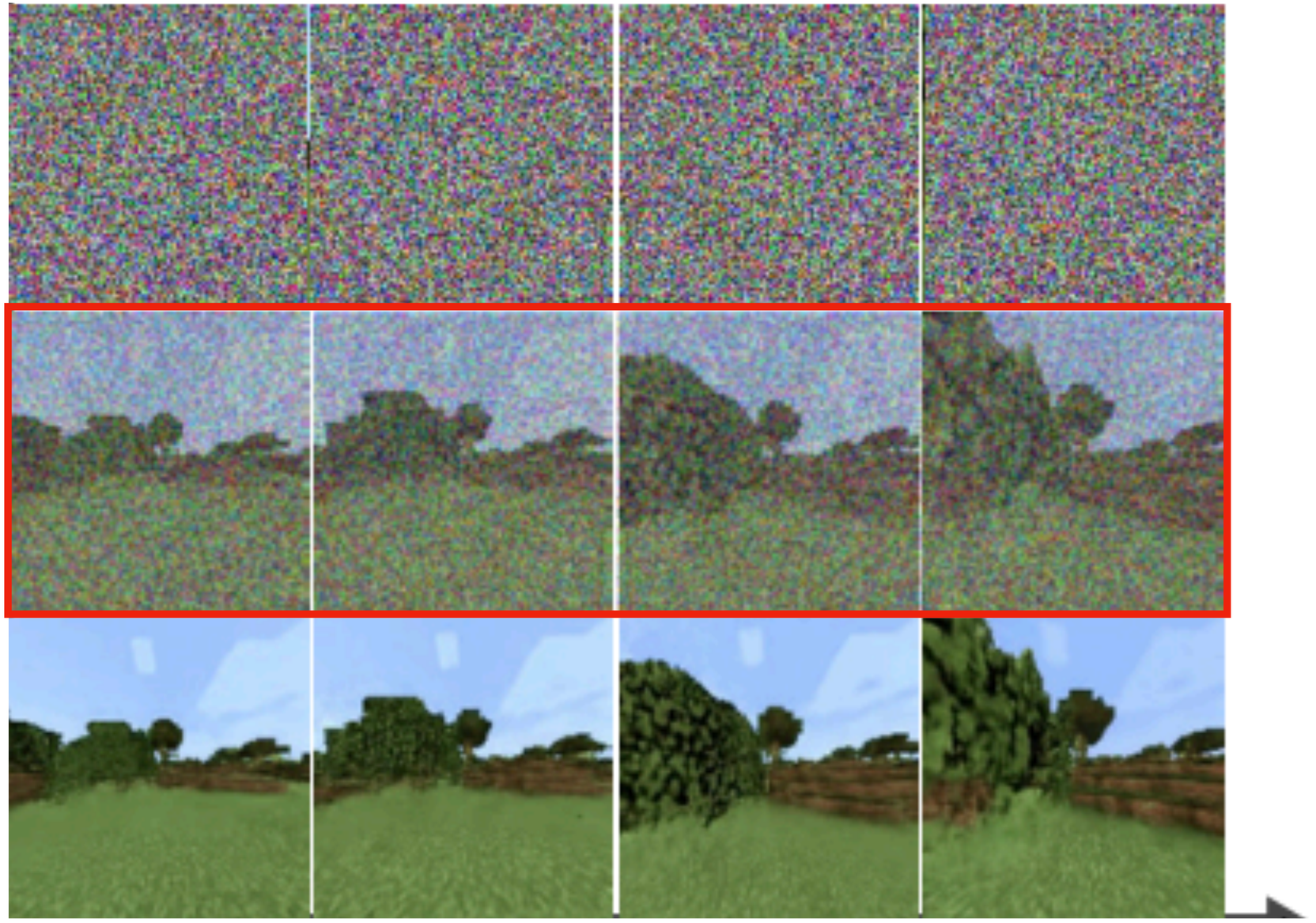


As we are giving the model a noisy version of all the frames at training time, its predictions won't veer too far from the truth.

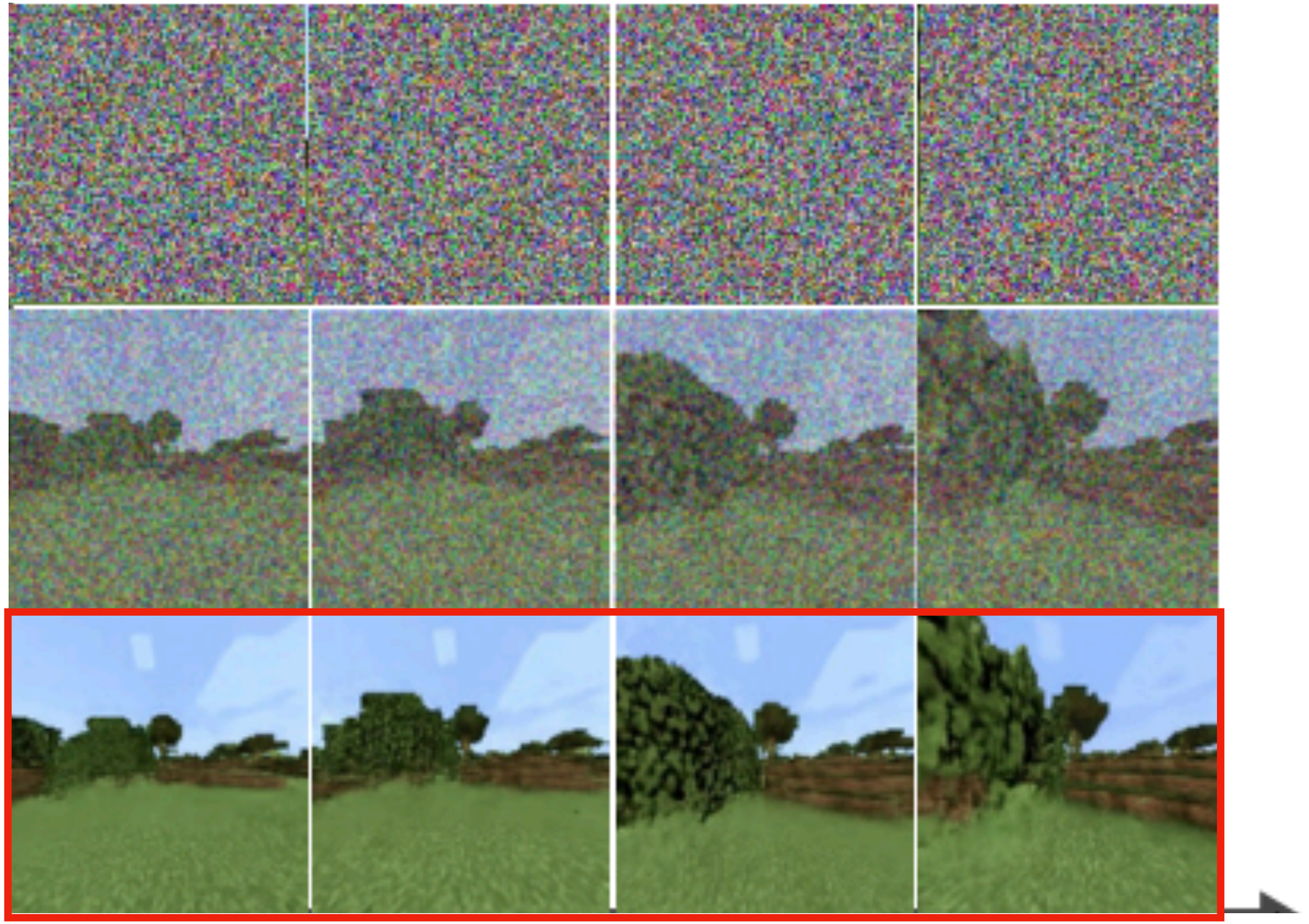
The problem with full-sequence Diffusion: Auto-Regressive Rollout



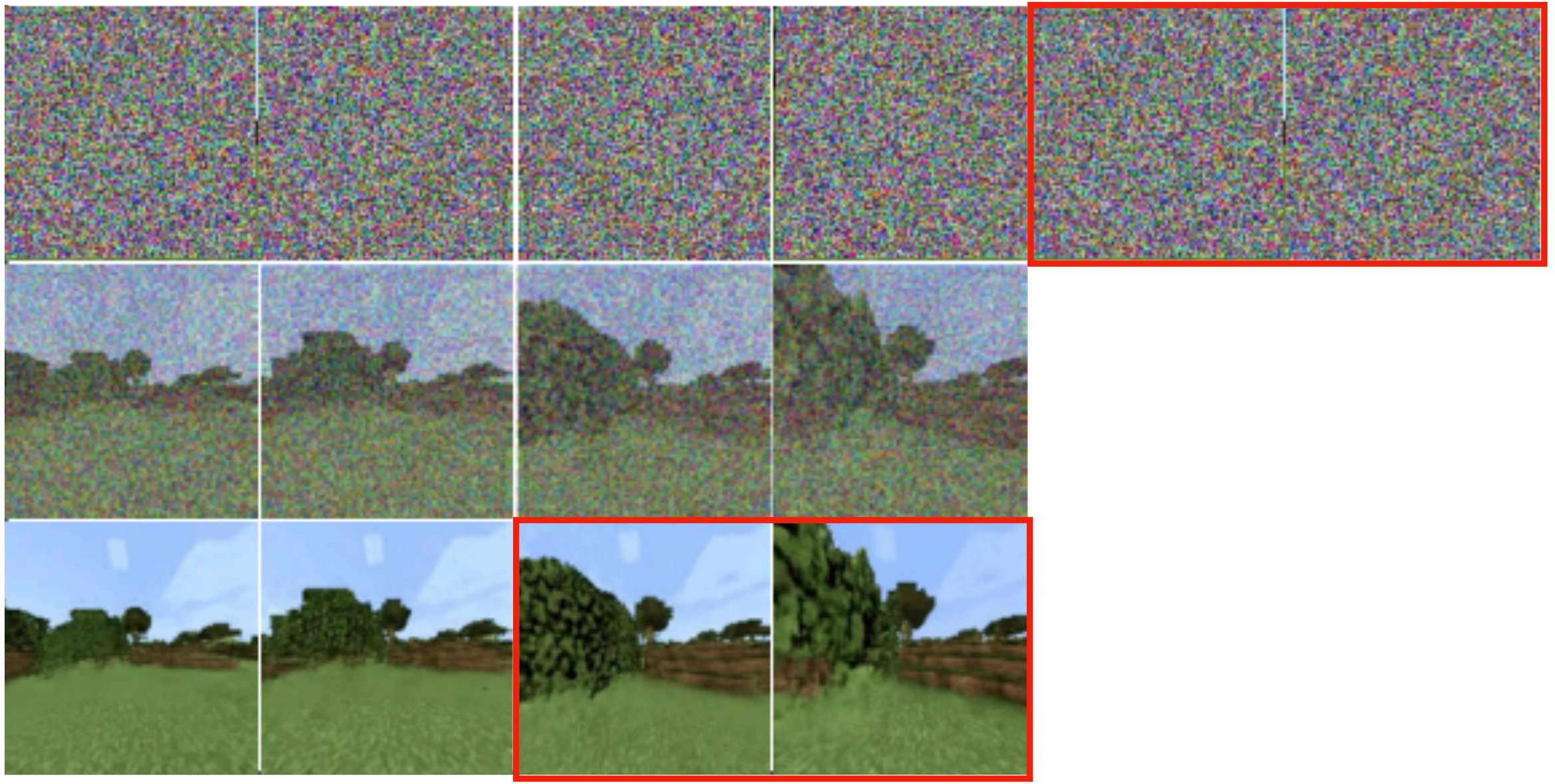
The problem with full-sequence Diffusion: Auto-Regressive Rollout



