Embodied Vision

Generative Modeling

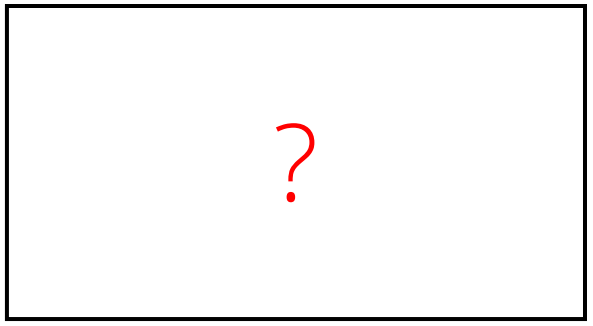
Tsung-Wei Ke

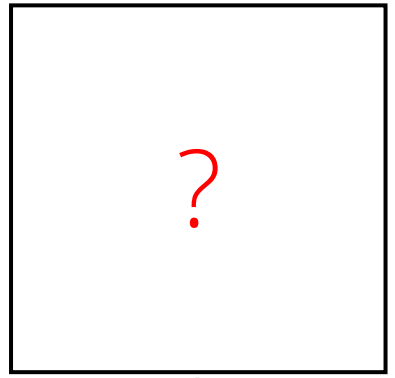
Spring 2026

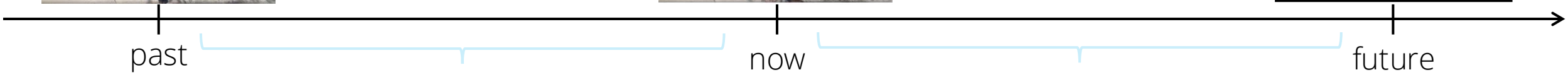
Content

- Generative Modeling
  - Autoregressive Models
  - Variational AutoEncoders
  - Denoising Diffusion Probabilistic Models
  - Flow Matching Models
- 4D Generative Modeling
  - 3D Geometry Generation
  - 3D Generation with Multi-view Video Generation
  - 4D Geometry Generation
  - 4D Generation with Multi-view Video Generation

We Inferred the Past Motion of an Observed Object… How About the Future Motion of the Object?







Stereo4D Learning How Things Move in 3D from Internet Stereo Videos . Jin et al.

Unfortunately, the Future is Not Unique. We May Have Multiple Plausible Futures

past

now

future

Nosedive

Faceplant

Prat fall

Squat

Can We Use Generative Models to Predict Futures?

Can We Use Generative Models to Predict Futures?

- Previously, we obtained 3D from 2D images; We obtained 4D from 2D videos.

- Can we generate 3D/4D world without 2D data?

Can We Use Generative Models to Generate a 3D World?

- Previously, we obtained 3D from 2D images.

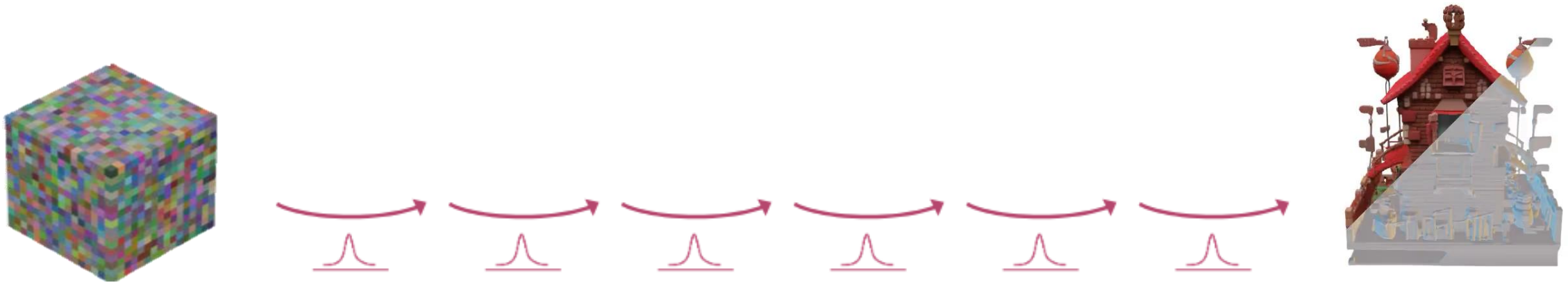
- How to generate a 3D world?
  - Generate 3d assets directly→~~The 3D generative is not competent yet, as the amount of available 3D data is still small~~
  - Generate multi-view 2D images and lift to 3D →Yes! 2D generation is already great, since we have a huge amount of 2D images available on the Internet.

Content

- Generative Modeling
  - Autoregressive Models
  - Variational AutoEncoders
  - Denoising Diffusion Probabilistic Models
  - Flow Matching Models
- 4D Generative Modeling
  - 3D Geometry Generation
  - 3D Generation with Multi-view Video Generation
  - 4D Geometry Generation
  - 4D Generation with Multi-view Video Generation

Generate 3D Assets with Diffusion Models

- Idea: Generate 3D assets with diffusion models
- The availability and scalability of training data?
- The choice of 3D representations
  - ➢ How to model shape and visual appearance?
  - ➢ Do we need a joint representations?



The availability and scalability of training data

Source	# Objects
IKEA [32]	219
GSO [17]	1K
EGAD [41]	2K
OmniObject3D [63]	6K
PhotoShape [46]	5K
ABO [13]	8K
Thingi10K [67]	10K
3d-Future [19]	10K
ShapeNet [9]	51K
Objaverse 1.0 [14]	800K
Objaverse-XL	**10.2M**

Representations

- Representations of shapes
  - Voxel grids
  - Point clouds
  - Meshes
  - 3D GS
  - …
- Representations of visual appearance
  - Texture maps
  - RGB
  - Spherical harmonics
  - …
- What are better design choices?

Structured 3D Latents

for Scalable and Versatile 3D Generation

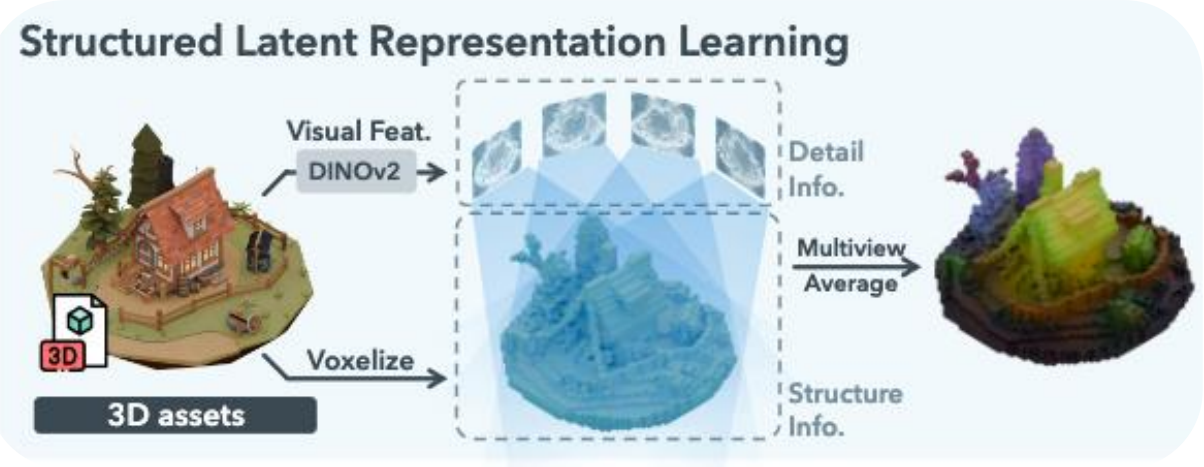
Jianfeng Xiang[1,3] Zelong Lv[2,3] Sicheng Xu[3] Yu Deng[3] Ruicheng Wang[2,3]
Bowen Zhang[2,3] Dong Chen[3] Xin Tong[3] Jiaolong Yang[3]

[1]Tsinghua University [2]USTC [3]Microsoft Research

CVPR 2025 Highlight

Structured Latents

- A unified 3D latent representations for high-quality 3D generation via **active voxels**
  - Represent shape by active voxel positions $\{\mathbf{p}_i\}_{i=1}^{L}$, which are sparse voxels intersecting the object's surface
  - Represent appearance by image features $\{\mathbf{f}_i\}_{i=1}^{L}$, which extracted from densely rendered views and attached onto active voxels



Issue: Image Features are Overly High Dimensional

- DINOv2 is used to extract image features, which has 2048 dimensions. Generating high-dimensional data is computationally expensive
- Idea: compress image features $\{\mathbf{f}_i\}_{i=1}^{L}$ to low-dimensional latents $\{\mathbf{z}_i\}_{i=1}^{L}$ with VAE by reconstructing (1) 3D GS, (2) Meshes and (3) Radiance fields

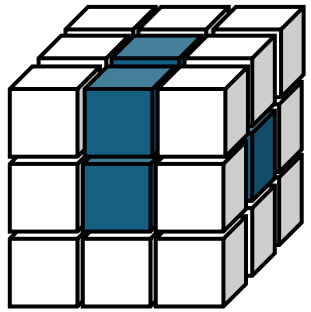
Generate Unified 3D Latent Representations

- A unified 3D latent representation $\{(\mathbf{p_i}, \mathbf{z_i})\}_{\mathbf{i=1}}^{\mathbf{L}}$ for high-quality 3D generation via **active voxels**, with active voxel positions $\{\mathbf{p_i}\}_{\mathbf{i=1}}^{\mathbf{L}}$ and visual latents $\{\mathbf{z_i}\}_{\mathbf{i=1}}^{\mathbf{L}}$
- A two-stage generation pipeline: first generate sparse structures $\{\mathbf{p_i}\}_{\mathbf{i=1}}^{\mathbf{L}}$ then generate visual representations $\{\mathbf{z_i}\}_{\mathbf{i=1}}^{\mathbf{L}}$
- Consider flow matching as the generative model

How to Generate Sparse Structures $\{p_i\}_{i=1}^{L}$

- Challenge 1: Active voxel positions $\{p_i\}_{i=1}^{L}$ are "set" representations, while generating such unordered data with diffusion models is challenging
- Solution 1: Reorganize active voxels as binary voxel grids

(a, b, c)
$\vdots$
(d, e, f)

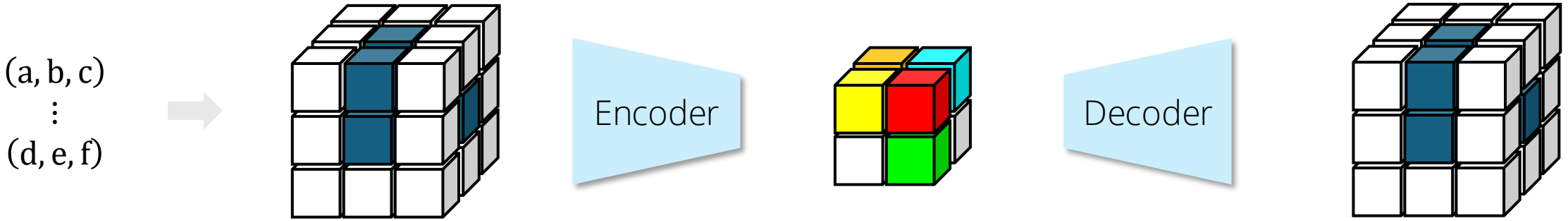


Sparse active voxel positions

Dense binary voxel grids

How to Generate Sparse Structures $\{p_i\}_{i=1}^{L}$

- Challenge 1: Active voxel positions $\{p_i\}_{i=1}^{L}$ are "set" representations, while generating such unordered data with diffusion models is challenging
- Solution 1: Reorganize active voxels as binary voxel grids
- Challenge 2: Voxel grids are costly to generate
- Solution 2: Compress voxel grids into low-dim voxel latents with VAE and learn diffusion models to generate these latents



Sparse active voxel positions

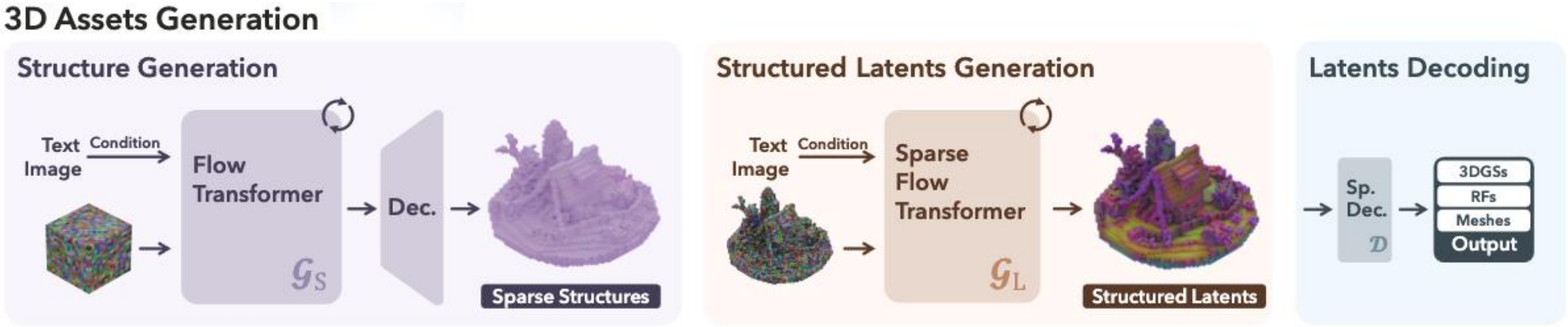
Dense binary voxel grids

Dense voxel latents

How to Generate Visual Representations $\{\mathbf{z_i}\}_{\mathbf{i=1}}^{\mathbf{L}}$

- Challenge 1: Visual representations $\{\mathbf{z_i}\}_{\mathbf{i=1}}^{\mathbf{L}}$ are "set" representations, while generating such unordered data with diffusion models is challenging
- Solution 1: Reorganize active voxels as $\mathbf{N \times N \times N \times d}$ voxel grids ($\mathbf{d}$ denotes the dimension of $\mathbf{z_i}$)
- Challenge 2: The $\mathbf{N \times N \times N \times d}$ voxel grids is overly high-dimensional
- Solution 2: Compress the $\mathbf{N \times N \times N \times d}$ voxel grids to lower-dimensional voxel grids with sparse 3D convolution

Generate Unified 3D Latent Representations



Results: Text to 3D Asset

A rustic log cabin with a stone chimney and a wooden porch.

Portable transistor radio, dark cover, speaker grille, brand logo on front.

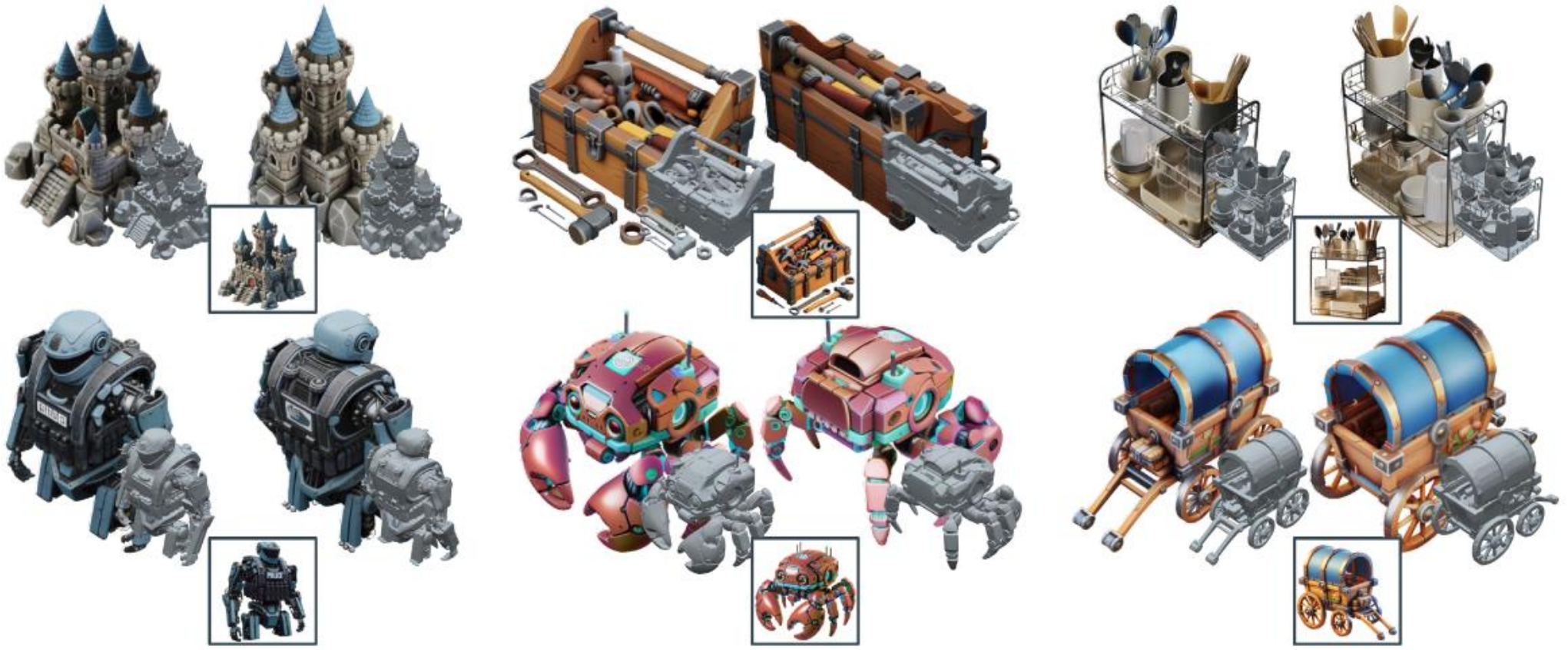
Futuristic red toy blaster with transparent magazine.

Blocky, orange and teal robot with articulated limbs.

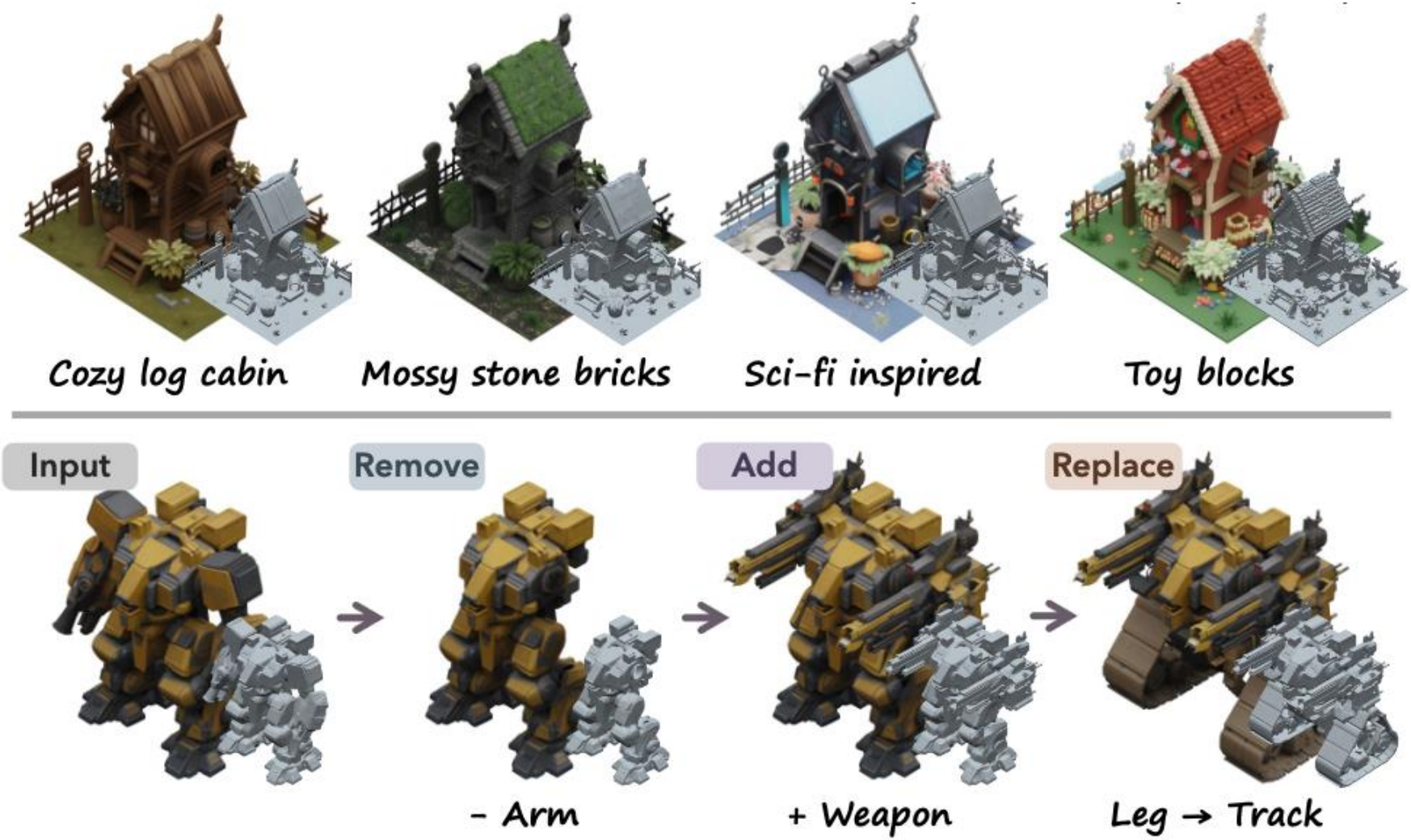
Metallic dog-like robot with articulated legs and futuristic design elements.

Yellow and black bulldozer with movable front blade.

Results: Image to 3D Asset



Results: 3D Asset Editing



Results: Composable 3D World

SAM 3D: 3Dfy Anything in Images

SAM 3D Team, **Xingyu Chen**[*], **Fu-Jen Chu**[*], **Pierre Gleize**[*], **Kevin J Liang**[*], **Alexander Sax**[*], **Hao Tang**[*], **Weiyao Wang**[*], **Michelle Guo**, **Thibaut Hardin**, **Xiang Li**[°], **Aohan Lin**, **Jiawei Liu**, **Ziqi Ma**[°], **Anushka Sagar**, **Bowen Song**[°], **Xiaodong Wang**, **Jianing Yang**[°], **Bowen Zhang**[°], **Piotr Dollár**[†], **Georgia Gkioxari**[†], **Matt Feiszli**[†§], **Jitendra Malik**[†§]

Meta Superintelligence Labs

[*]Core Contributor (Alphabetical, Equal Contribution), [°]Intern, [†]Project Lead, [§]Equal Contribution

We present SAM 3D, a generative model for visually grounded 3D object reconstruction, predicting geometry, texture, and layout from a single image. SAM 3D excels in natural images, where occlusion and scene clutter are common and visual recognition cues from context play a larger role. We achieve this with a human- and model-in-the-loop pipeline for annotating object shape, texture, and pose, providing visually grounded 3D reconstruction data at unprecedented scale. We learn from this data in a modern, multi-stage training framework that combines synthetic pretraining with real-world alignment, breaking the 3D "data barrier". We obtain significant gains over recent work, with at least a 5 : 1 win rate in human preference tests on real-world objects and scenes. We will release our code and model weights, an online demo, and a new challenging benchmark for in-the-wild 3D object reconstruction.

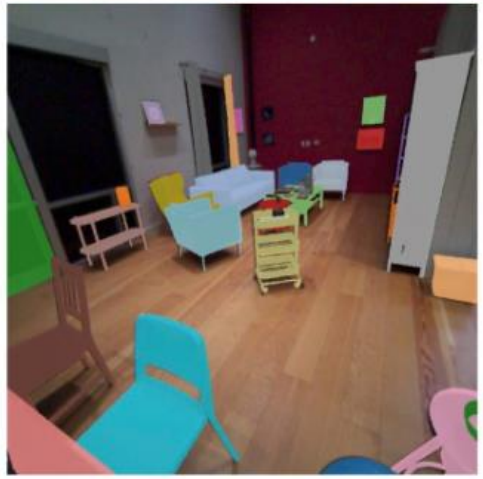
Demo: https://www.aidemos.meta.com/segment-anything/editor/convert-image-to-3d
Code: https://github.com/facebookresearch/sam-3d-objects
Website: https://ai.meta.com/sam3d

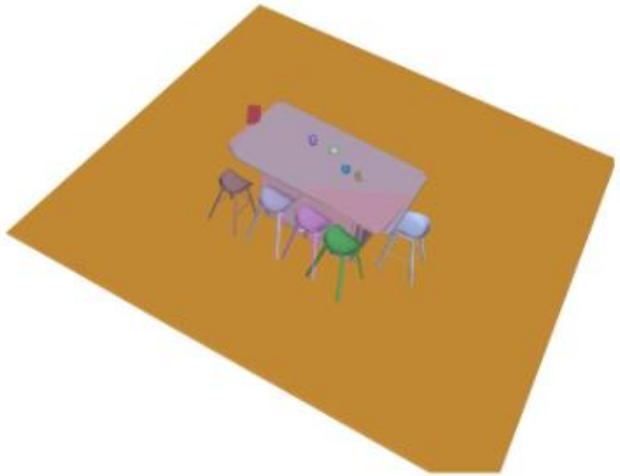
Meta

Results: Reconstruct Digital Twins from RGB Images





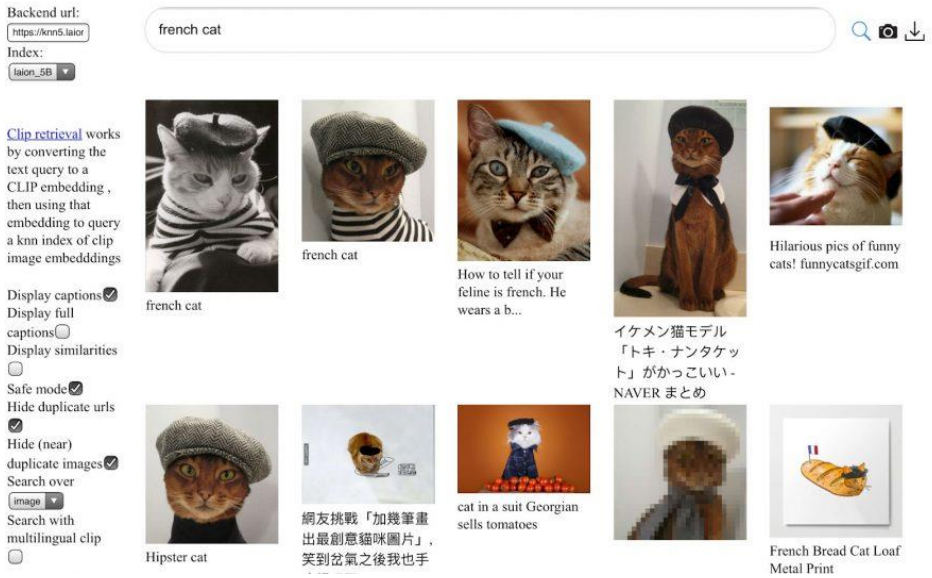


However, 3D Asset Generation Has Scale Issues

Is it possible to generate 3D from image generation?
Will it be better than 3D asset generation?

10M+ object meshes

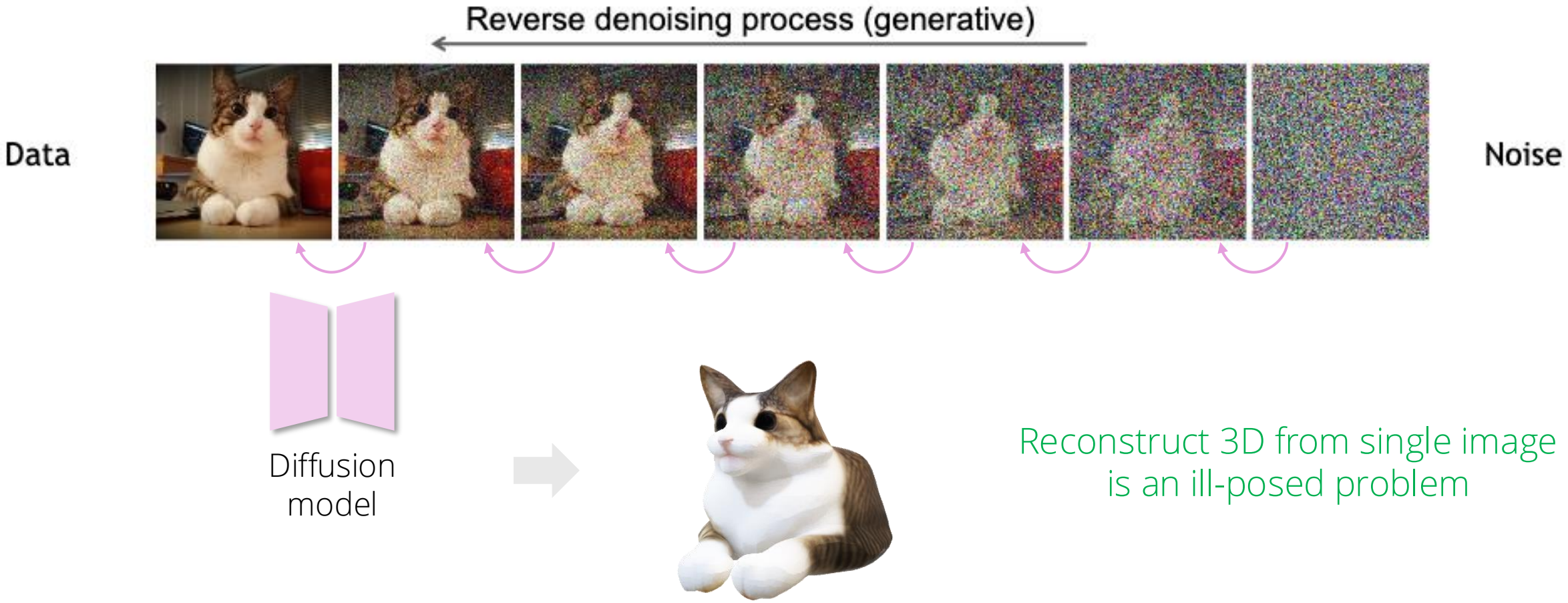


5B+ text-image pairs

Content

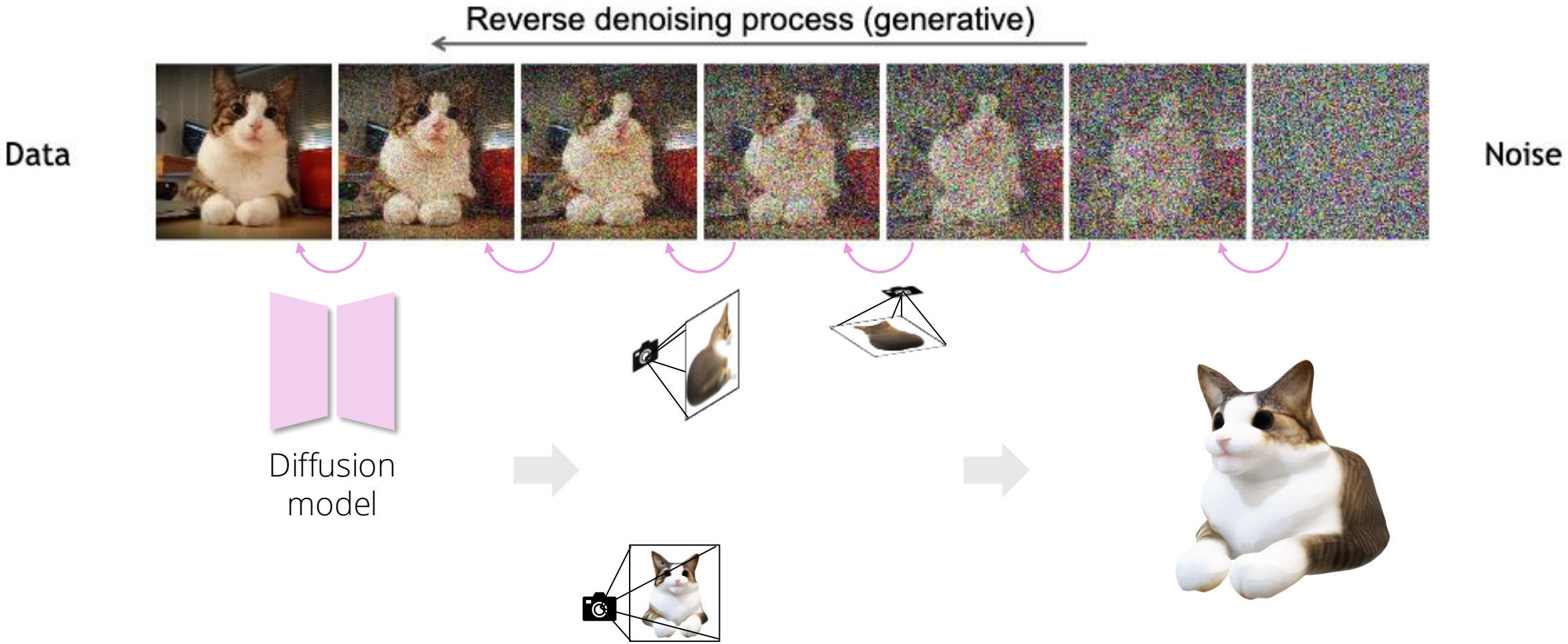
- Generative Modeling
  - Autoregressive Models
  - Variational AutoEncoders
  - Denoising Diffusion Probabilistic Models
  - Flow Matching Models
- 4D Generative Modeling
  - 3D Geometry Generation
  - 3D Generation with Multi-view Video Generation
  - 4D Geometry Generation
  - 4D Generation with Multi-view Video Generation

Can We Obtain 3D Object from 2D Image Generation?



Slide credit: https://cvpr2023-tutorial-diffusion-models.github.io/

Can We Generate Multiple Views of the Input Image from Image Diffusion Model?



Slide credit: https://cvpr2023-tutorial-diffusion-models.github.io/

Let's Revisit The Learning Objective of Diffusion Models

- $p_\theta(x_0) \coloneqq \int p_\theta(x_{0:T}) dx_{0:T}$
- $p_\theta(x_{0:T}) \coloneqq p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)$

- Diffusion models minimize the variational bound (ELBO) on negative log likelihood

$$\mathbb{E}[-\log p_\theta(x_0)] \leq \mathbb{E}_q \left[-\log \frac{p_\theta(x_{0:T})}{q(x_{1:T}|x_0)} \right] = \mathbb{E}_q \left[-\log p(x_T) - \sum_{t \geq 1} \log \frac{p_\theta(x_{t-1}|x_t)}{q(x_t|x_{t-1})} \right]$$

$$= \mathbb{E}_q \left[\underbrace{D_{KL}(q(x_T|x_0) \| p_\theta(x_T))}_{L_T} + \sum_{t \geq 1} \underbrace{D_{KL}(q(x_{t-1}|x_t, x_0) \| p_\theta(x_{t-1}|x_t))}_{L_{t-1}} - \underbrace{\log p_\theta(x_0|x_1)}_{L_0} \right]$$

Denoising Diffusion Probabilistic Models. Ho et al. NeuRIPS 2020.