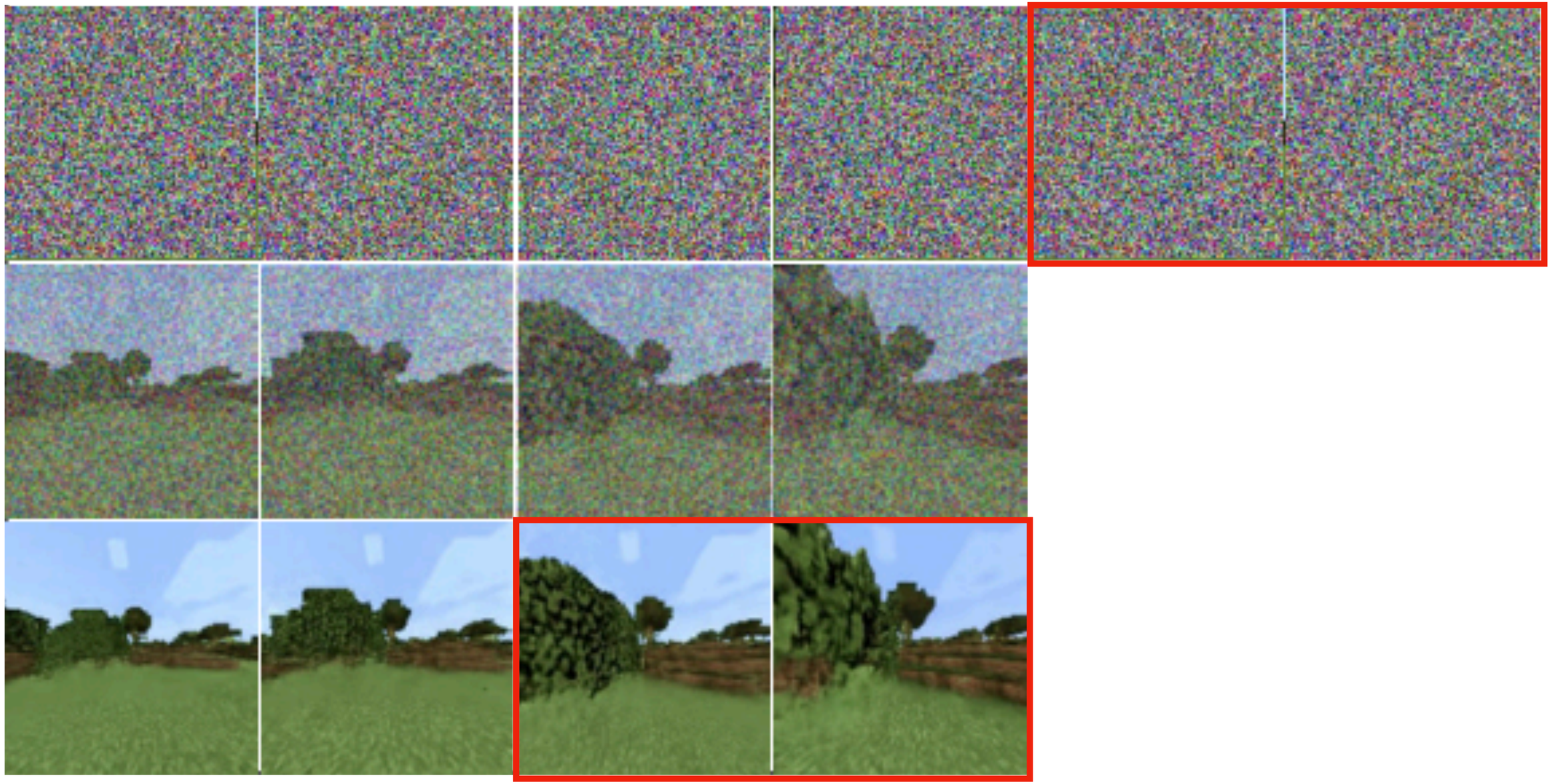
The problem with full-sequence Diffusion: Auto-Regressive Rollout



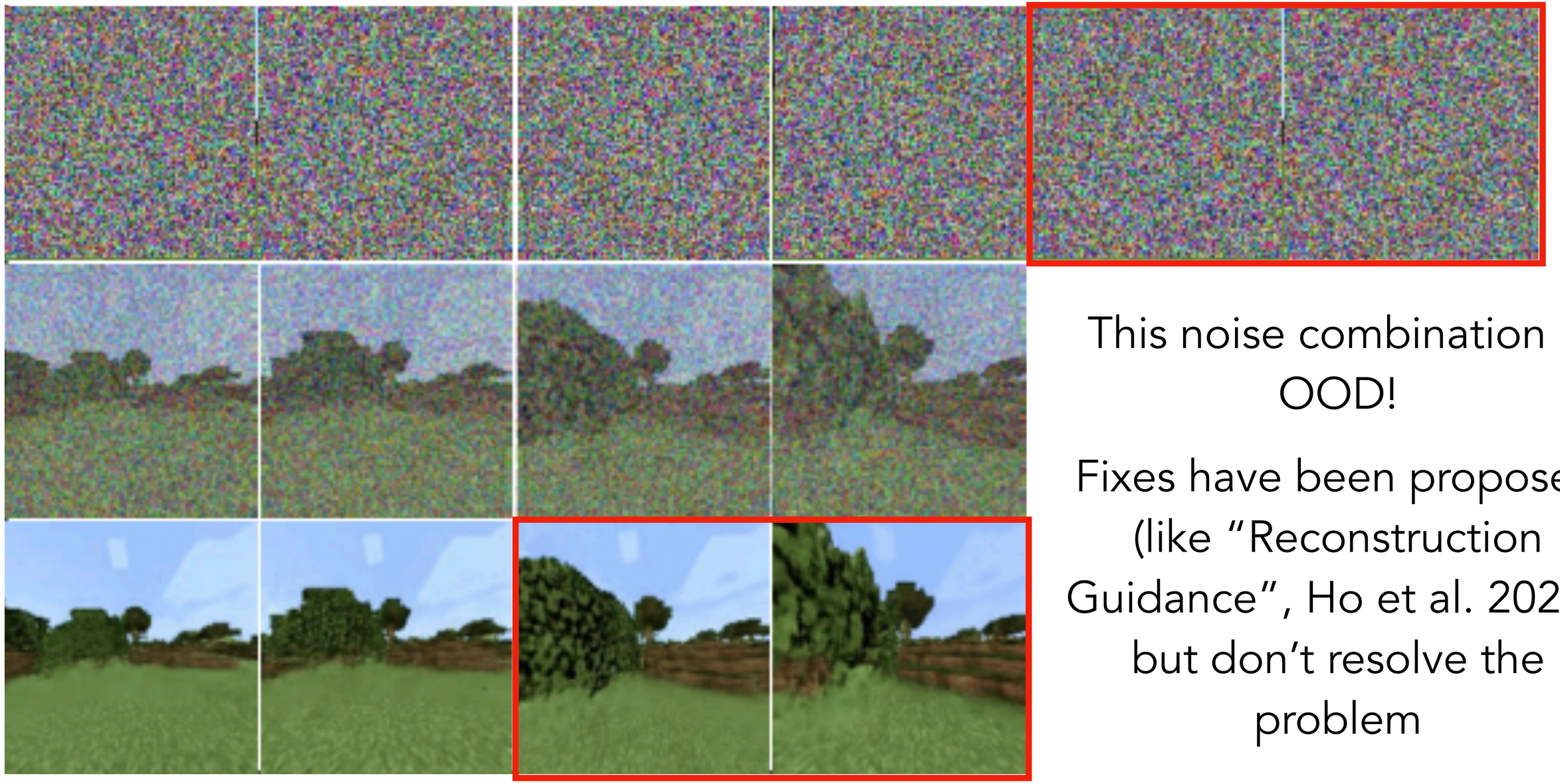
The problem with full-sequence Diffusion: Auto-Regressive Rollout



The problem with full-sequence Diffusion: Auto-Regressive Rollout



The problem with full-sequence Diffusion: Auto-Regressive Rollout



This noise combination is OOD!

Fixes have been proposed (like “Reconstruction Guidance”, Ho et al. 2020), but don’t resolve the problem

Auto-Regressive Rollout

Next-token prediction

compounding error

Full-Sequence Diffusion

consistency issues

Auto-Regressive Rollout

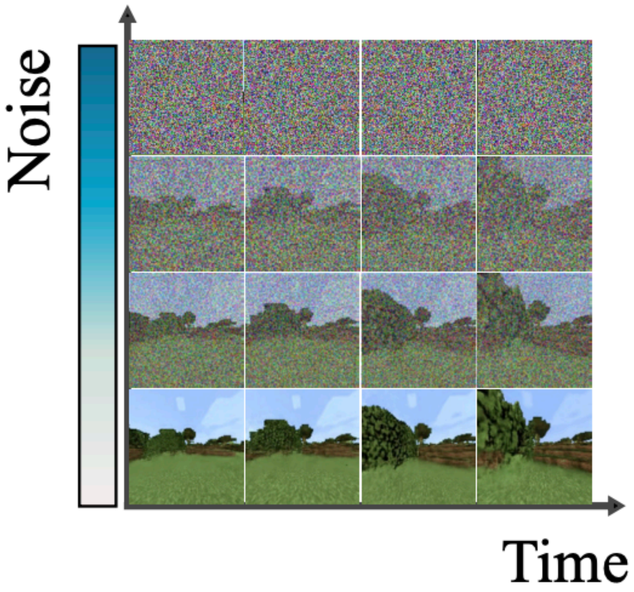
Next-token prediction

compounding error

Full-Sequence Diffusion

consistency issues

Full-Sequence Diffusion



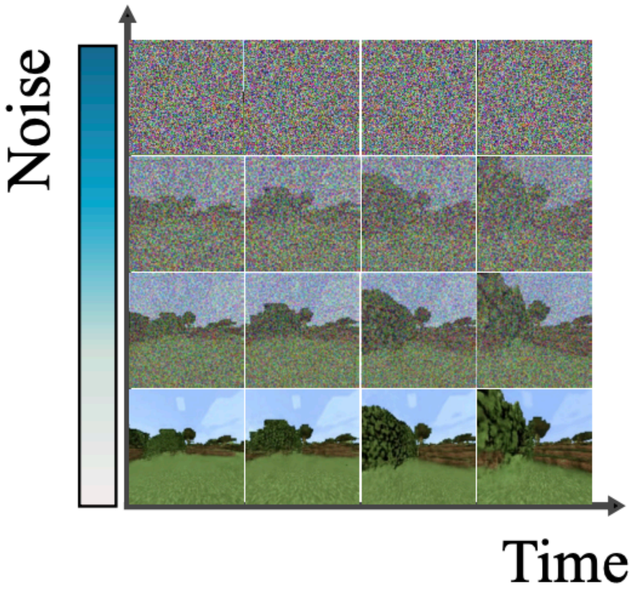
Observation

Latent State

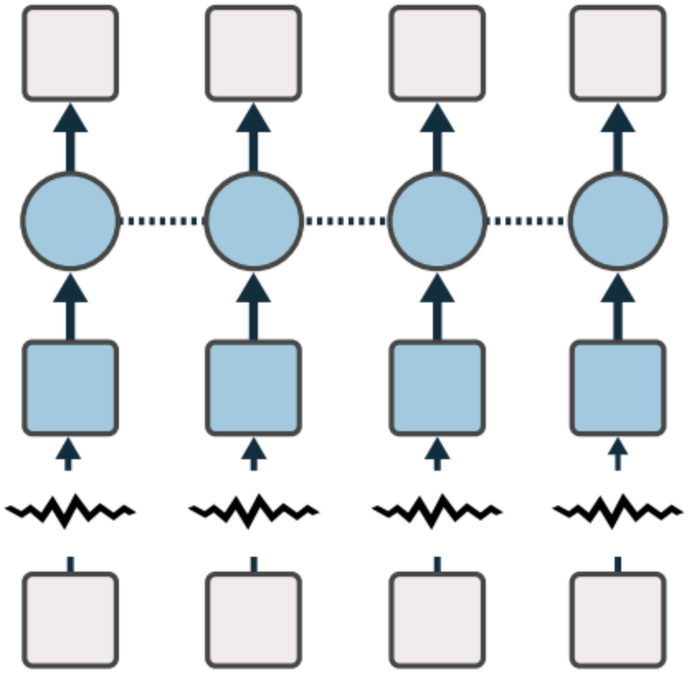
Generation

Add Noise

Full-Sequence Diffusion



Training



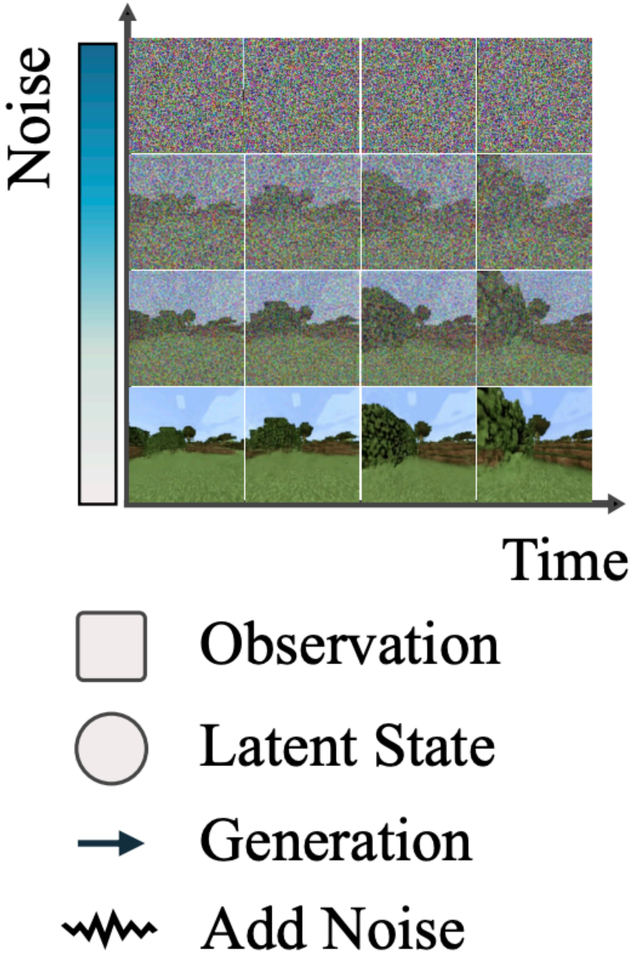
Observation

Latent State

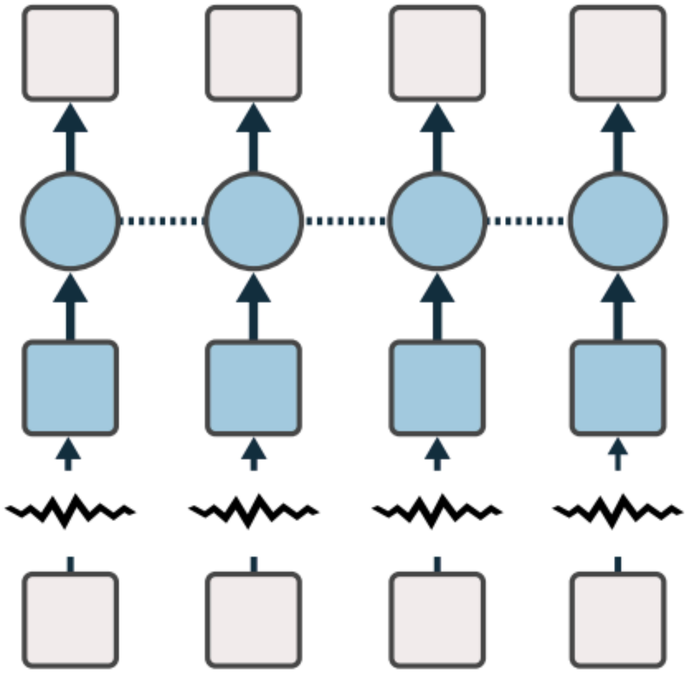
Generation

Add Noise

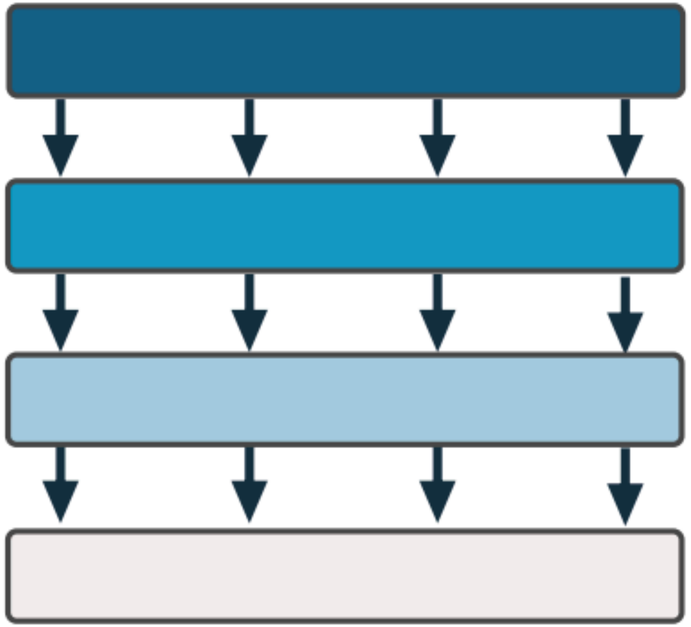
Full-Sequence Diffusion



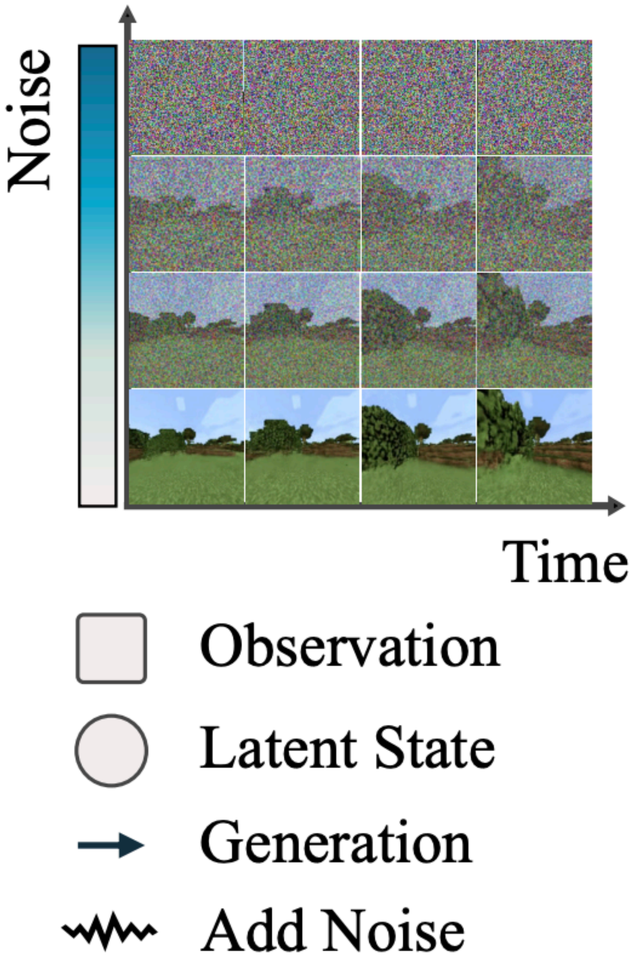
Training



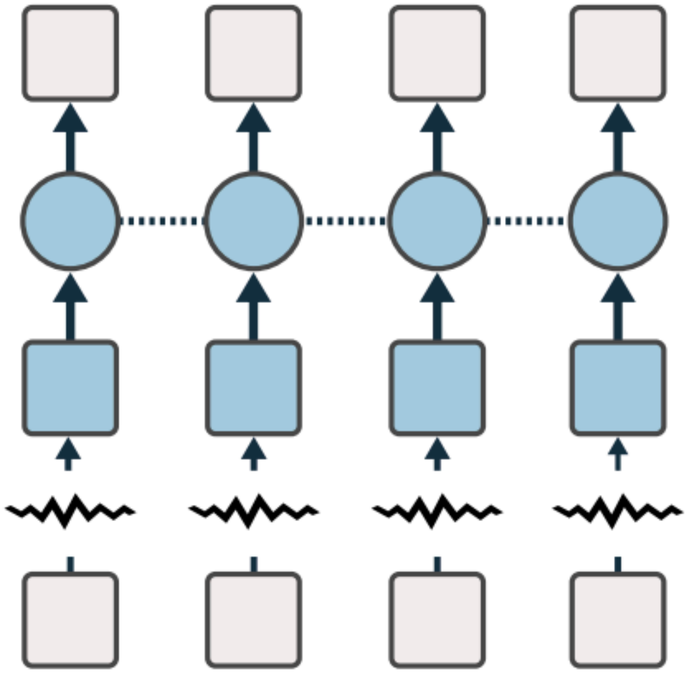
Inference



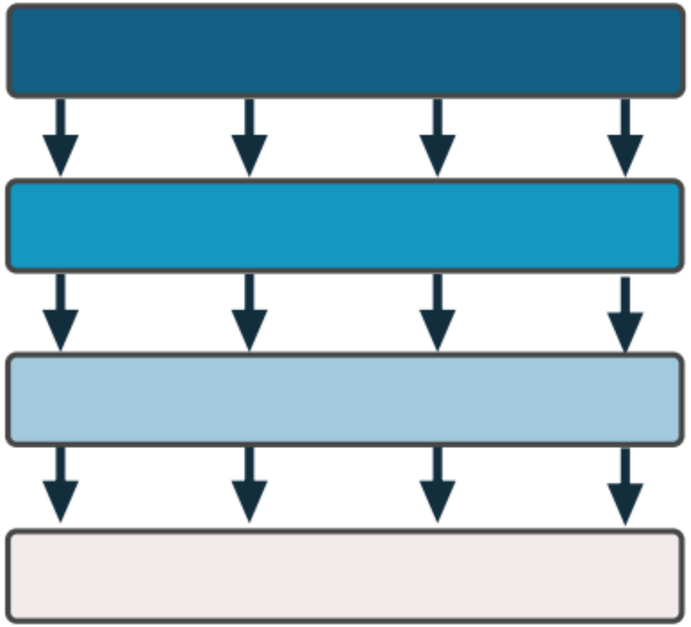
Full-Sequence Diffusion



Training



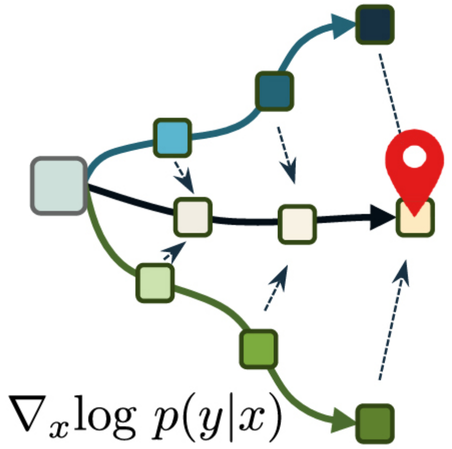
Inference