The Learning Objective of Diffusion Models

- $\tilde{\mu}_t(x_t, x_0) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t} x_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t$
- $x_t(x_0, \epsilon) = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$
- To maximize the likelihood $p_\theta(x_0)$, we need to minimize the negative log likelihood of intermediate diffusion samples

$$L_{t-1} = \mathbb{E}_q[D_{KL}(\ q(x_{t-1}|x_t, x_0) \| p_\theta(x_{t-1}|x_t)\ )] = \mathbb{E}_q\left[\frac{1}{2\sigma_t^2}\|\tilde{\mu}_t(x_t, x_0) - \mu_\theta(x_t, t)\|^2\right] + C$$

Do you know how to show this?
Check the Eq 8 and 9 of the paper

This results from how we parameterize our diffusion model

$$L_{t-1} \propto \mathbb{E}_{x_0,\epsilon}\left[\left\|\frac{1}{\sqrt{\alpha_t}}\left(x_t(x_0, \epsilon) - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon\right) - \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)\right)\right\|^2\right]$$

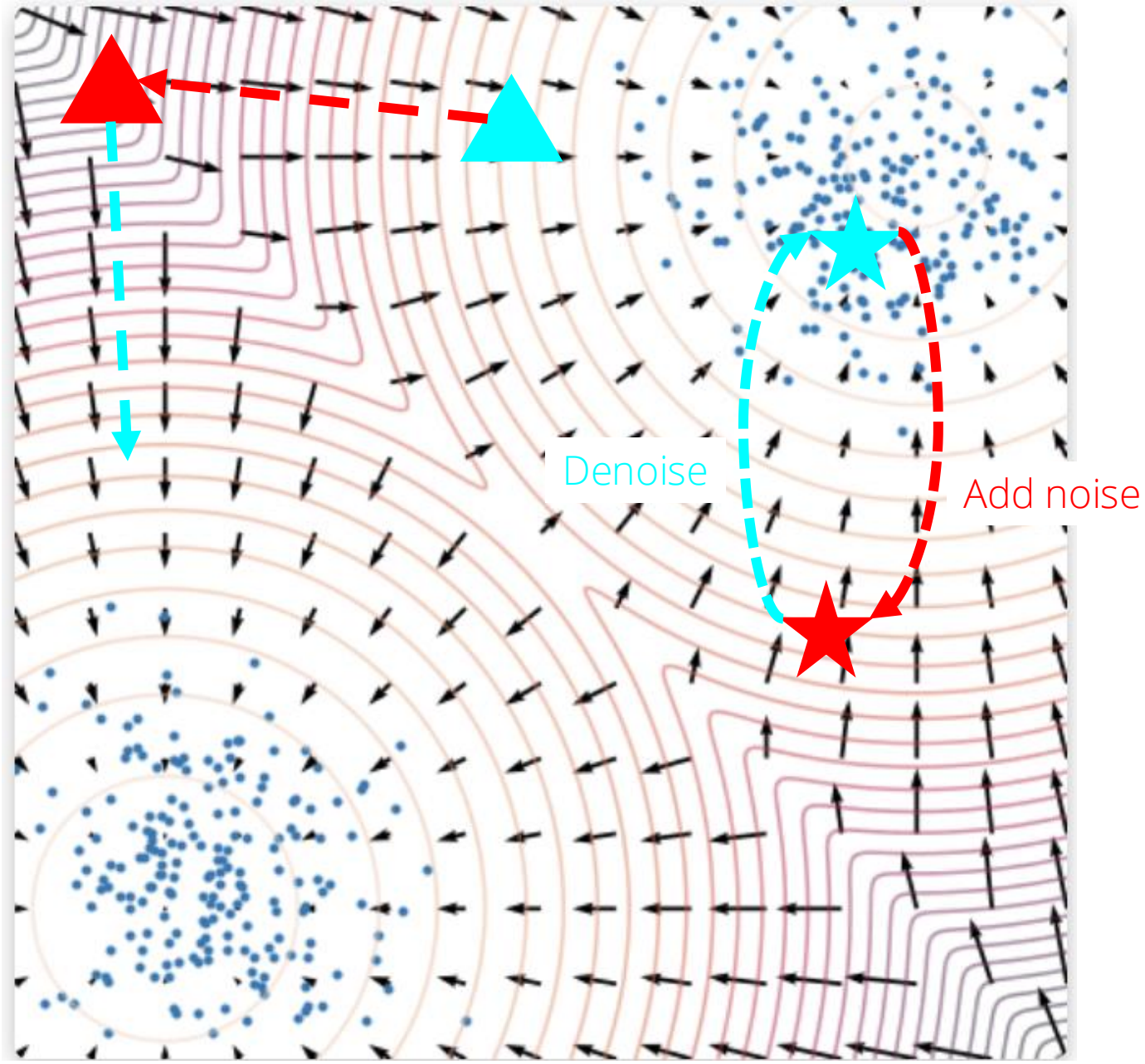
We Can Optimize the Generated Sample by Minimizing the Intermediate Diffusion Loss

- $x_t(x_0, \epsilon) = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$

- To maximize the likelihood $p_\theta(x_0)$, we need to minimize the negative log likelihood of intermediate diffusion samples

$$L_{t-1} \propto \mathbb{E}_{x_0,\epsilon}[\|\epsilon - \epsilon_\theta(x_t, t)\|^2] = \mathbb{E}_{x_0,\epsilon}\left[\left\|\epsilon - \epsilon_\theta\left(\sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon, t\right)\right\|^2\right]$$

- But here, instead of optimizing the diffusion model parameterized by θ, we aim to optimize x_0. In other words, if x_0 is sampled from the same distribution (of the training data), the diffusion loss would be small.

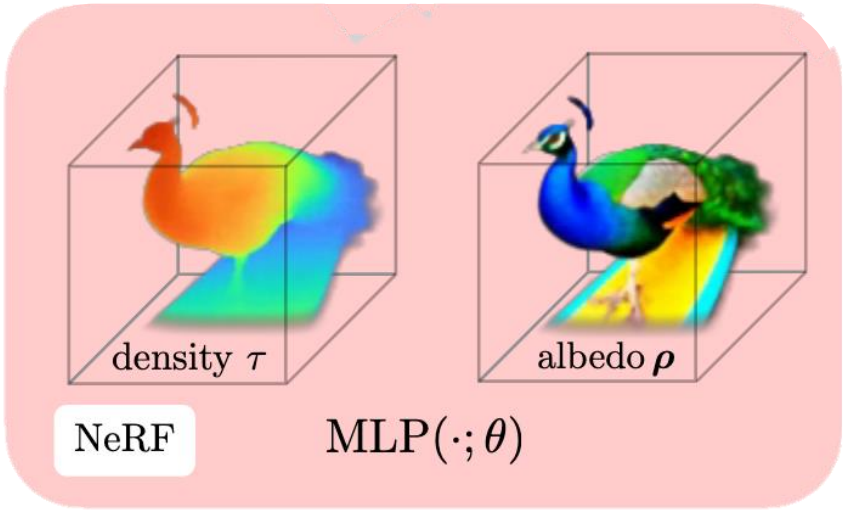


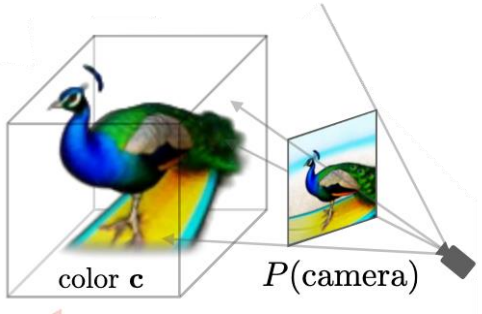
Observation: Noising good $\mathbf{x_0}$ is more easily to denoise back to clean samples than bad $\mathbf{x_0}$.

Idea: optimizing denoising loss to search good $\mathbf{x_0}$

Image credit https://yang-song.net/blog/2021/score/

3D-from-2D Reconstruction Recipe: Render a View from Differentiable 3D Model





- **Goal**: A 3D volume to be optimized, which can can be voxel grids, NERF, or 3DGS
- **Idea**: optimize the differentiable 3D volume with the prior knowledge in 2D generative models
- **Supervision signal**: rendering loss between rendered and generated images

DreamFusion: Text-to3D Using 2D Diffusion. Poole et al. ICLR 2022.

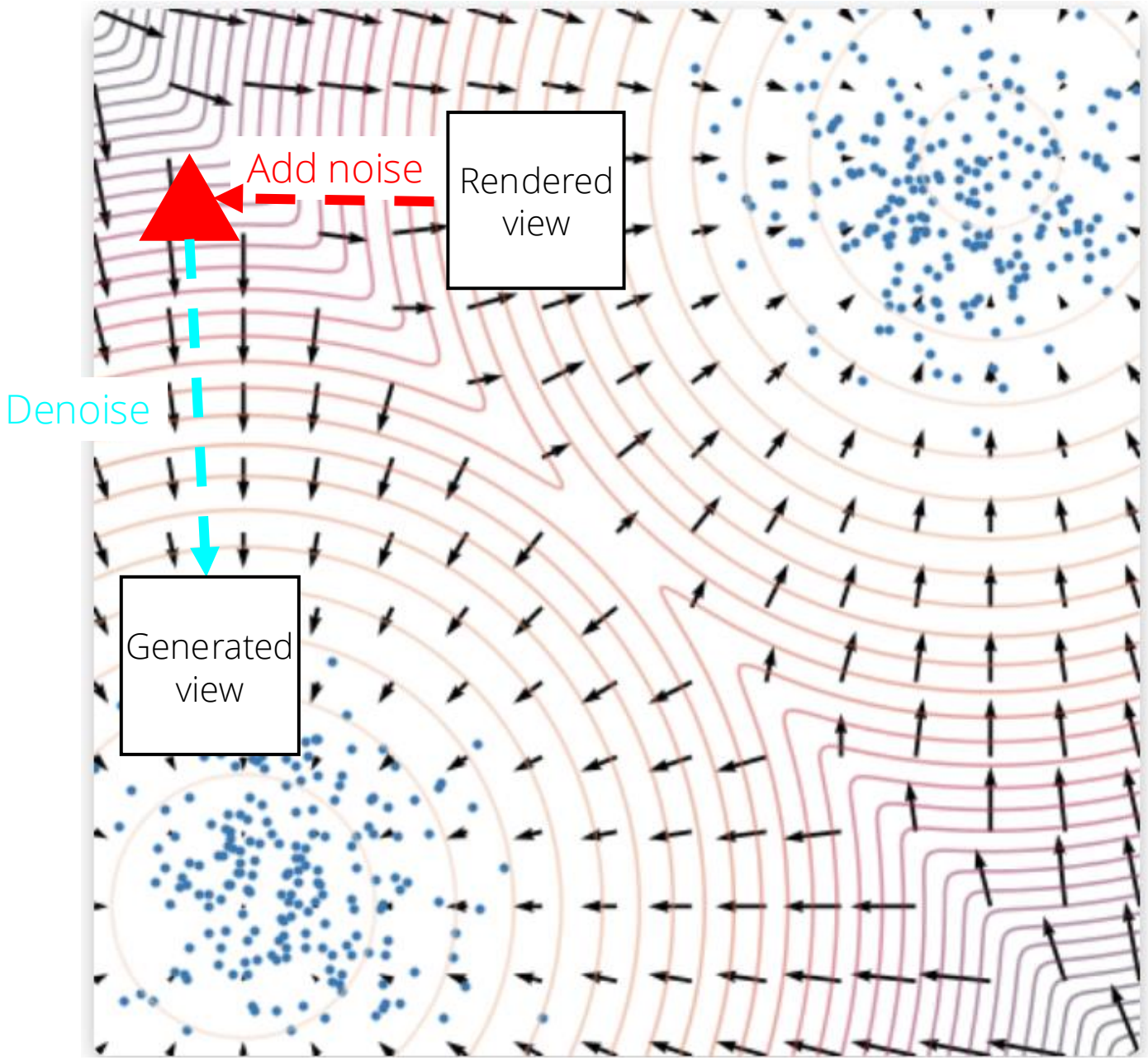
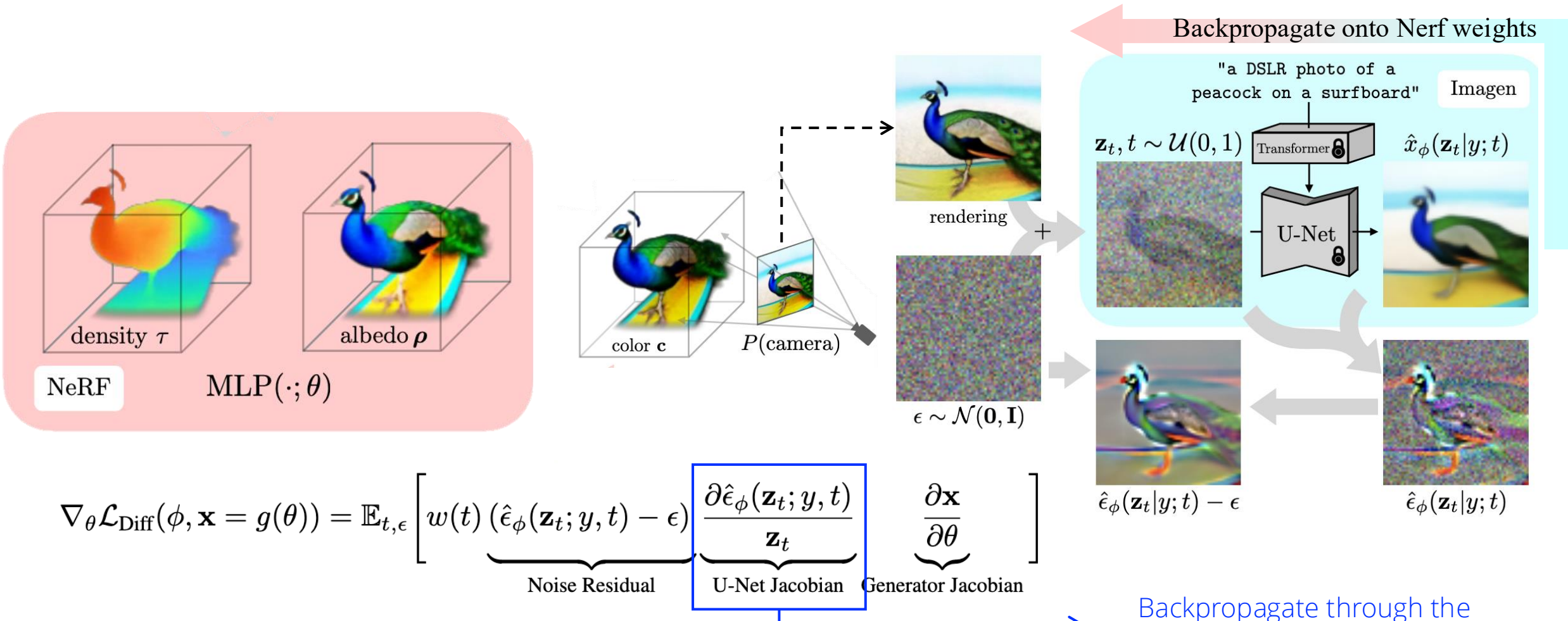


Image credit https://yang-song.net/blog/2021/score/

3D-from-2D Reconstruction Recipe: Render a View from Differentiable 3D Model

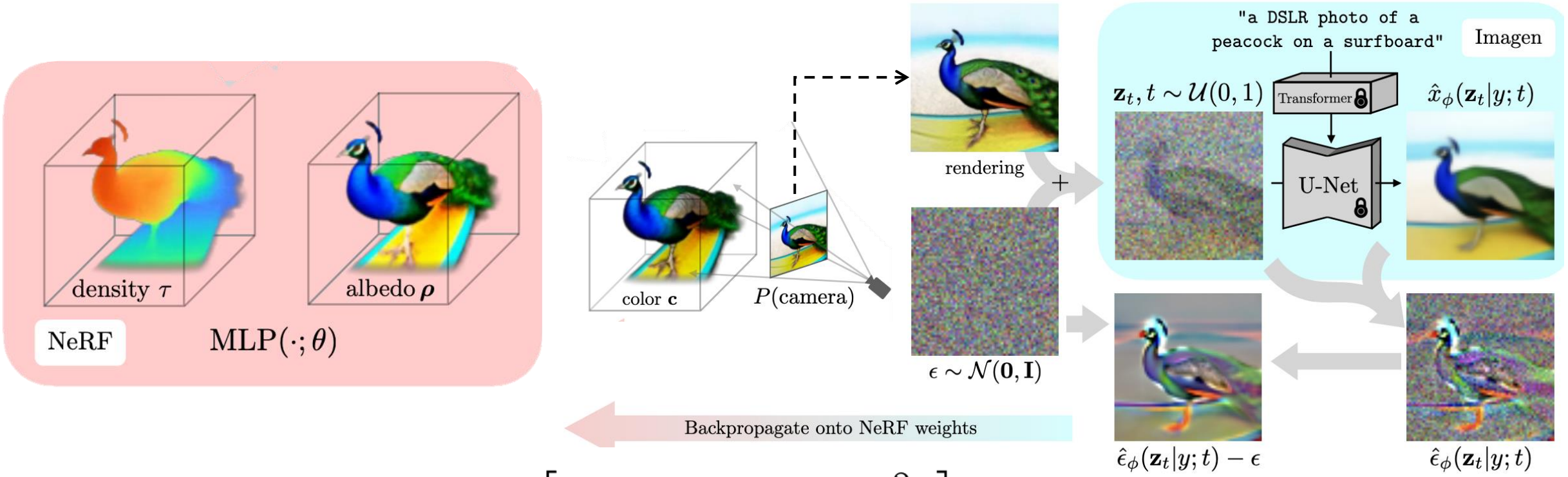


$$\nabla_\theta \mathcal{L}_{\text{Diff}}(\phi, \mathbf{x} = g(\theta)) = \mathbb{E}_{t,\epsilon}\left[w(t) \underbrace{(\hat{\epsilon}_\phi(\mathbf{z}_t; y, t) - \epsilon)}_{\text{Noise Residual}} \underbrace{\frac{\partial \hat{\epsilon}_\phi(\mathbf{z}_t; y, t)}{\mathbf{z}_t}}_{\text{U-Net Jacobian}} \underbrace{\frac{\partial \mathbf{x}}{\partial \theta}}_{\text{Generator Jacobian}} \right]$$

Backpropagate through the diffusion model is very expensive!