Examples: SORA, Meta Movie Gen, Diffusion Probabilistic Modeling for Video Generation, [Yang et al. 2022], Video Diffusion Models [Ho et al. 2022]

Capabilities of Existing Models

Guidance



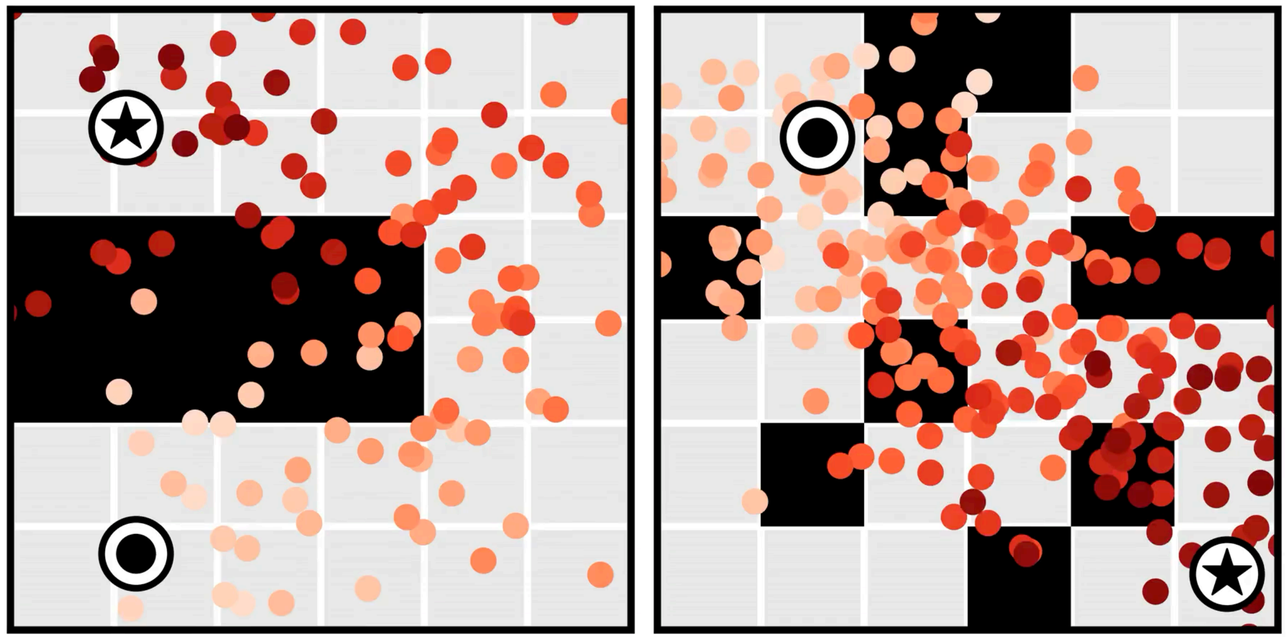
Teacher Forcing

Full-Seq. Diffusion

Planning: Sampling to Arrive at Particular Future

Full-Sequence Diffusion

Planning via **Guidance**



Compute loss on future state,
backdrop to past states
bias denoising

Trained on State, Action pairs

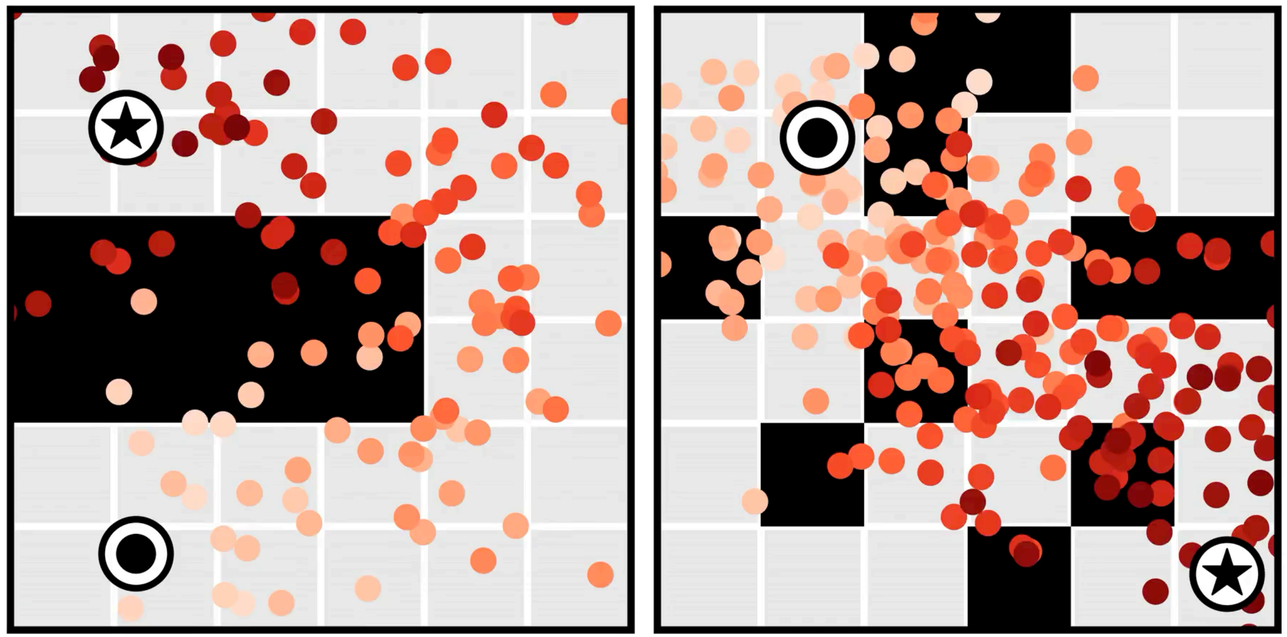
$$(S_0, a_0), \ldots, (S_n, a_n)$$

Planning with Diffusion for
Flexible Behavior Synthesis,
Tanner and Du et al., 2024

Planning: Sampling to Arrive at Particular Future

Full-Sequence Diffusion

Planning via **Guidance**



Compute loss on future state,
backdrop to past states
bias denoising

Trained on State, Action pairs

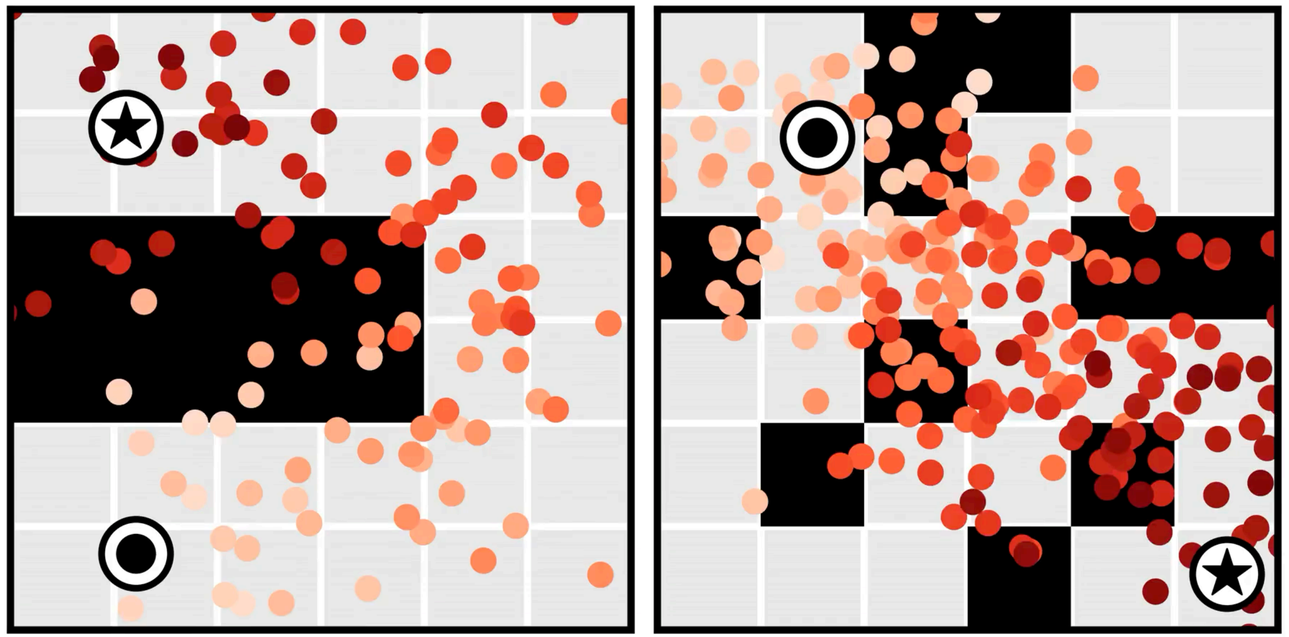
$$(S_0, a_0), \ldots, (S_n, a_n)$$

Planning with Diffusion for
Flexible Behavior Synthesis,
Tanner and Du et al., 2024

Planning: Sampling to Arrive at Particular Future

Full-Sequence Diffusion

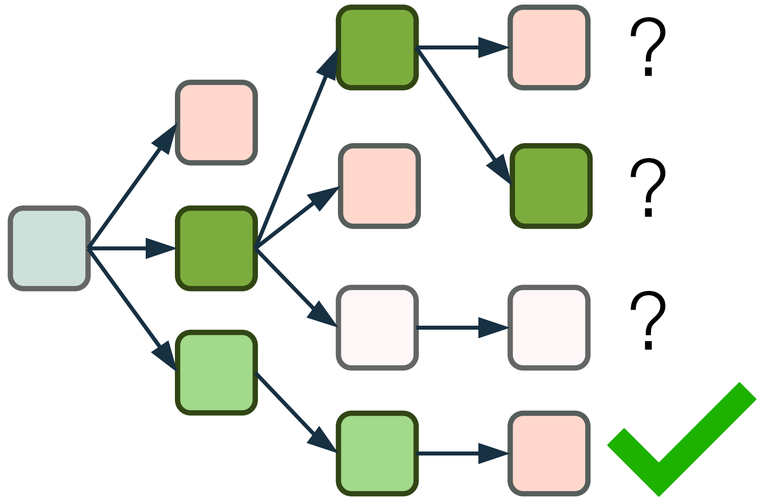
Planning via **Guidance**



Compute loss on future state,
backdrop to past states
bias denoising

Next-Token Prediction

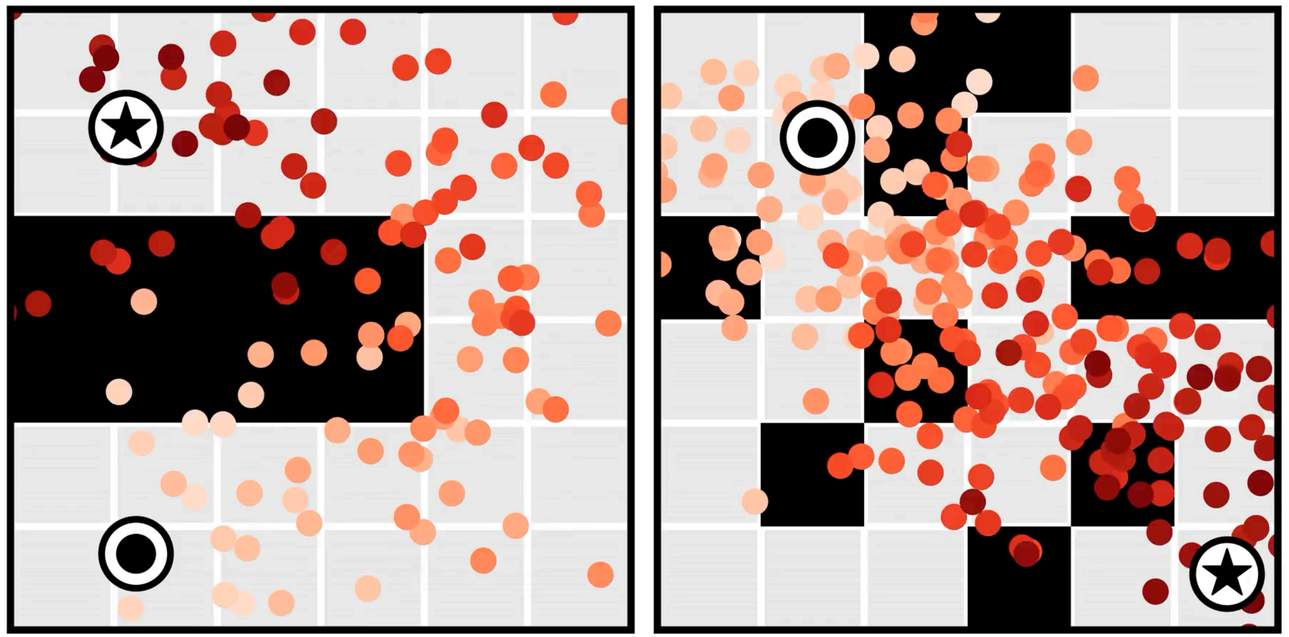
Planning via **Rollout & Search**



Planning: Sampling to Arrive at Particular Future

Full-Sequence Diffusion

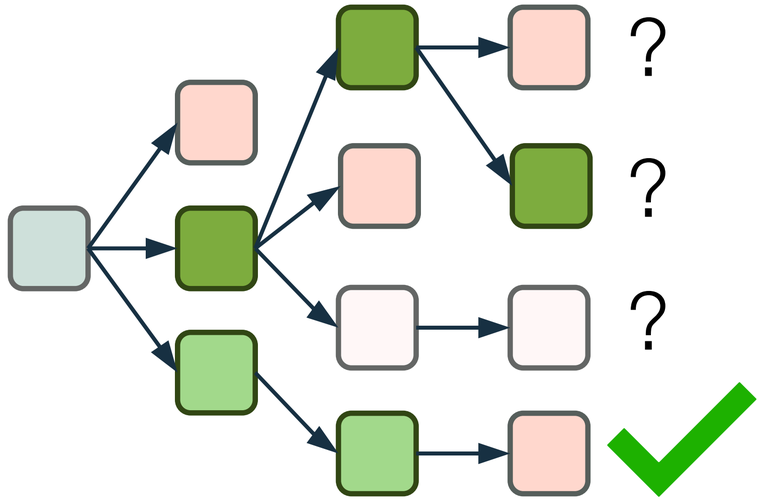
Planning via **Guidance**



Compute loss on future state,
backdrop to past states
bias denoising

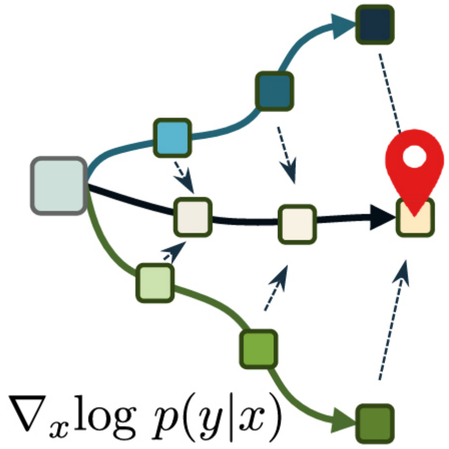
Next-Token Prediction

Planning via **Rollout & Search**

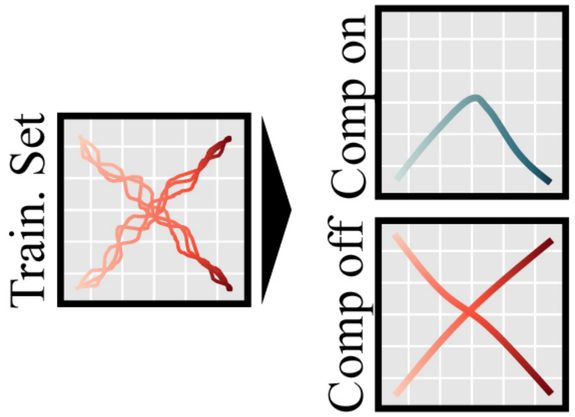


Capabilities of Existing Models

Guidance



Compositionality

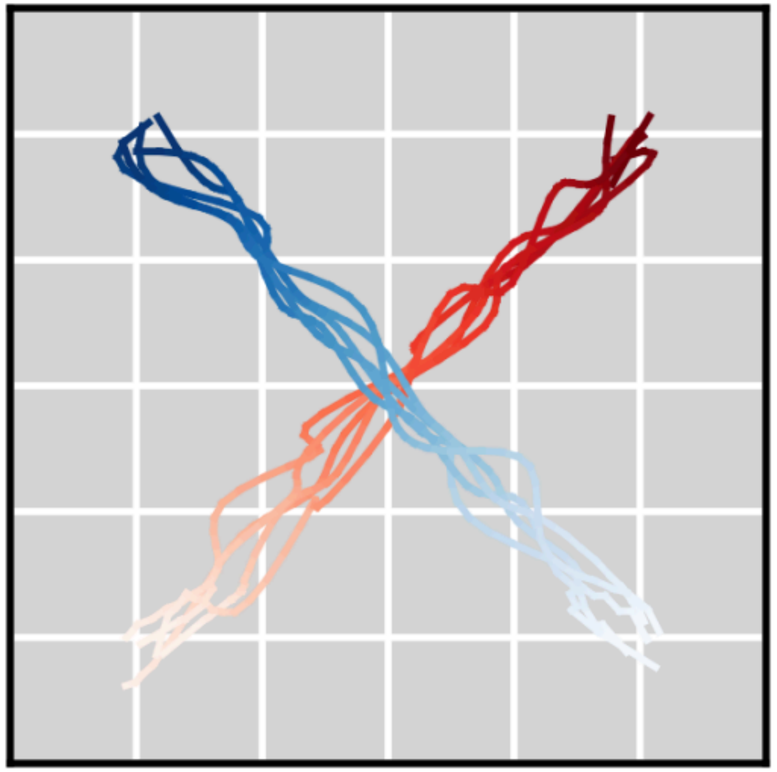


Teacher Forcing ✗ ✓

Full-Seq. Diffusion ✓ ✗

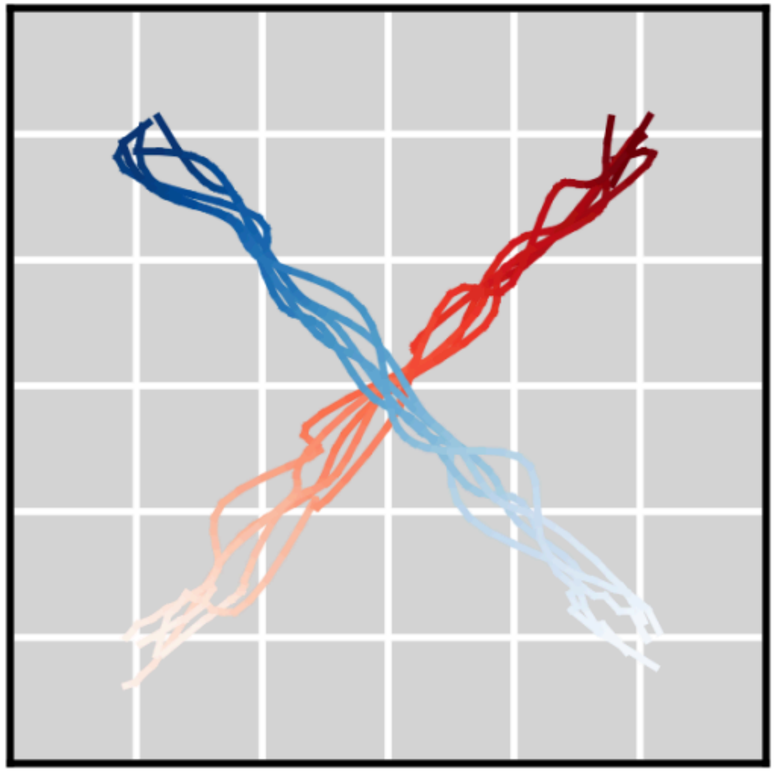
Compositionality and Conditioning on the Past