DreamFusion: Text-to3D Using 2D Diffusion. Poole et al. ICLR 2022.

3D-from-2D Reconstruction Recipe



$$\nabla_\theta \mathcal{L}_{\text{SDS}}(\phi, \mathbf{x} = g(\theta)) \triangleq \mathbb{E}_{t,\epsilon}\left[w(t)\left(\hat{\epsilon}_\phi(\mathbf{z}_t; y, t) - \epsilon\right)\frac{\partial \mathbf{x}}{\partial \theta}\right]$$

Approximate the gradients by bypassing the whole diffusion model

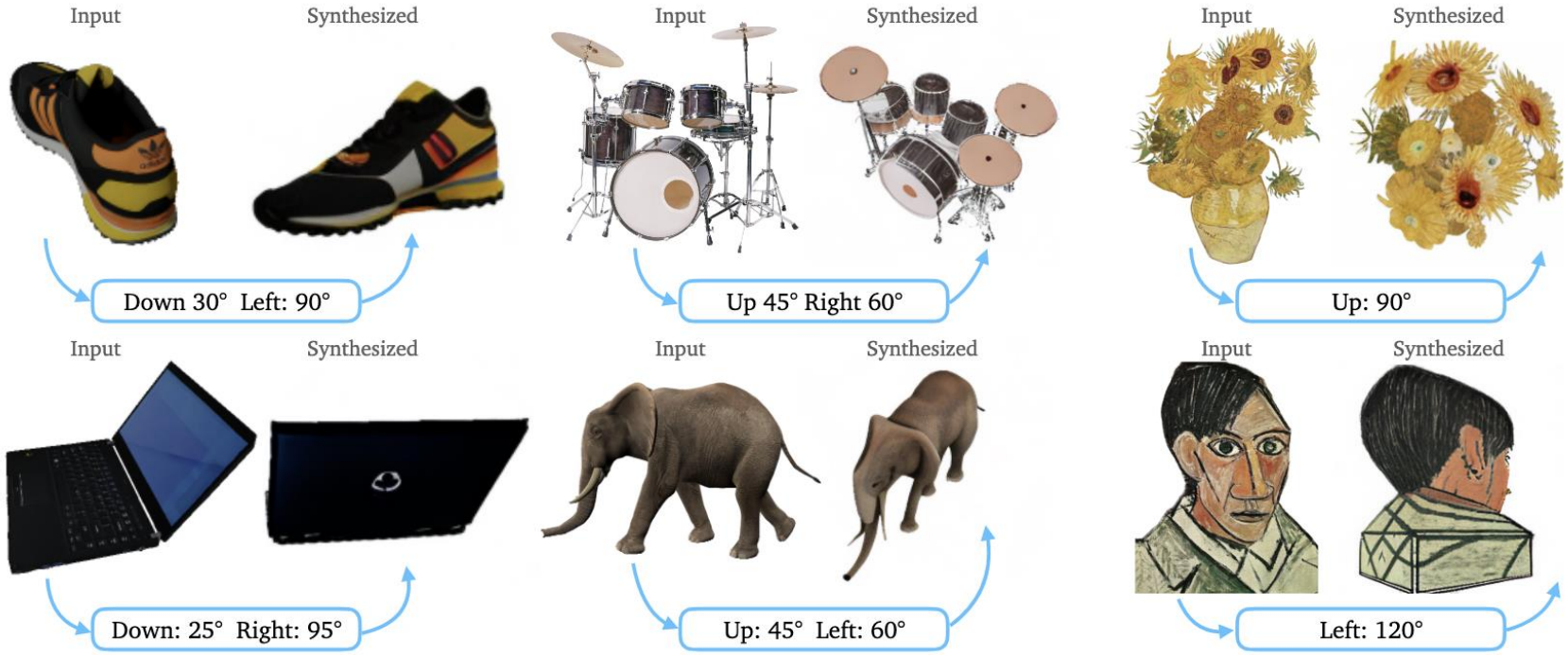
DreamFusion: Text-to3D Using 2D Diffusion. Poole et al. ICLR 2022.

Results: Distillation of 3D from Image Diffusion

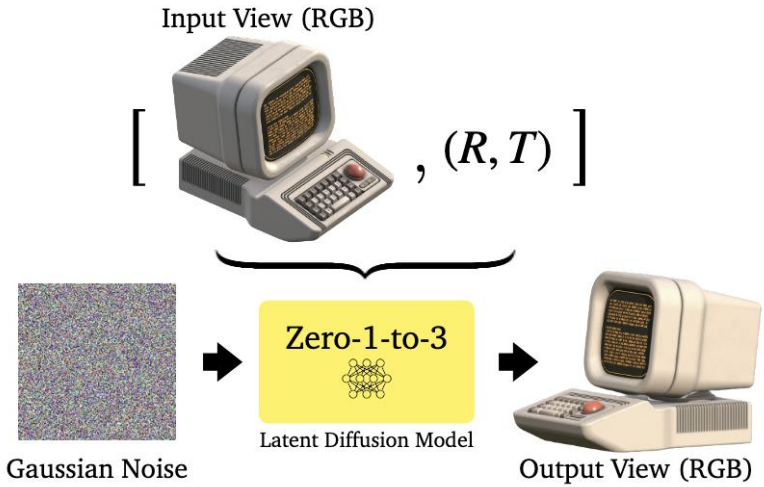
Problem: Without any 3D prior knowledge, generated views are inconsistent

DreamFusion: Text-to3D Using 2D Diffusion. Poole et al. ICLR 2022.

Better Ideas: Fine-tune Image Diffusion Model for Multi-View Generation



Enable conditioning on camera angles



Zero-1-to-3: Zero-shot One Image to 3D Object. Liu et al. ICCV 2023.

Generate Training Data of Camera Poses and Rendered Views from 3D Asset Dataset

Source	# Objects
IKEA [32]	219
GSO [17]	1K
EGAD [41]	2K
OmniObject3D [63]	6K
PhotoShape [46]	5K
ABO [13]	8K
Thingi10K [67]	10K
3d-Future [19]	10K
ShapeNet [9]	51K
Objaverse 1.0 [14]	800K
Objaverse-XL	**10.2M**

Results: More Consistent View Generation

Zero-1-to-3: Zero-shot One Image to 3D Object. Liu et al. ICCV 2023.

Inconsistency is still the Problem

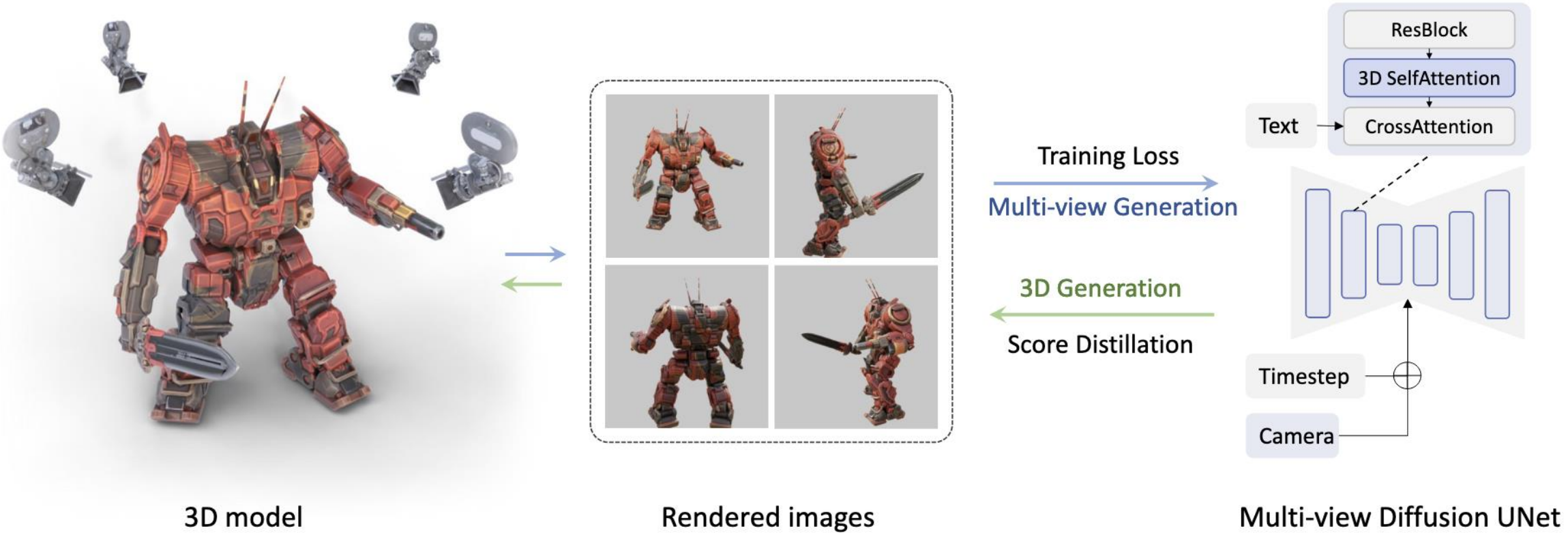
Multi-face Janus Problem

Content Drift Problem

Figure 1: Typical multi-view consistency problems of 2D-lifting methods for 3D generation. Left: “A photo of a horse walking” where the horse has two faces. Right: “a DSLR photo of a plate of fried chicken and waffles with maple syrup on them” where the chicken gradually becomes a waffle.

MVDream: Multi-view Diffusion for 3D Generation. Shi et al. NeuRIPS 2023.

Idea: Concurrently Predict Multiple Views



MVDream: Multi-view Diffusion for 3D Generation. Shi et al. NeuRIPS 2023.

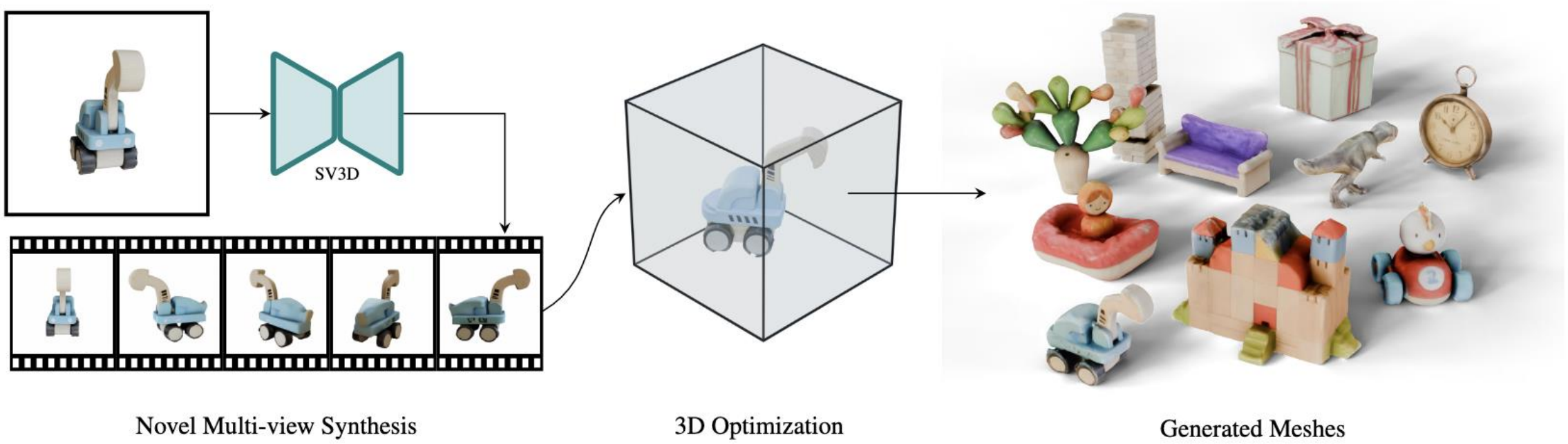
More Consistent Multi-View Generation

a bald eagle carved out of wood

a bulldog wearing a black pirate hat

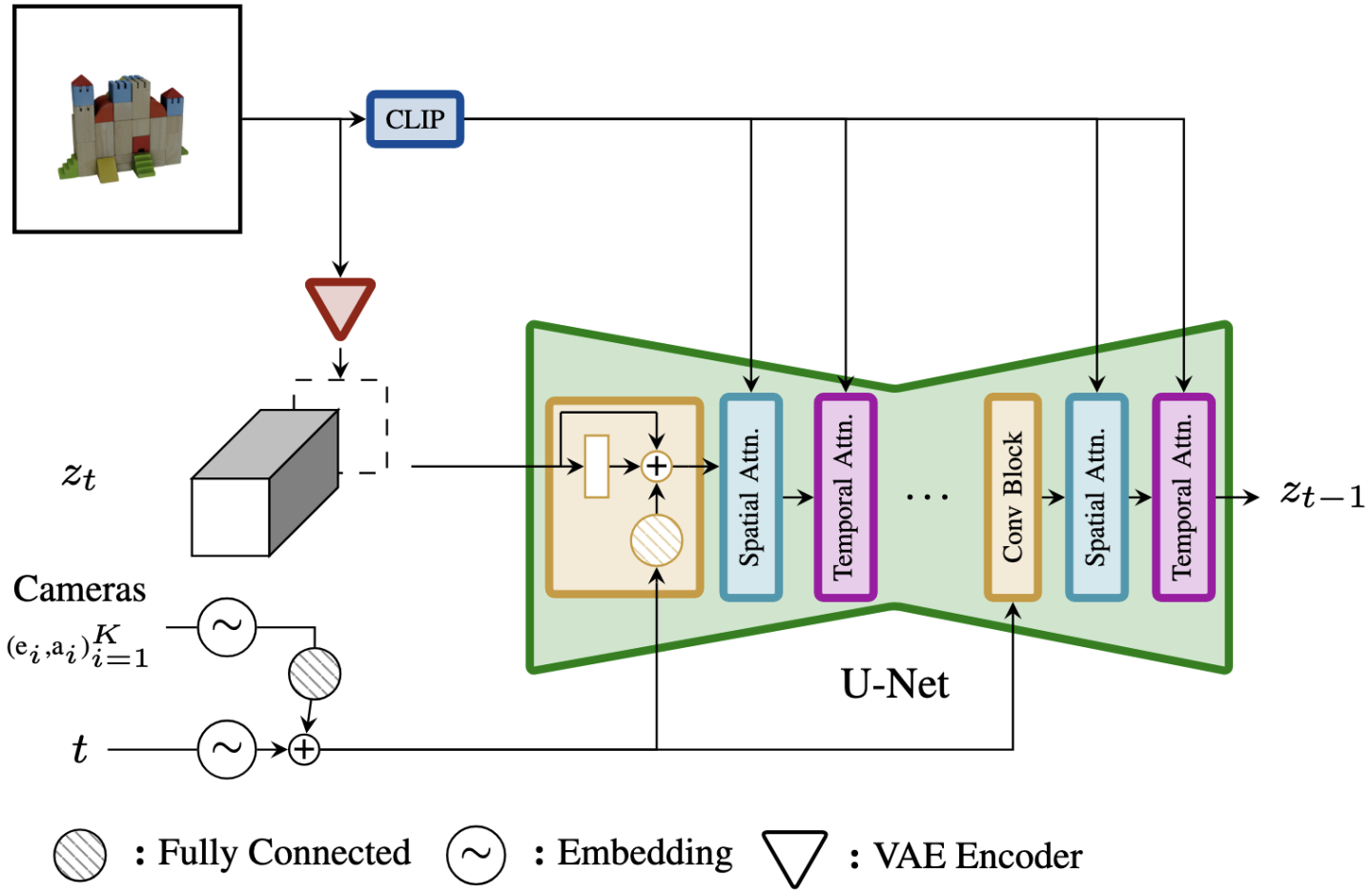
MVDream: Multi-view Diffusion for 3D Generation. Shi et al. NeuRIPS 2023.

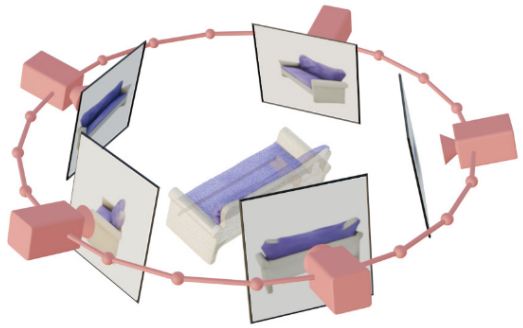
A Better Idea: Concurrently Predict Videos of Multiple Views



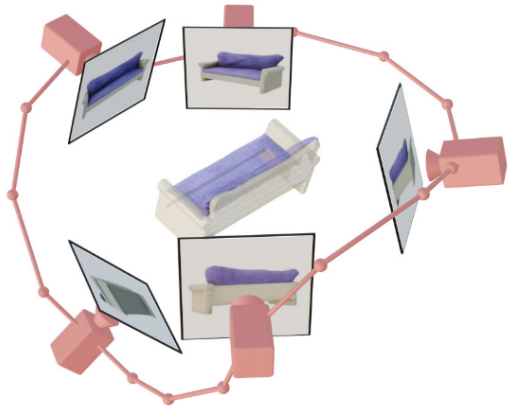
SV3D: Novel Multi-view Synthesis and 3D Generation from a Single Image using Latent Video Diffusion. Voleti et al. ECCV 2024.