Training Set

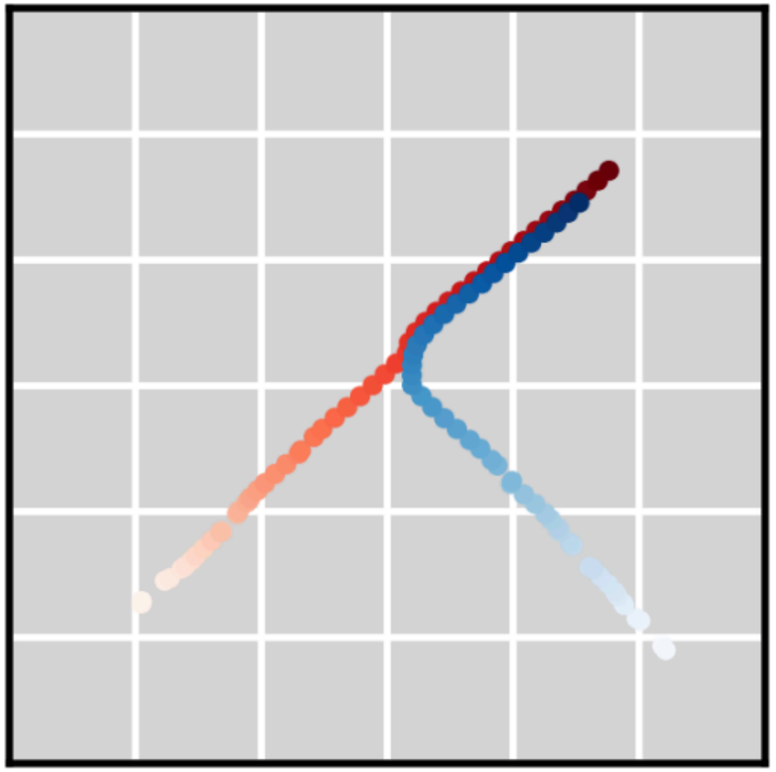


Compositionality and Conditioning on the Past

Training Set

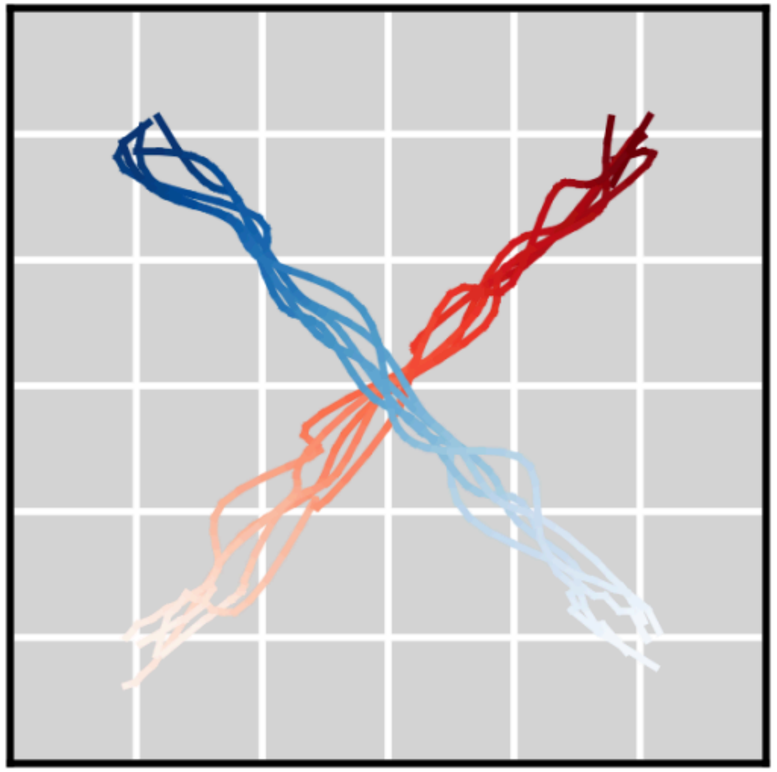


What we often want

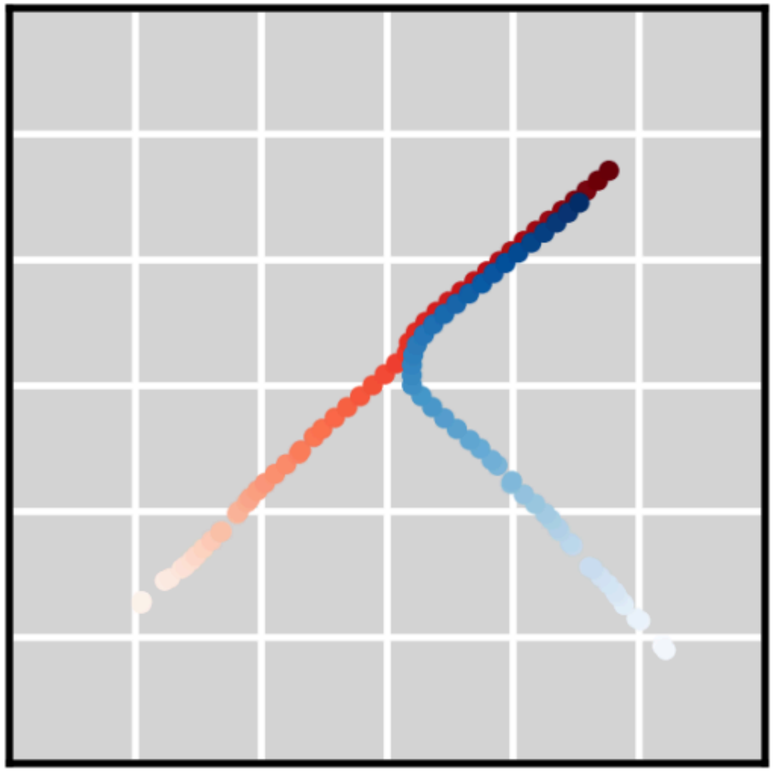


Compositionality and Conditioning on the Past

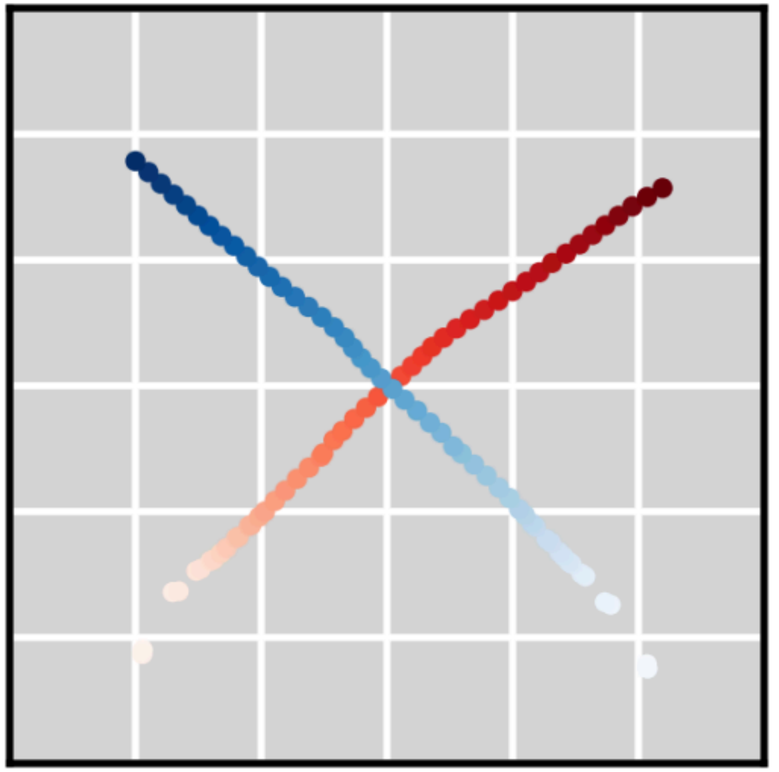
Training Set



What we often want

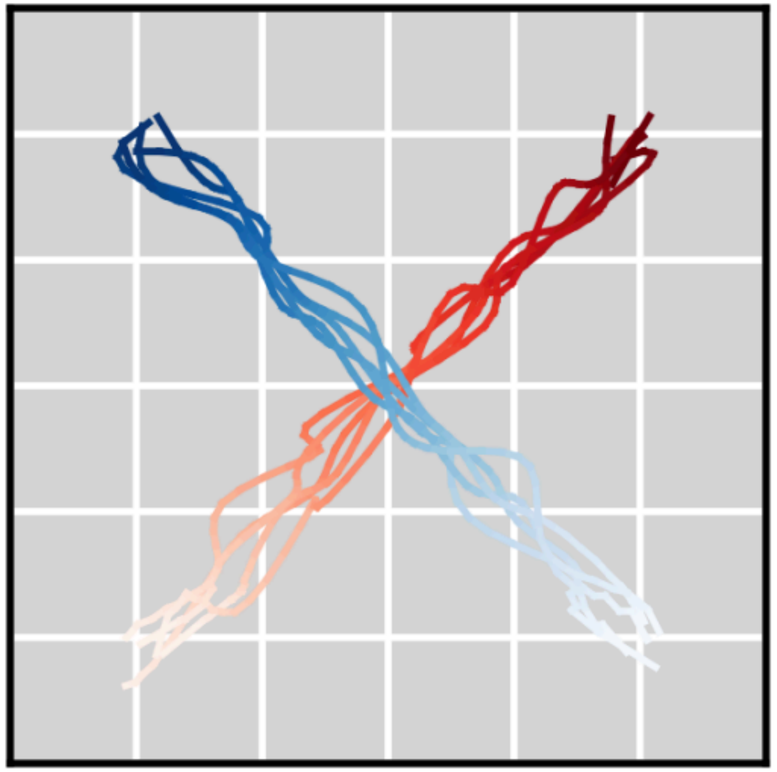


Full-Seq Diffusion

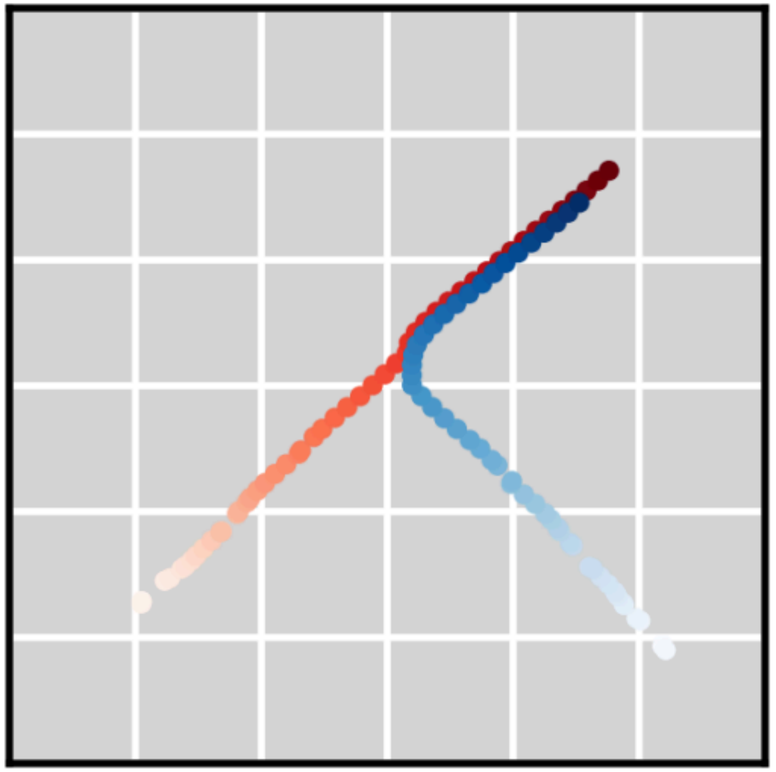


Compositionality and Conditioning on the Past

Training Set

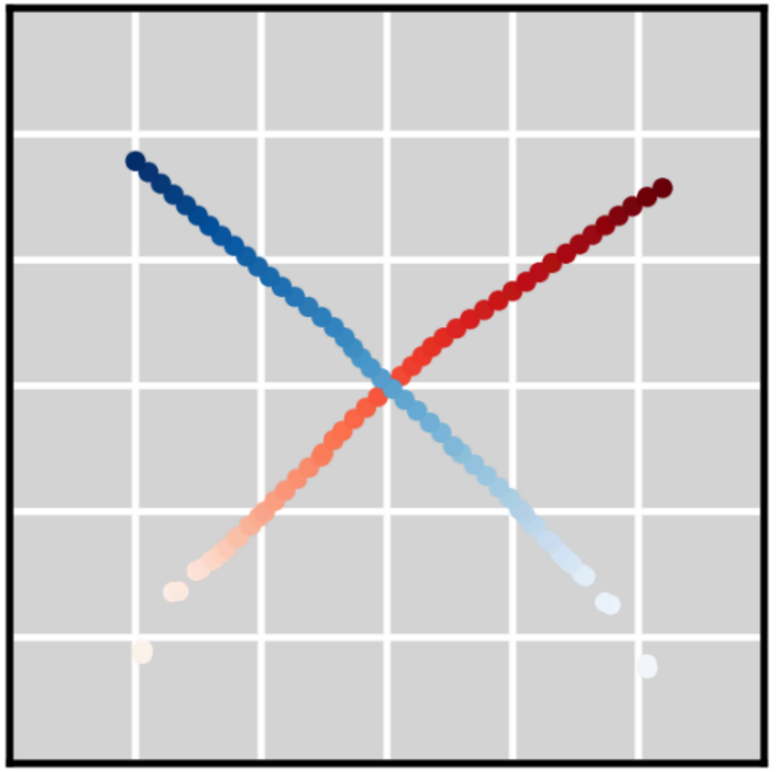


What we often want



$p(x_2 \mid x_1)p(x_1 \mid x_0)p(x_0)$

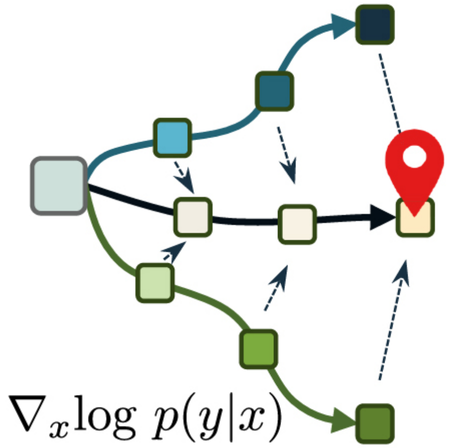
Full-Seq Diffusion



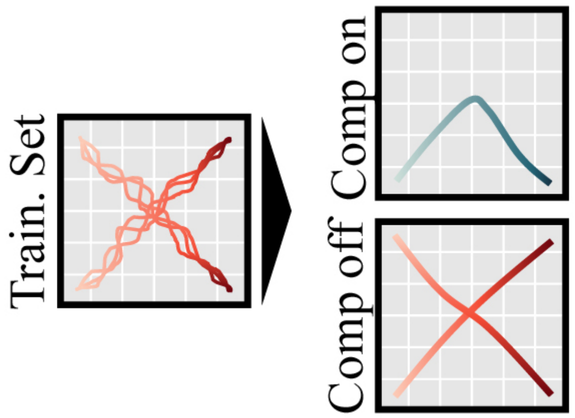
$p(x_0, \ldots, x_N)$

Capabilities of Existing Models

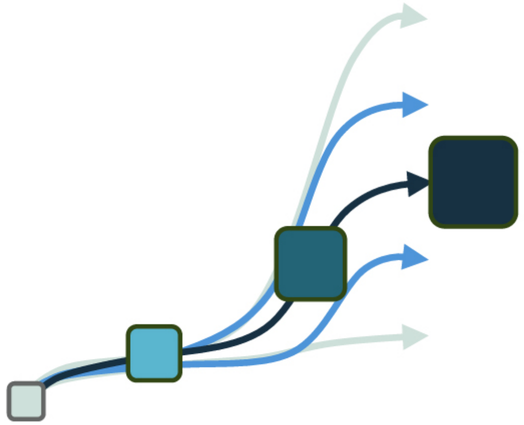
Guidance



Compositionality

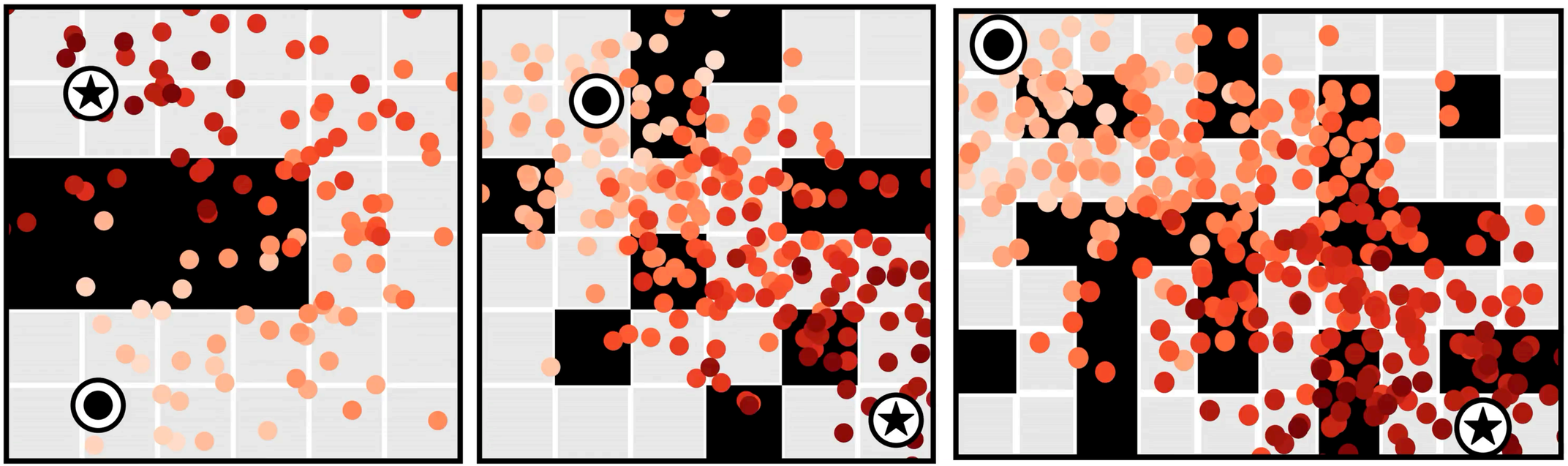


Causal Uncertainty



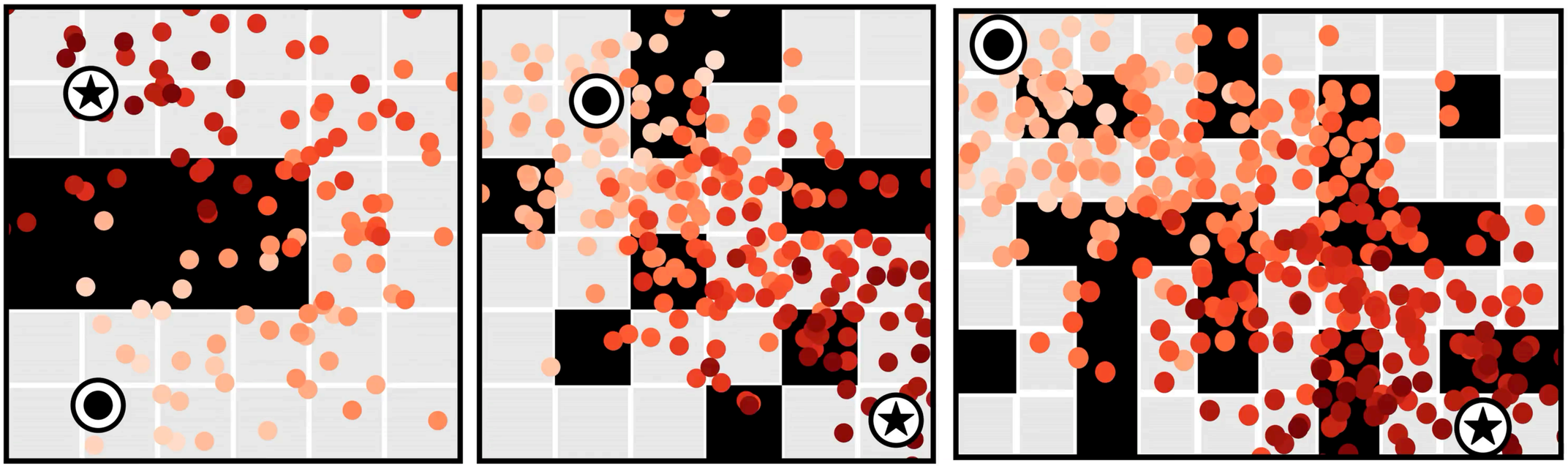
Teacher Forcing	✗	✓	✓
Full-Seq. Diffusion	✓	✗	✗