Camera-view Trajectory as Video



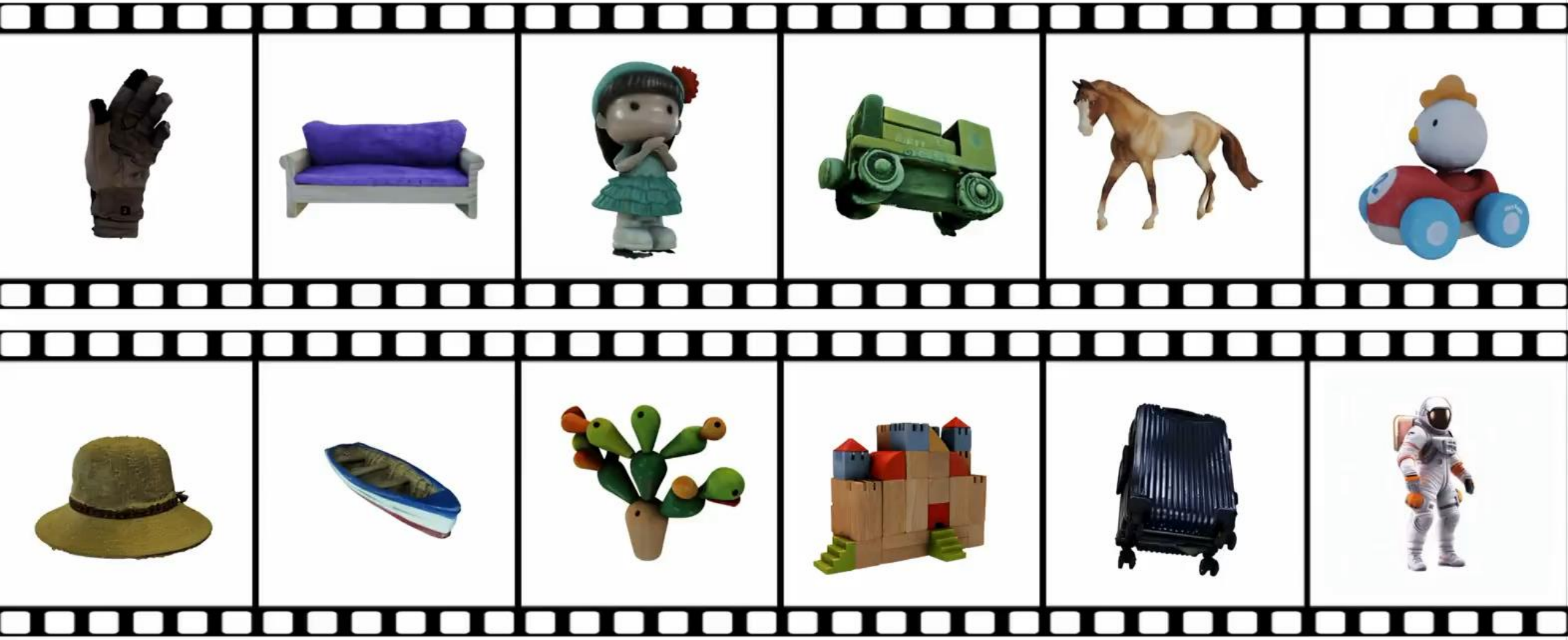


Static



Dynamic

SV3D: Novel Multi-view Synthesis and 3D Generation from a Single Image using Latent Video Diffusion. Voleti et al. ECCV 2024.

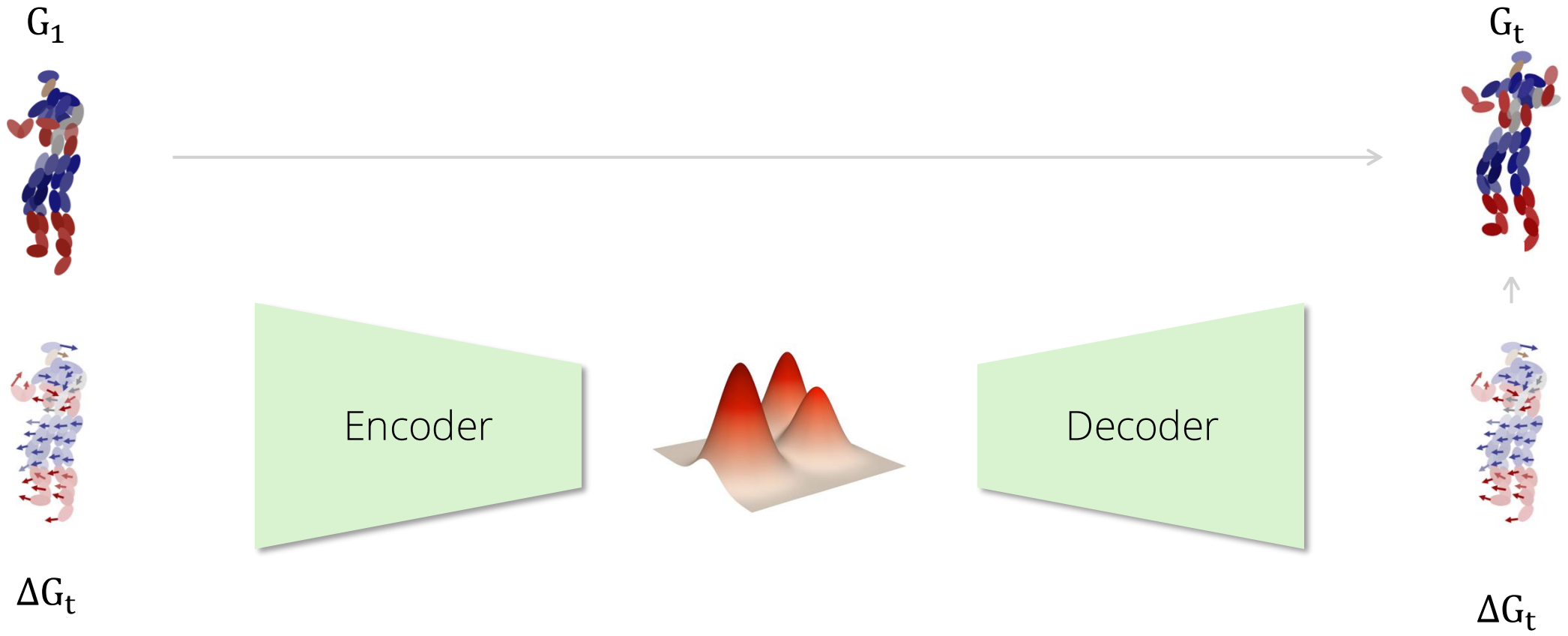


SV3D: Novel Multi-view Synthesis and 3D Generation from a Single Image using Latent Video Diffusion. Voleti et al. ECCV 2024.

Content

- Generative Modeling
  - ➢Autoregressive Models
  - ➢Variational AutoEncoders
  - ➢Denoising Diffusion Probabilistic Models
  - ➢Flow Matching Models
- 4D Generative Modeling
  - ➢3D Geometry Generation
  - ➢3D Generation with Multi-view Video Generation
  - ➢4D Geometry Generation
  - ➢4D Generation with Multi-view Video Generation

We Have Covered in Previous Lecture. The Idea is to Decouple Shape and Motion Generation



Gaussian Variation Field Diffusion for High-fidelity Video-to-4D Synthesis. Zhang et al.

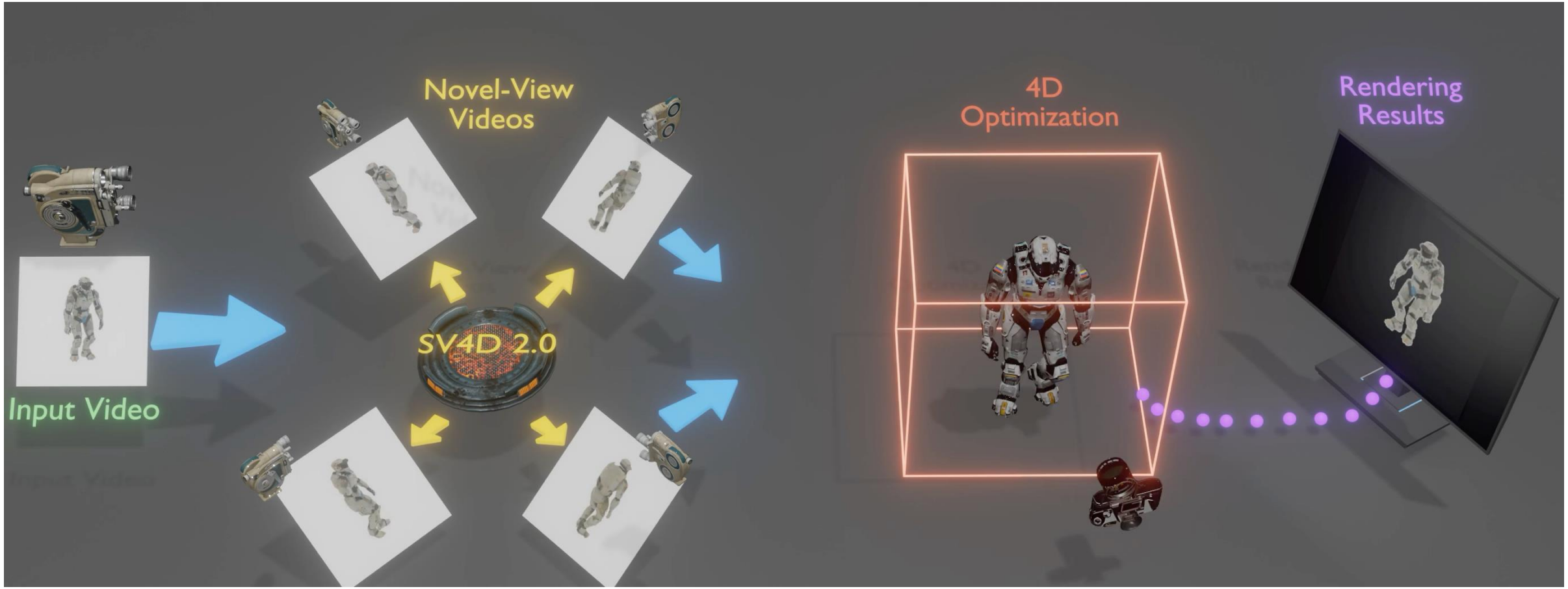
Content

- Generative Modeling
  - Autoregressive Models
  - Variational AutoEncoders
  - Denoising Diffusion Probabilistic Models
  - Flow Matching Models
- 4D Generative Modeling
  - 3D Geometry Generation
  - 3D Generation with Multi-view Video Generation
  - 4D Geometry Generation
  - 4D Generation with Multi-view Video Generation

How to Generate a 4D World with Video Diffusion?

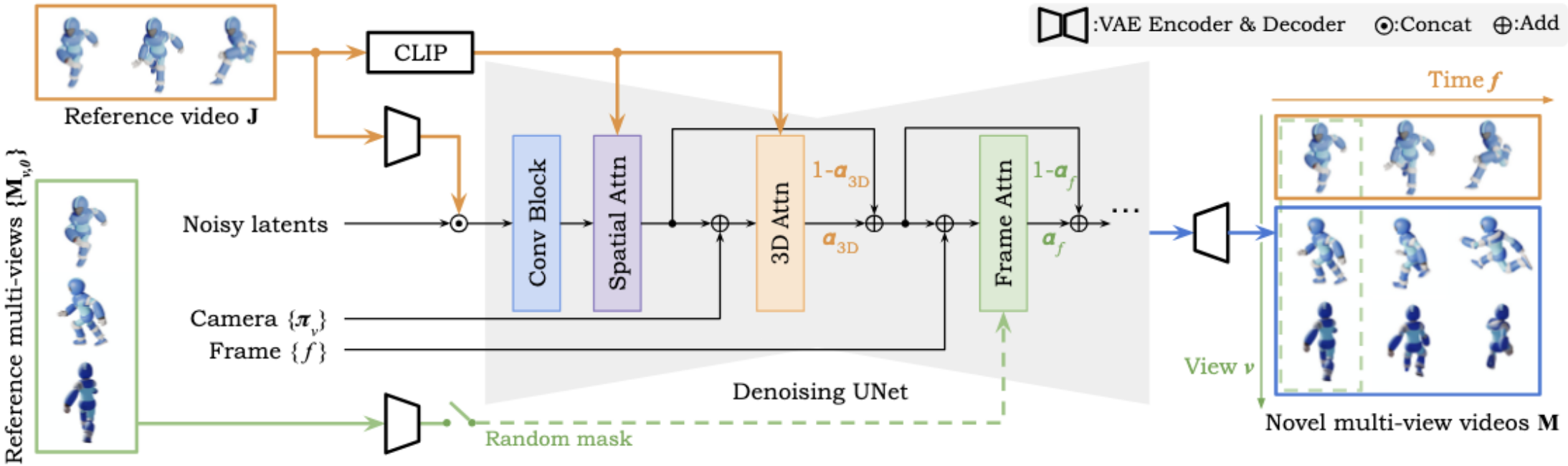
- How to generate a 4D world?
  - Generate multi-view 2D videos and lift to 4D
  - Generate multi-view 2D images and lift to 3D, and generate motion of these 3D assets
  - Generate 3D videos

Idea: Generate Multi-view Videos and Lift to 4D



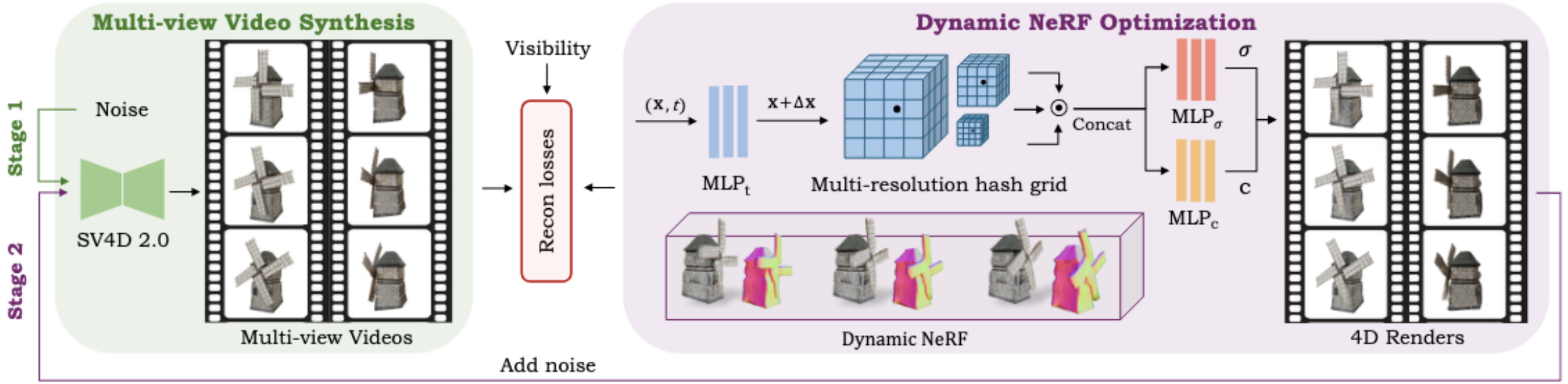
SV4D 2.0: Enhancing Spatio-Temporal Consistency in Multi-View Video Diffusion for High-Quality 4D Generation. Yao et al.

Generate Multi-view Videos and Lift to 4D



SV4D 2.0: Enhancing Spatio-Temporal Consistency in Multi-View Video Diffusion for High-Quality 4D Generation. Yao et al.

Generate Multi-view Videos and Lift to 4D



Generate Multi-view Videos and Lift to 4D

Input Video

4D Optimization

Input Video

4D Optimization

Input Video

4D Optimization

Input Video

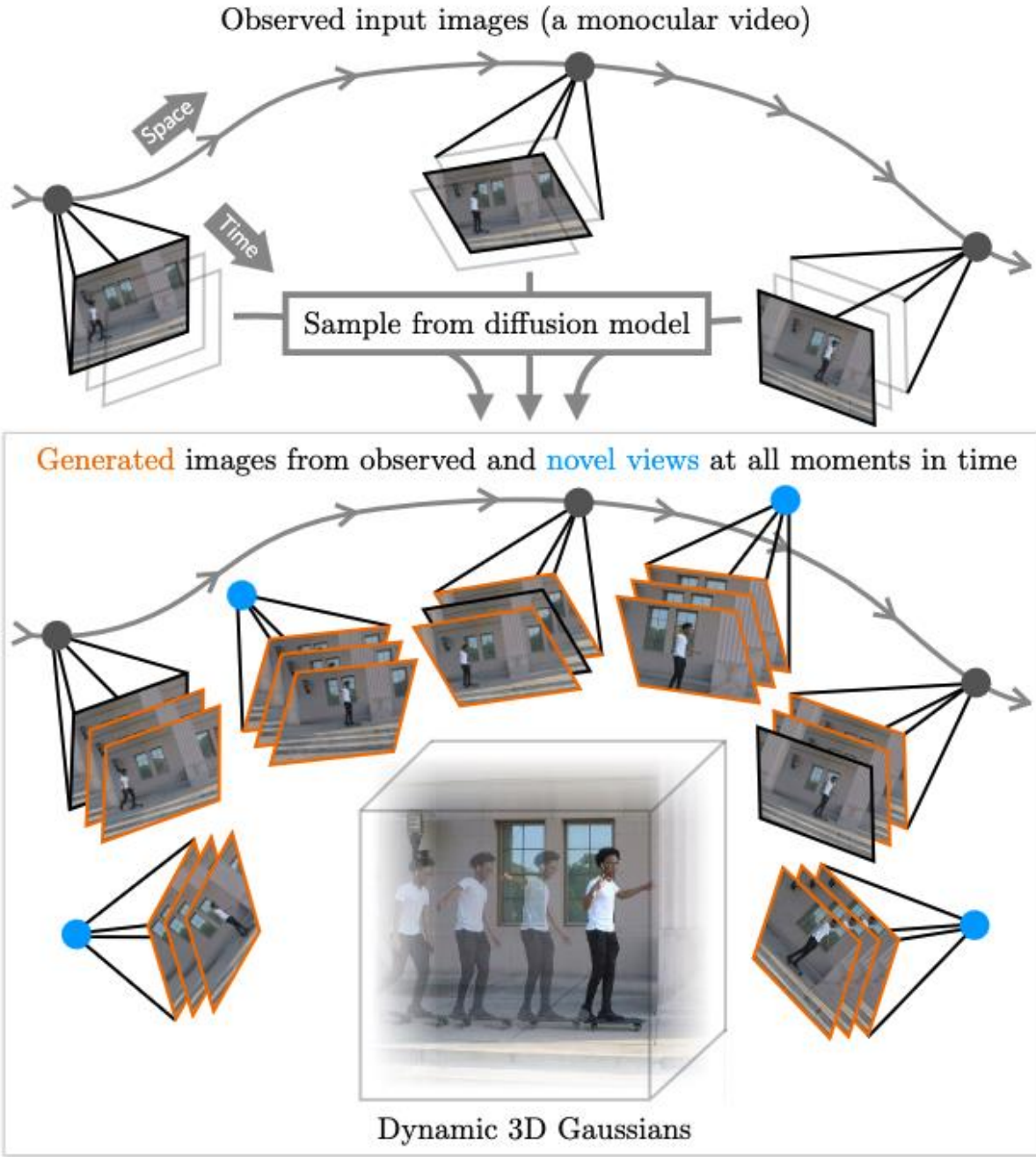
4D Optimization

SV4D 2.0: Enhancing Spatio-Temporal Consistency in Multi-View Video Diffusion for High-Quality 4D Generation. Yao et al.

CAT4D: Create Anything in 4D with Multi-View Video Diffusion Models

Rundi Wu[1,2] Ruiqi Gao[1] Ben Poole[1] Alex Trevithick[1,3]
Changxi Zheng[2] Jonathan T. Barron[1] Aleksander Hołyński[1]

[1]Google DeepMind [2]Columbia University [3]UC San Diego

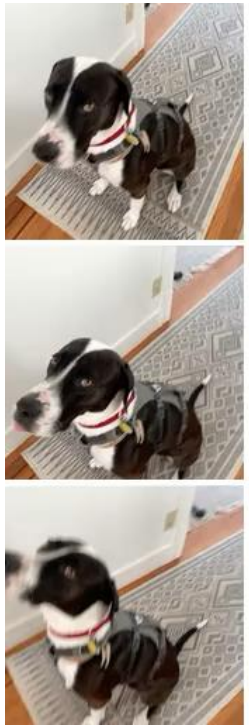


CAT4D: Create Anything in 4D with Multi-View Video Diffusion Models. Wu et al.

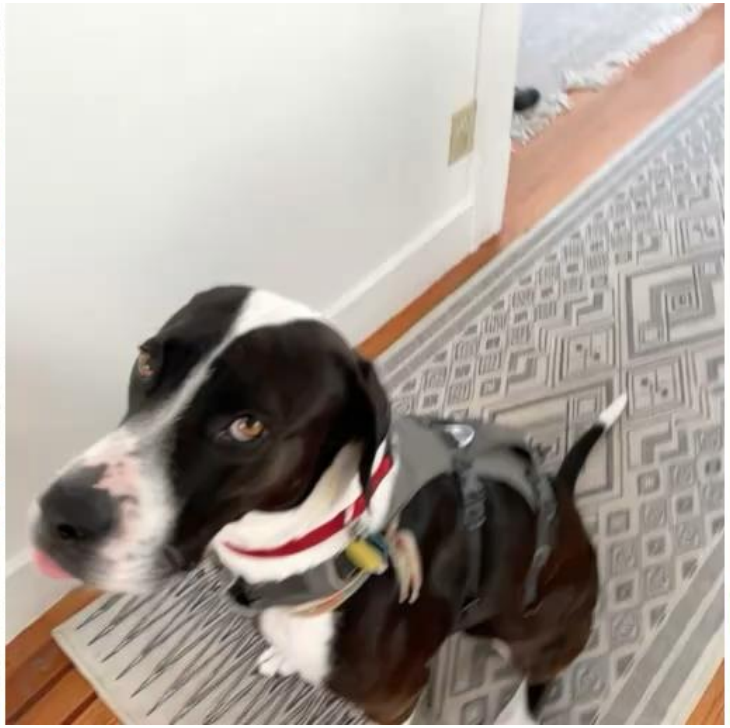

Sample from multi-view video diffusion model
Optimize dynamic 3D Gaussians
Input video
Generated multi-view videos
Dynamic 3D scene

Results: 4D Reconstruction

input



Fixed View Varying Time



Varying View Fixed Time

Varying View Varying Time

CAT4D: Create Anything in 4D with Multi-View Video Diffusion Models. Wu et al.

Results: 4D Reconstruction

input

Fixed View Varying Time

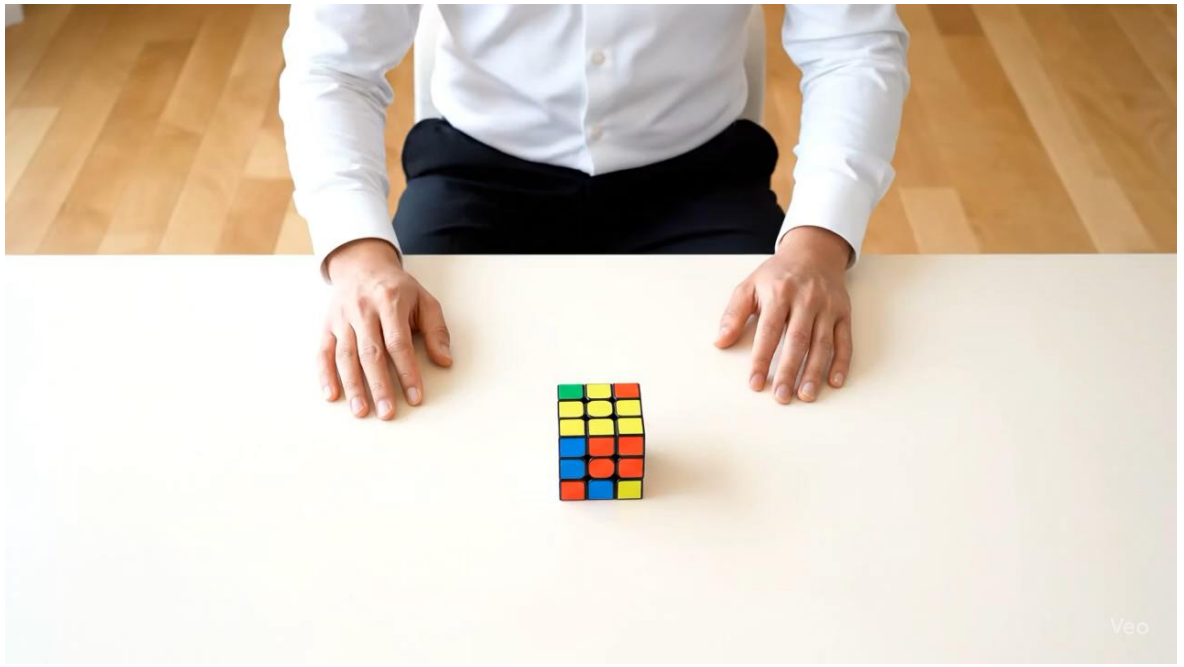
Varying View Fixed Time

Varying View Varying Time

Have you seen the problem?

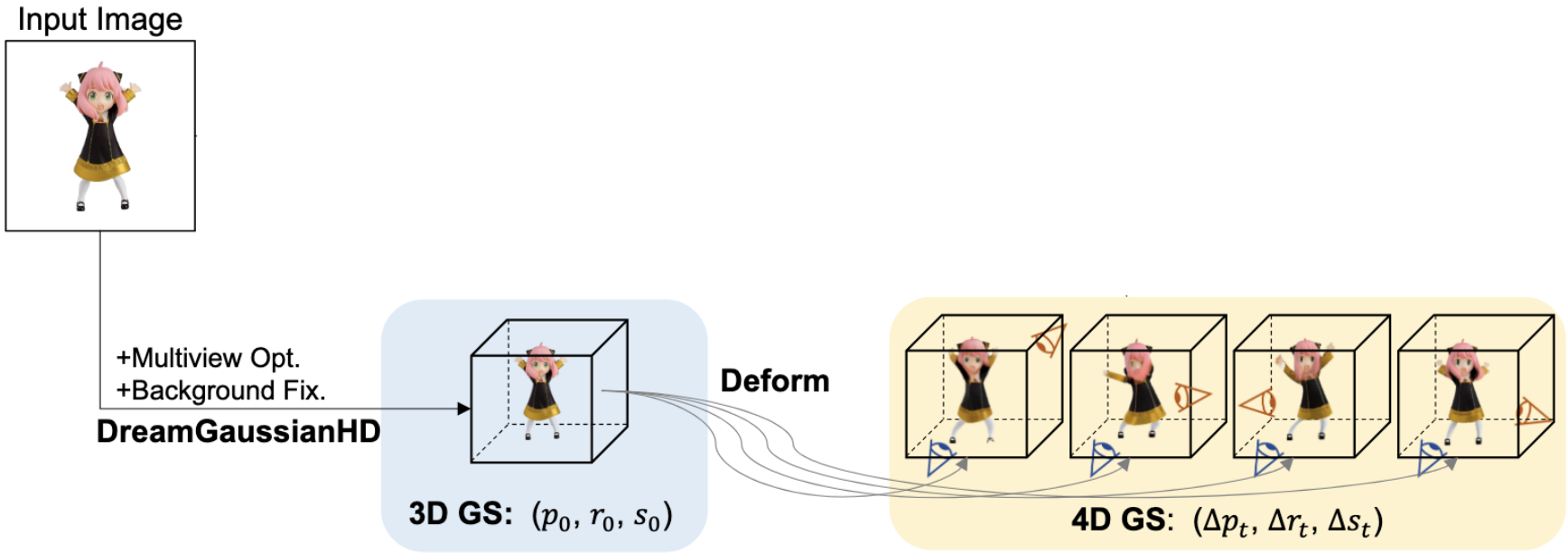
Generated Videos are Often Physically Incorrect. We Need to Model the Motion Explicitly.

Video generated by Veo3



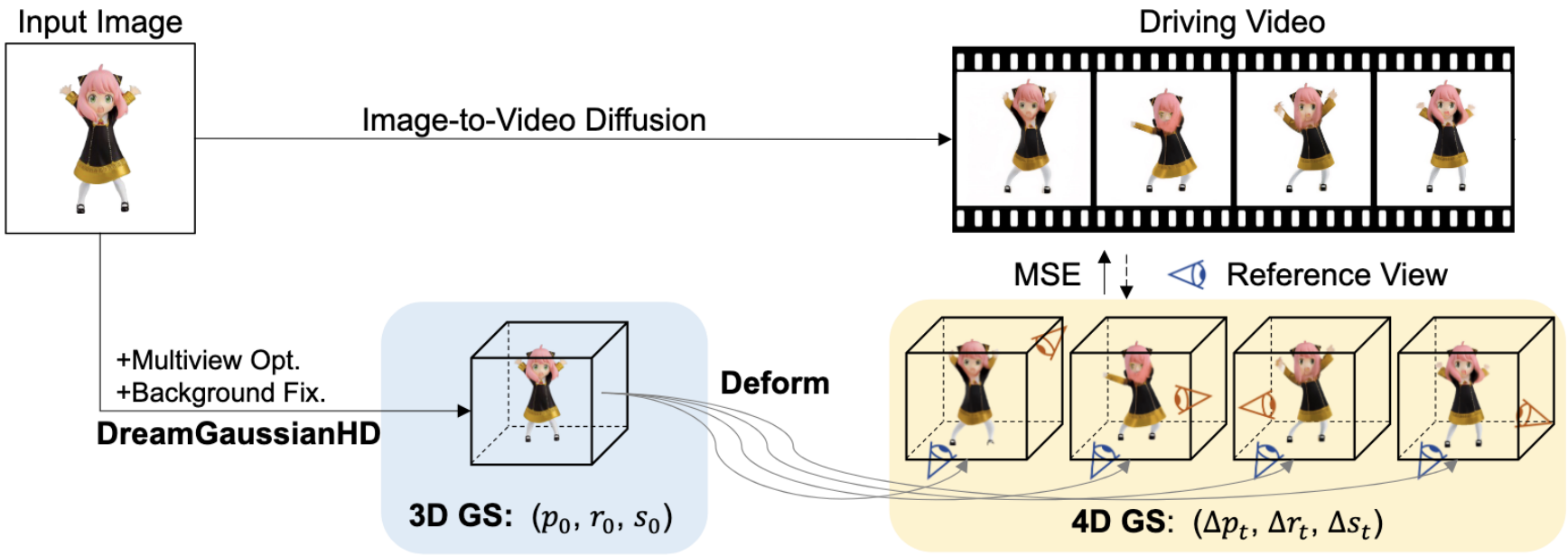
Video generated by Veo3

Idea: Create 3D Assets by Generating Multi-view Images and Generate the 3D Motion

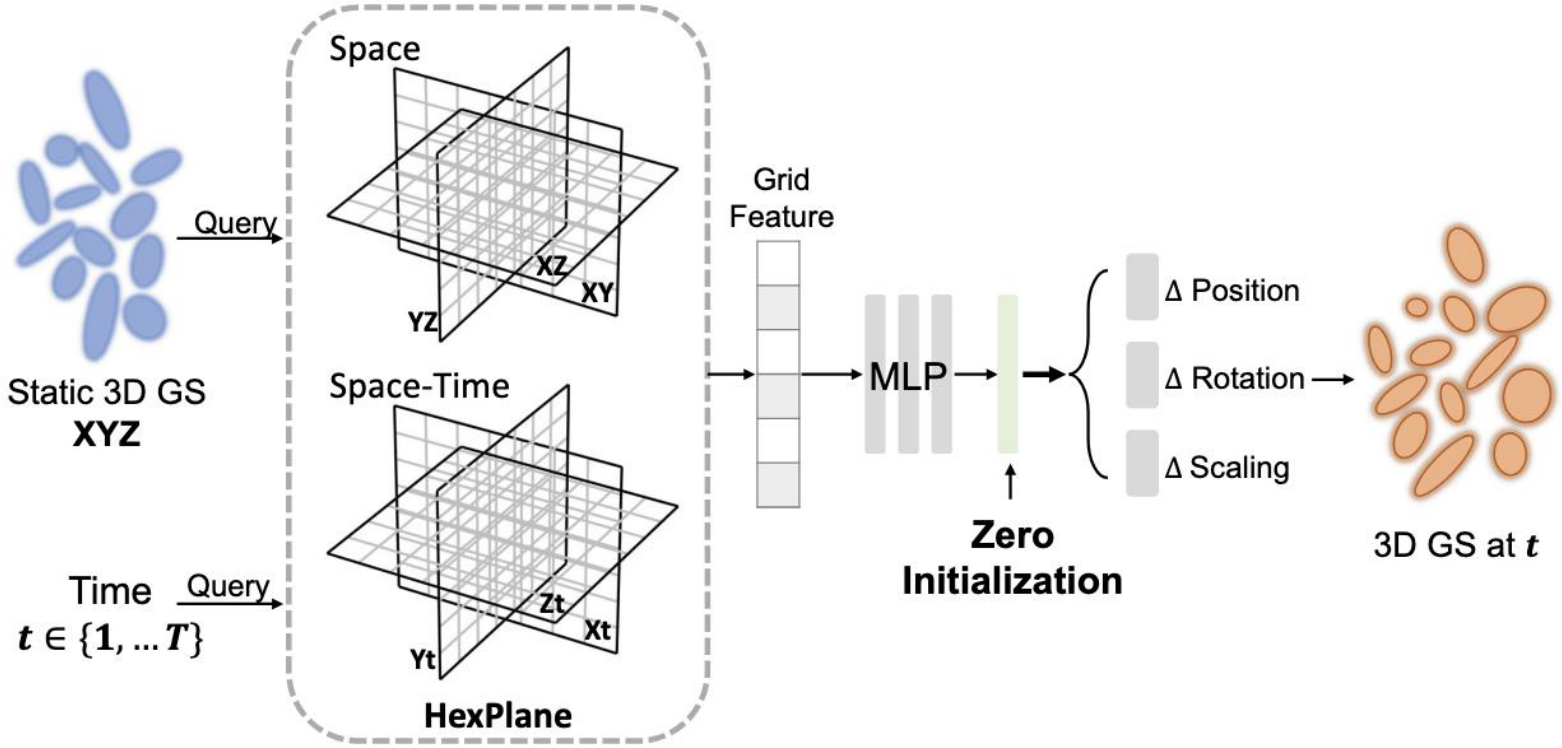


DreamGaussian4D: Generative 4D Gaussian Splatting. Ren et al.