Causal Uncertainty



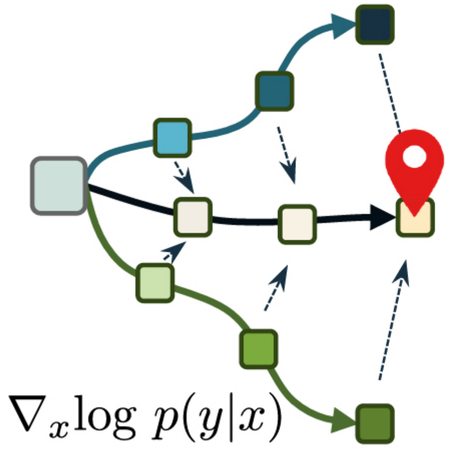
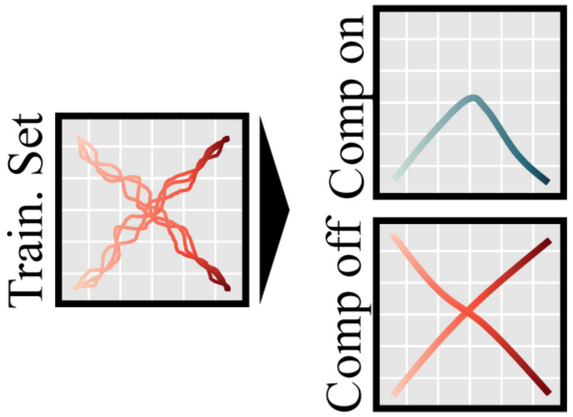
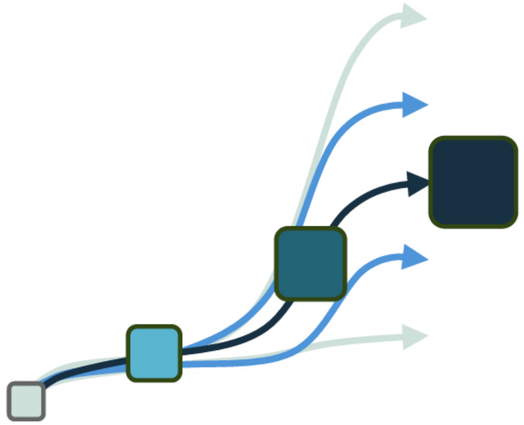
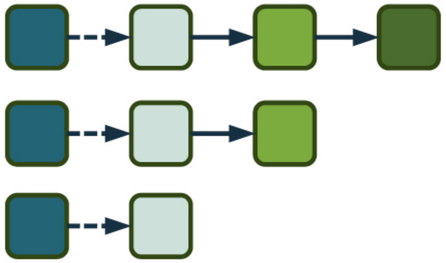
Full-Seq. Diffusion: uncertainty is removed from all states at the same speed. But a small difference in early states might lead to a *large* difference later! Doesn't work for planning…

Causal Uncertainty

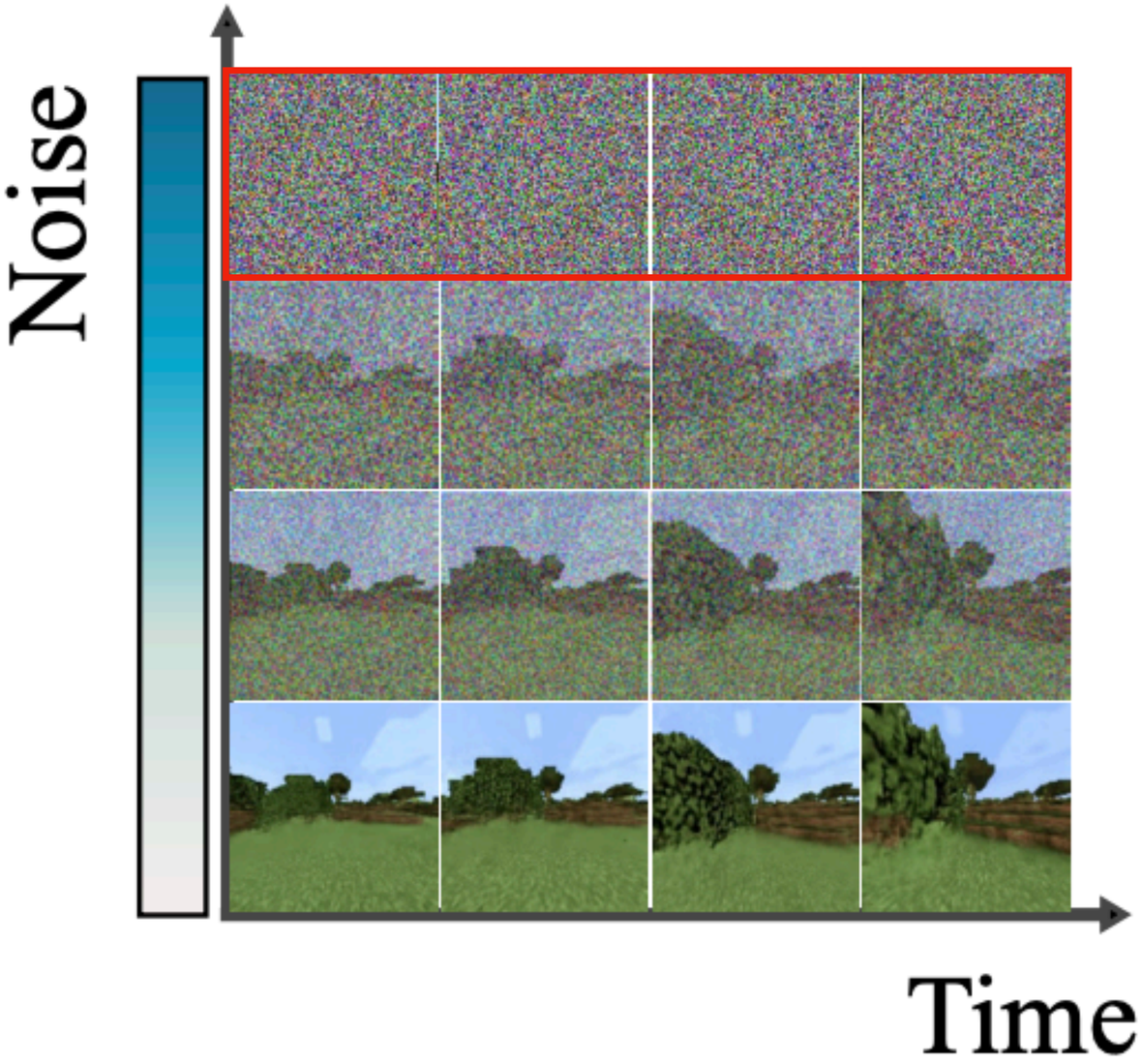


Full-Seq. Diffusion: uncertainty is removed from all states at the same speed. But a small difference in early states might lead to a *large* difference later! Doesn't work for planning…

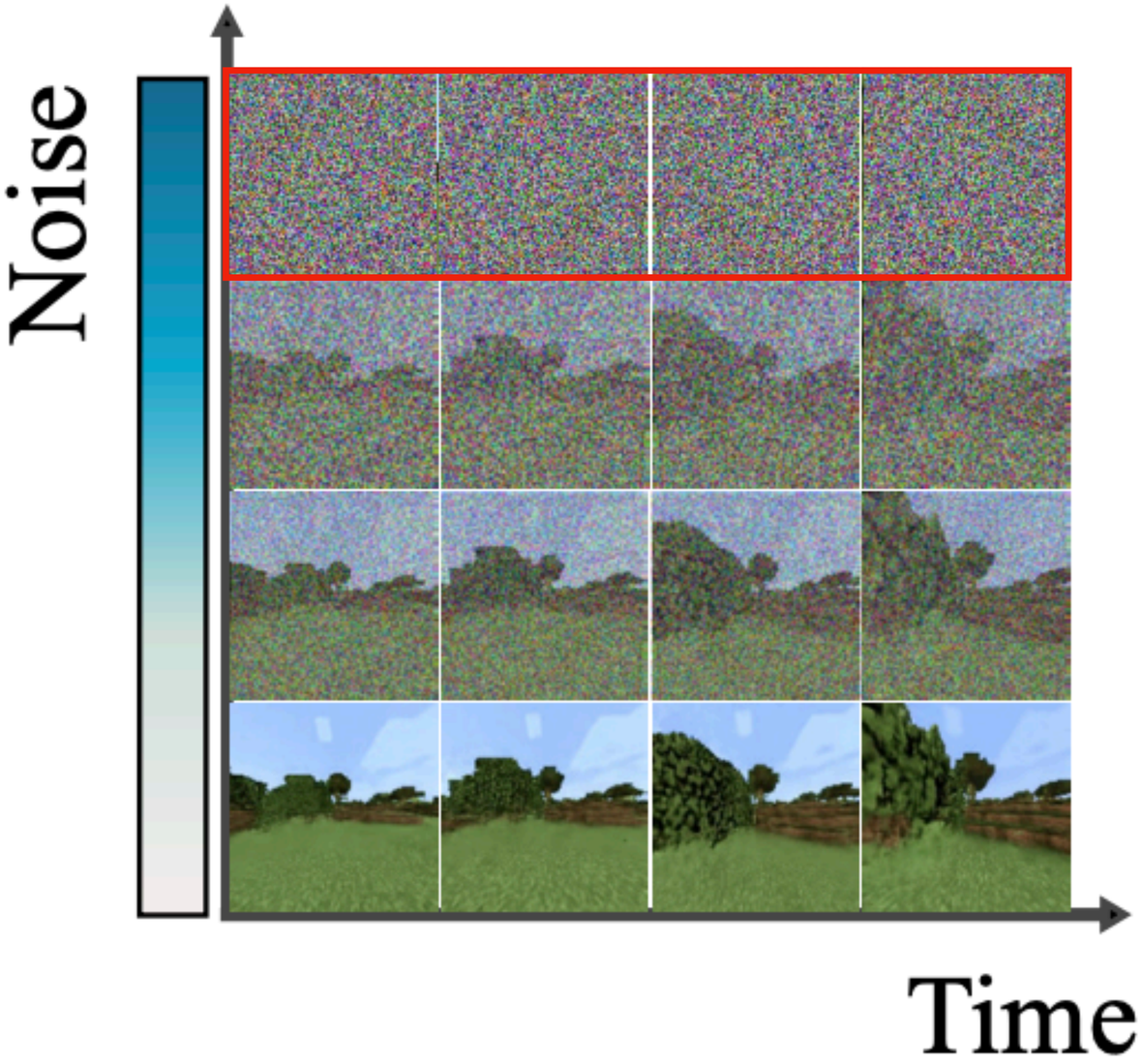
Capabilities of Existing Models

	Guidance	Compositionality	Causal Uncertainty	Flexible Horizon
				
Teacher Forcing	✗	✓	✓	✓
Full-Seq. Diffusion	✓	✗	✗	✗

Full-Seq. Diffusion: All Frames at Equal Noise Level



Full-Seq. Diffusion: All Frames at Equal Noise Level



Full-Sequence Diffusion