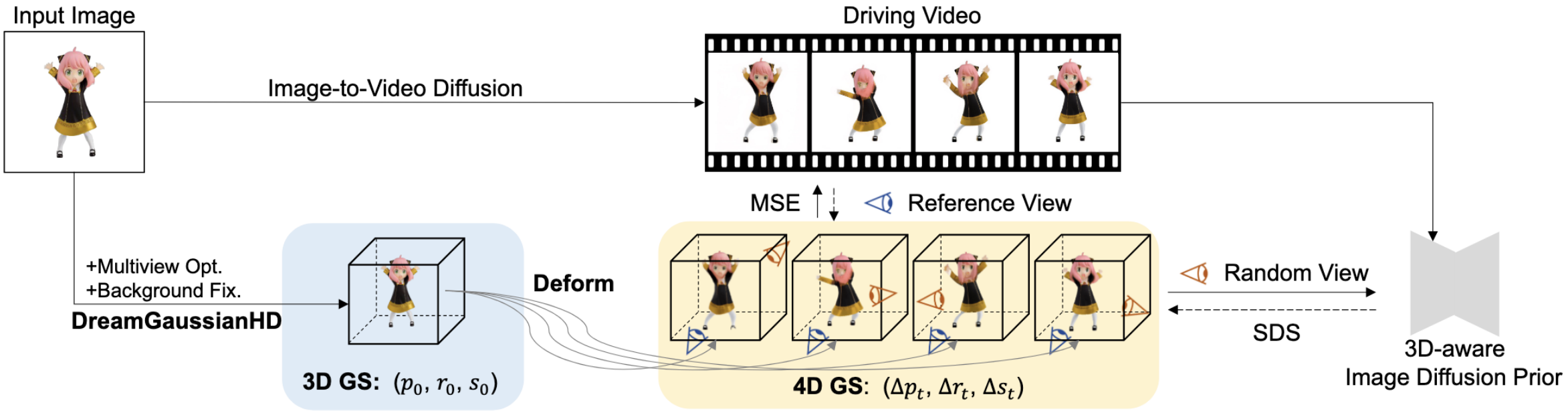
Generate the Motion Field from a Reference Video



Predict the Motion of 3D Gaussians with a Neural Network



Create 3D Assets by Generating Multi-view Images and Generate the 3D Motion



Given a single-view image:

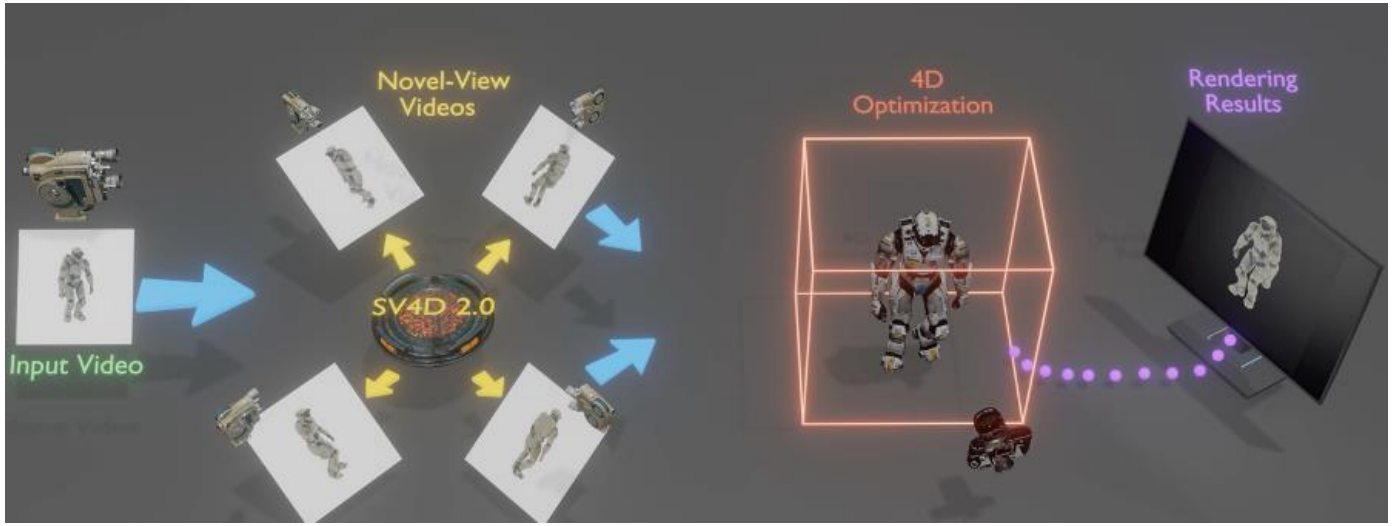
1. 3D-from-2D reconstruction
2. Flow prediction which deforms the 3D model at each frame
3. Image-to-Video prediction to guide the optimization of flow prediction

Once we have deformed 3D model at each frame:

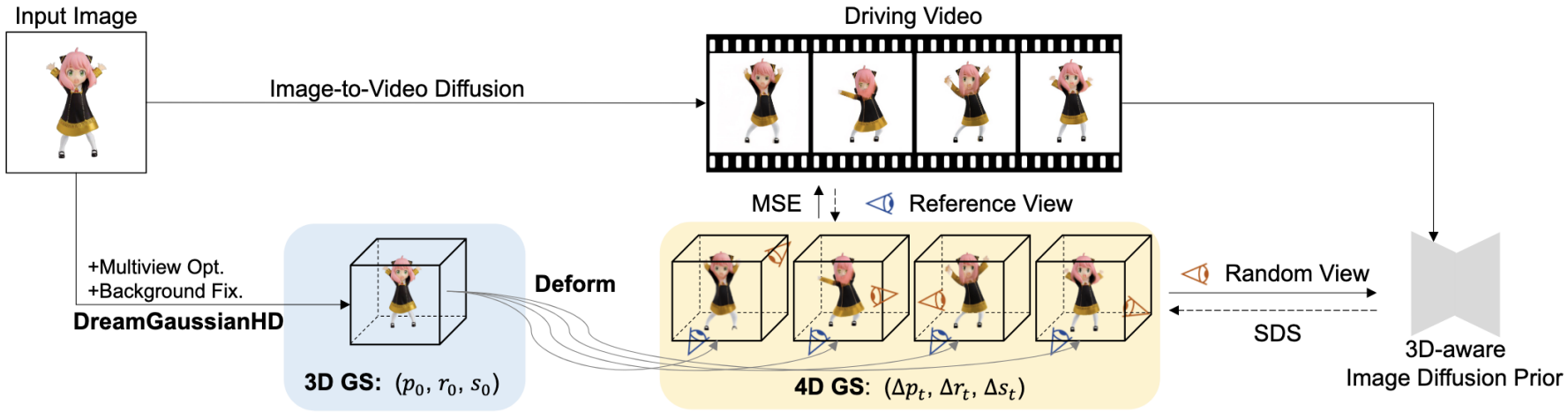
$$\mathcal{L}_{\text{Ref}} = \frac{1}{\mathcal{T}} \sum_{\tau=1}^{\mathcal{T}} ||f(\phi(S,\tau), o_{\text{Ref}}) - I^{\tau}_{\text{Ref}}||_2^2,$$

$$\nabla_{\phi}\mathcal{L}_{\text{SDS}} = \mathbb{E}_{t,\tau,\epsilon,o}[(\epsilon_{\theta}(\hat{I}; t, I^{\tau}_{\text{Ref}}, o) - \epsilon)\frac{\partial I}{\partial \phi}], \text{ and } \hat{I} = f(\phi(S,\tau), o),$$

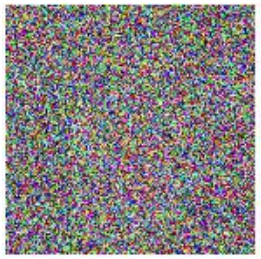
What are the Problems?

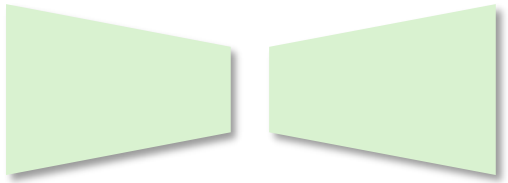


- The distribution of motion is implicitly captured by the video generator
- The distribution of motion is coupled with the distribution of appearance
- Can we generate generate motion explicitly?



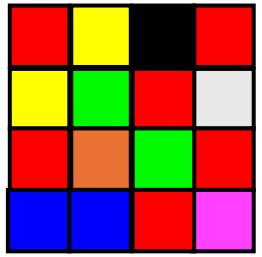
Idea: Generate 3D Videos

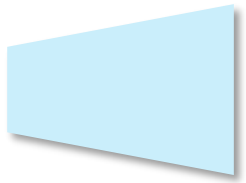




Generative model

Low-resolution features



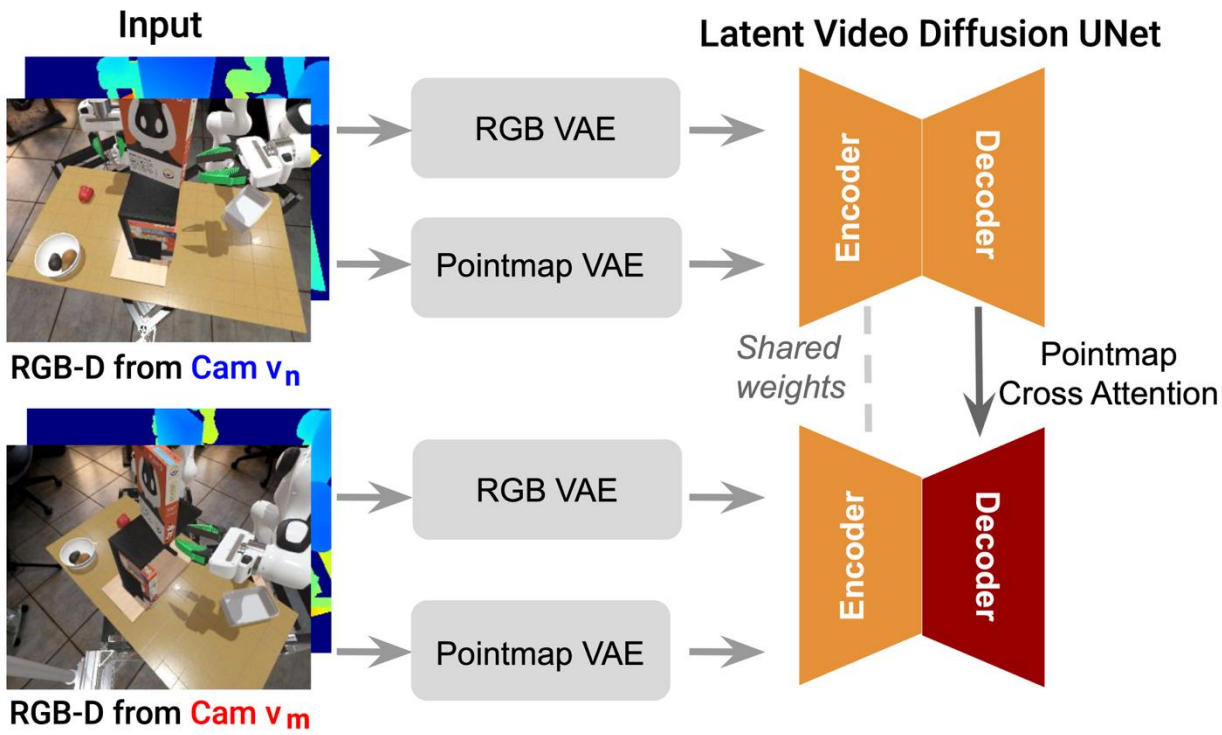


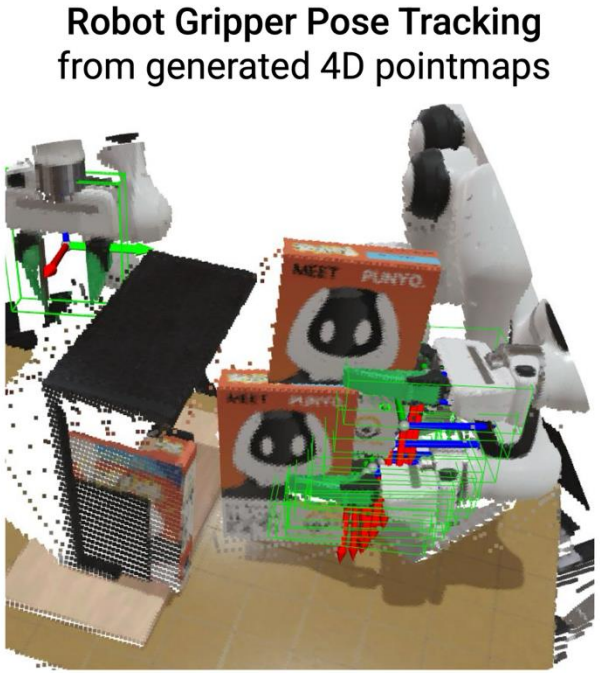
Decoder

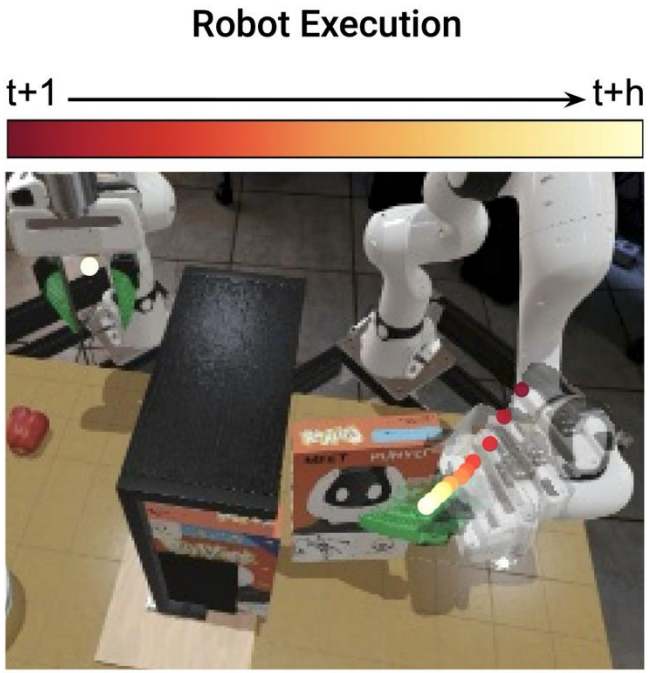
RGB image

XYZ Point map

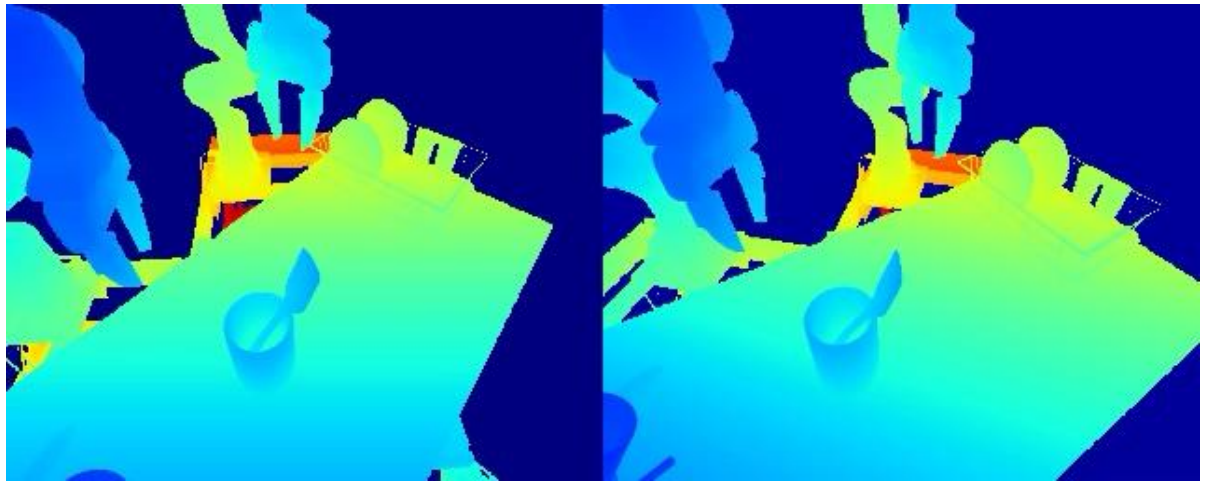
3D Video Generator



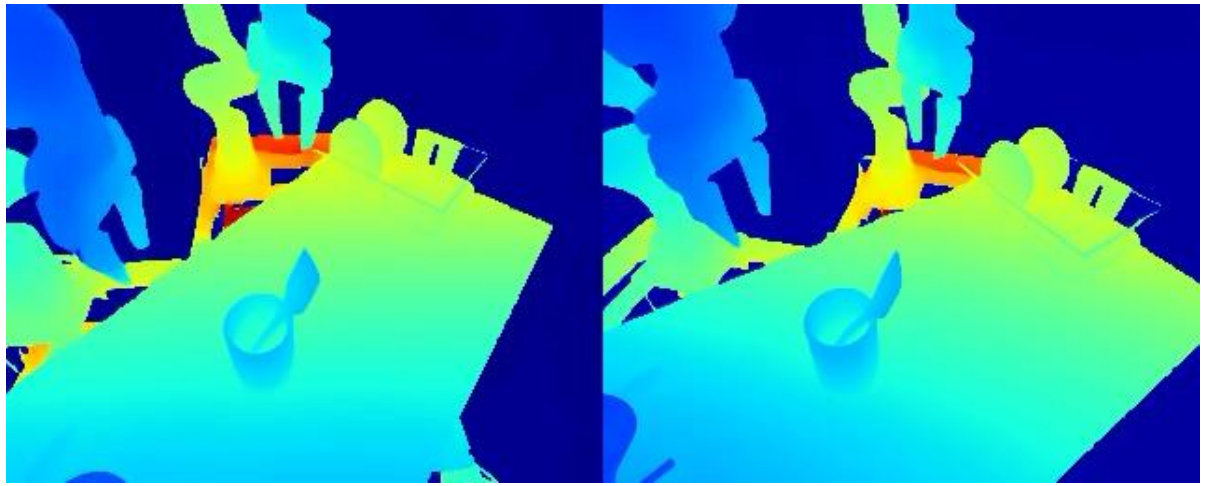




Ground-truth

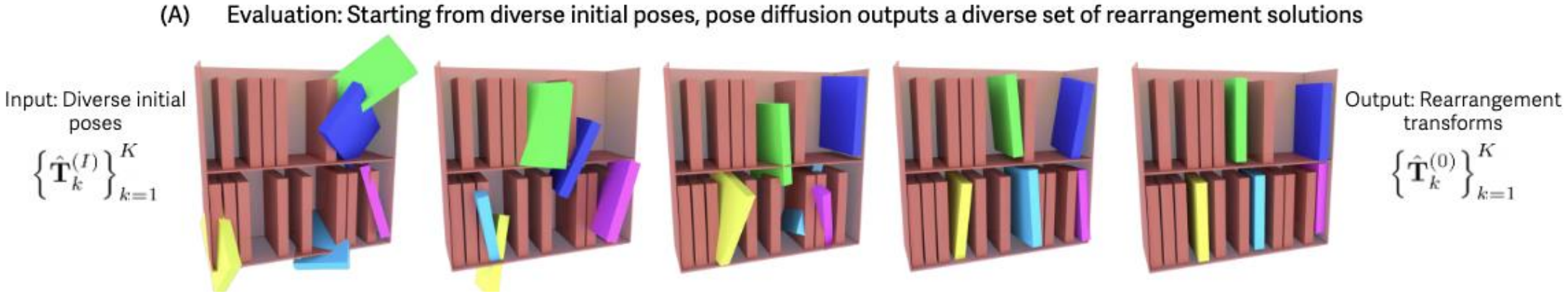


Generation



What Other Motion Can We Generate with Diffusion Models?

Diffusion Models that Generate Rigid-body Motion



- Idea: Use a diffusion model to generate 6 DoF object poses for arranging objects in a scene
- Problem: 6 DoF object poses belongs to $SE(3)$ group (a special set of 3×4 matrices). However, the diffusion model we discussed is based on the Gaussian formulation. How to generalize the diffusion model to $SE(3)$?

Shelving, Stacking, Hanging: Relational Pose Diffusion for Multi-modal Rearrangement. Simeonov et al.

Generalize Diffusion Models to $SE(3)$

- Idea 1: Formulate the noising and denoising process with $SE(3)$

SE(3)-DiffusionFields: Learning smooth cost functions for joint grasp and motion optimization through diffusion

Julen Urain*[1], Niklas Funk*[1], Jan Peters[1,2,3,4], Georgia Chalvatzaki[1]

Fig. 1: Pick and place task in which the robot has to pick a mug and move it to the target pose (in the shelves) without colliding. We exploit diffusion models for jointly optimizing both grasp and motion and show the successful trajectory from left to right.

Generalize Diffusion Models to $SE(3)$

- Idea 2: Extend the definition of noise from a Gaussian noise to a random transform $T_{\Delta}^{rand} \in SE(3)$, and define the sample at diffusion step i as a transform $T^{(i)} \in SE(3)$

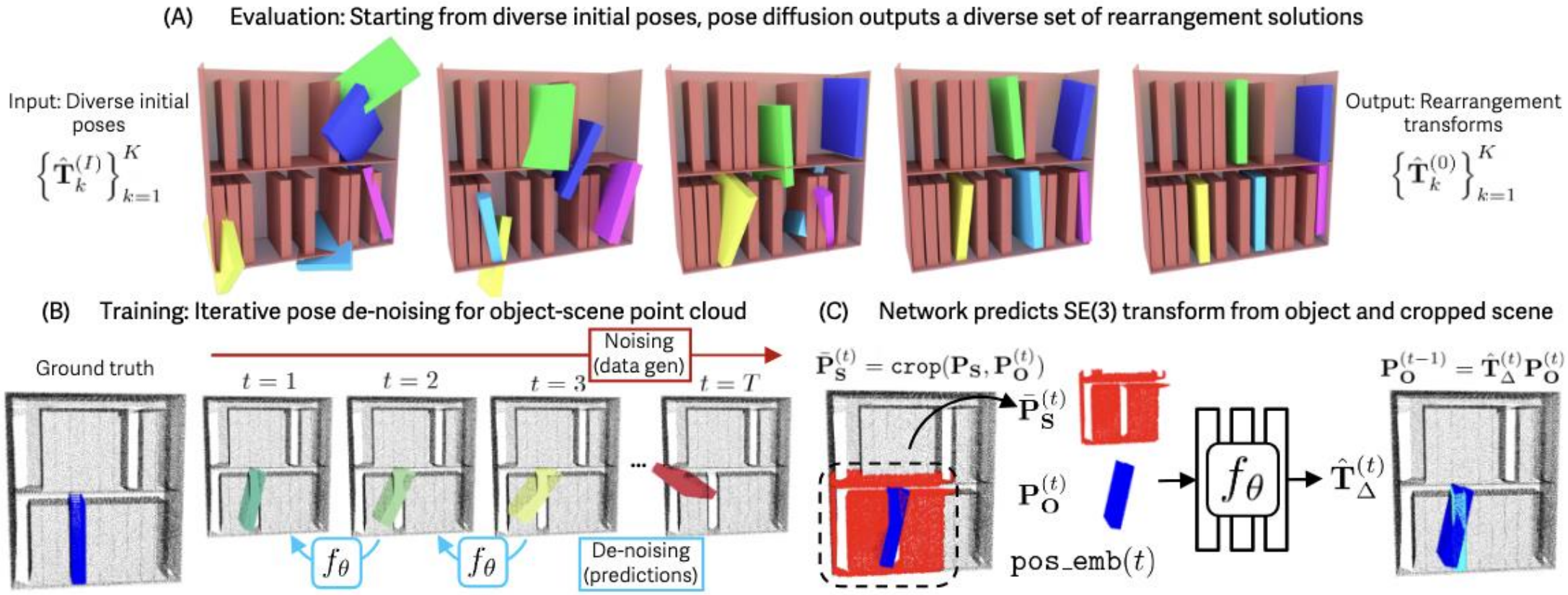
$$x_{i-1} = \frac{1}{\sqrt{\alpha_i}}\left(x_i - \frac{1-\alpha_i}{\sqrt{1-\bar{\alpha}_i}}\epsilon_\theta(x_i, i)\right) + \delta_i \boldsymbol{z}$$

$$T^{(i-1)} = T_{\Delta}^{rand} T_{\Delta}^{i} T^{(i)}$$

- In DDPM, we sample from a Gaussian noise and denoising it into a sample. Here, initial rotations are drawn from a uniform grid over $SO(3)$ and translations are uniformly drawn from the bounding box of the scene.

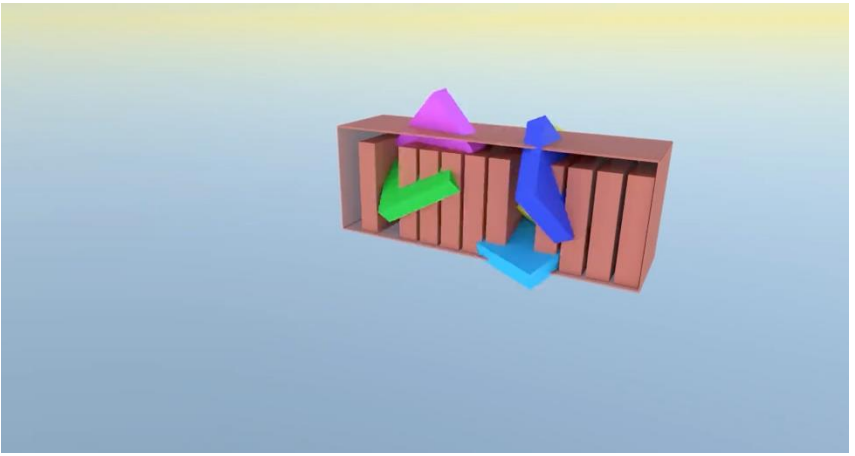
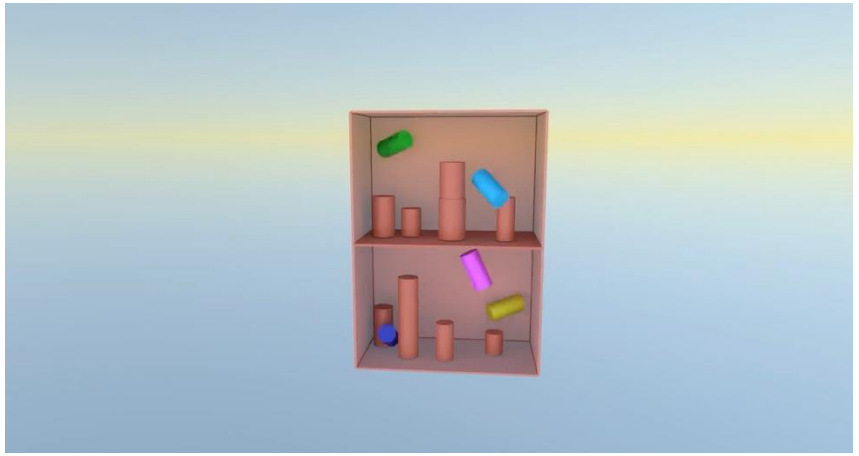
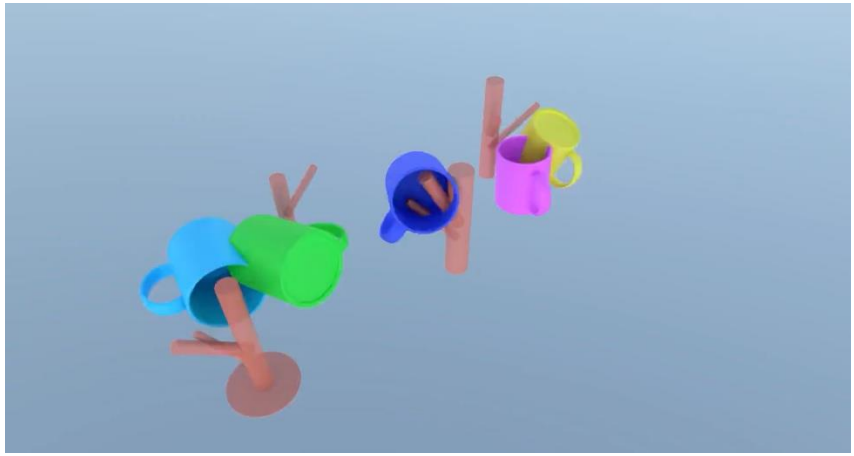
Shelving, Stacking, Hanging: Relational Pose Diffusion for Multi-modal Rearrangement. Simeonov et al.

Diffusion Models that Generate Rigid-body Motion

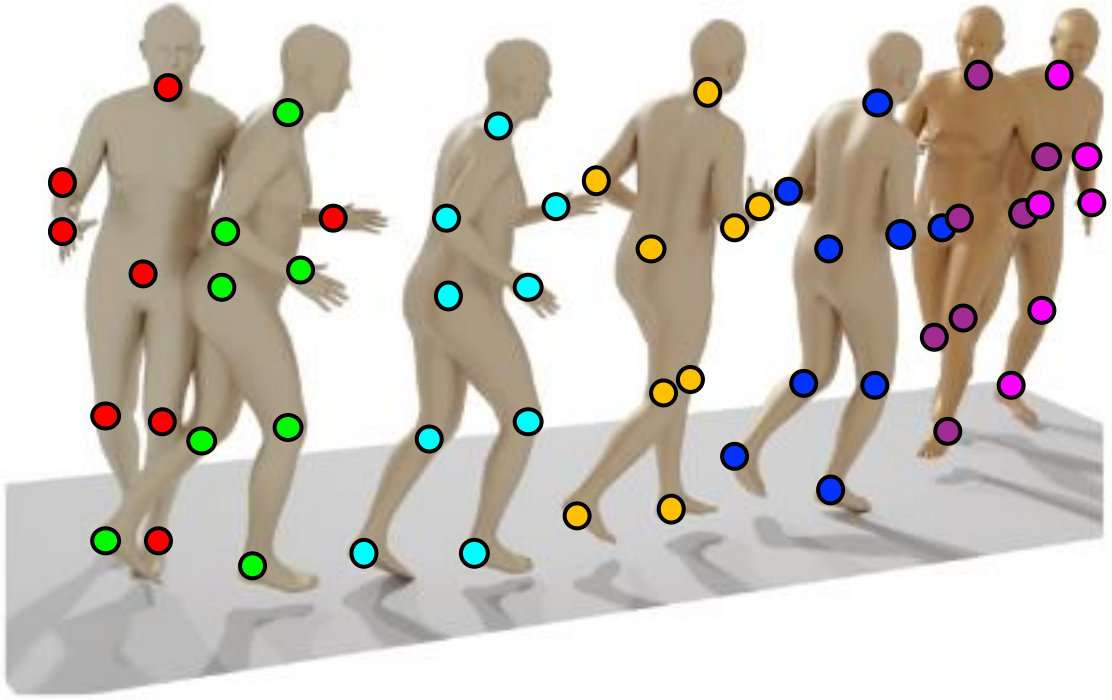


Shelving, Stacking, Hanging: Relational Pose Diffusion for Multi-modal Rearrangement. Simeonov et al.

Results

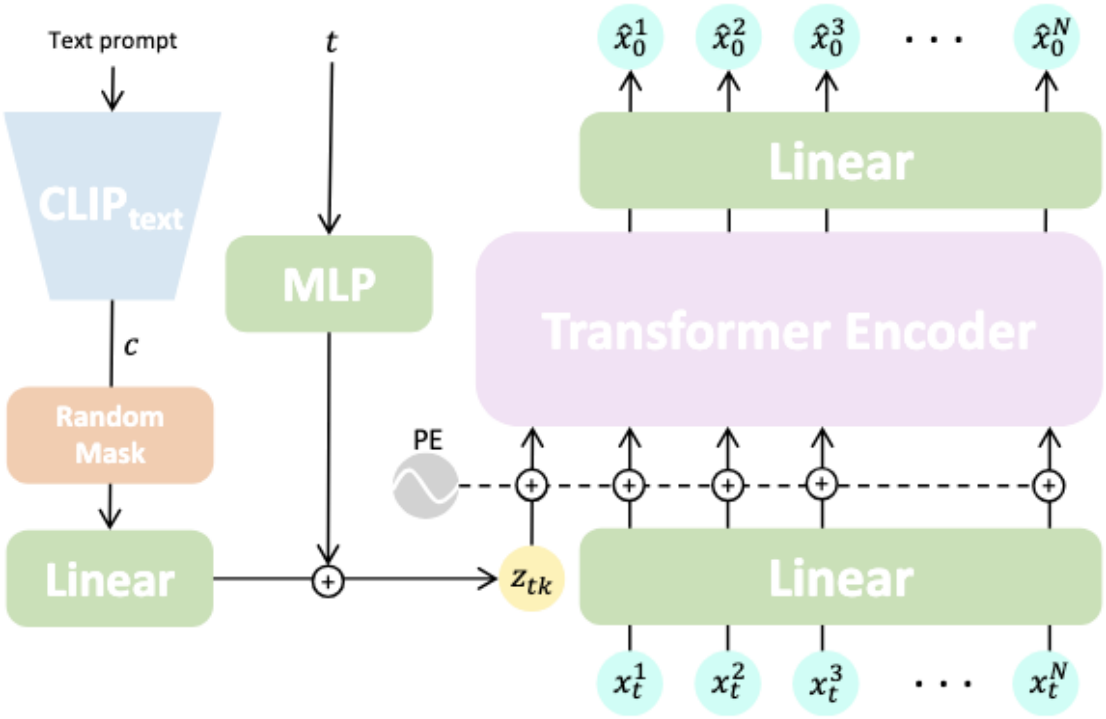
Diffusion Models that Generate Particle Motion



- Idea: Generate human poses $x^{1:N} = (x^1, x^2, x^3, x^4, \ldots x^N)$ represented by either joint rotations or positions

MDM: Human Motion Diffusion Model. Tevet et al.

Diffusion Models that Generate Particle Motion

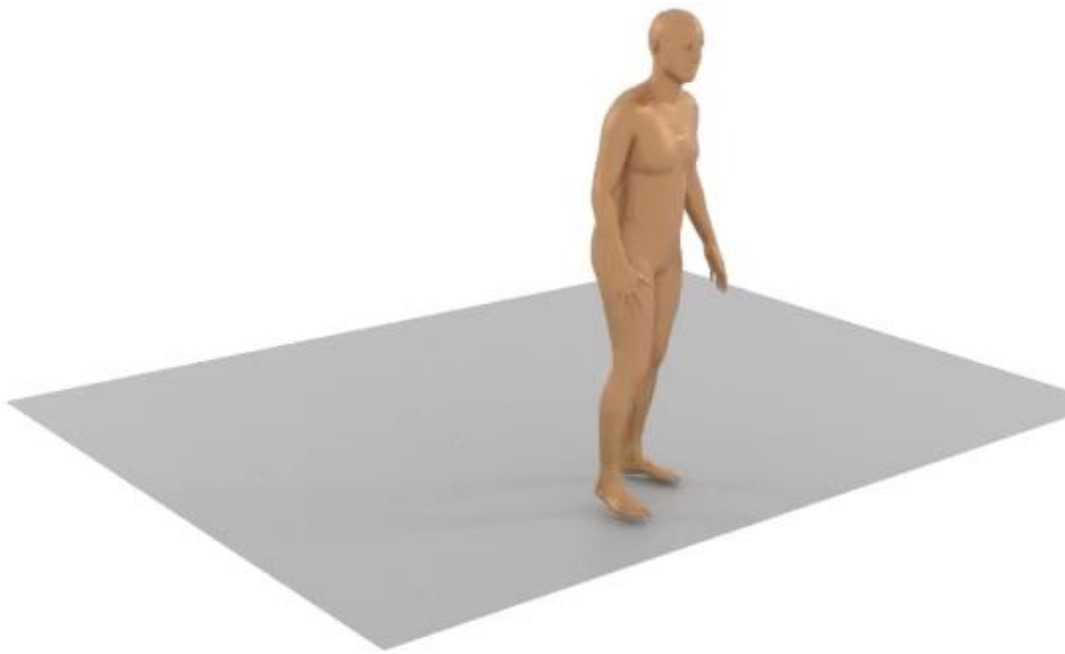


- Idea: Generate human poses $x^{1:N} = (x^1, x^2, x^3, x^4, \ldots x^N)$ represented by either joint rotations or positions

MDM: Human Motion Diffusion Model. Tevet et al.

Results

“a person turns to his right and paces back and forth.”



MDM: Human Motion Diffusion Model. Tevet et al.

Are Our Generated 4D Worlds or Motions Physically Correct?

https://deepmind.google/technologies/veo/veo-2/

What We Will Cover the Next Week

- Physical modeling
  - Rigid-body motion