



16. Generative Models

Generative Model

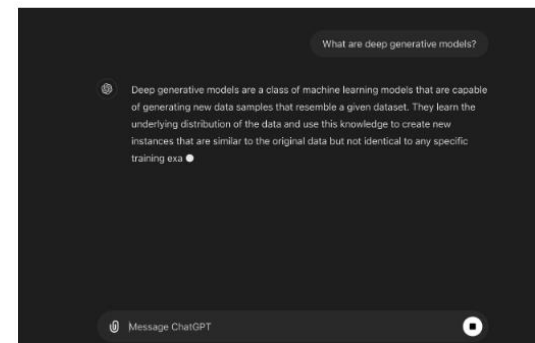
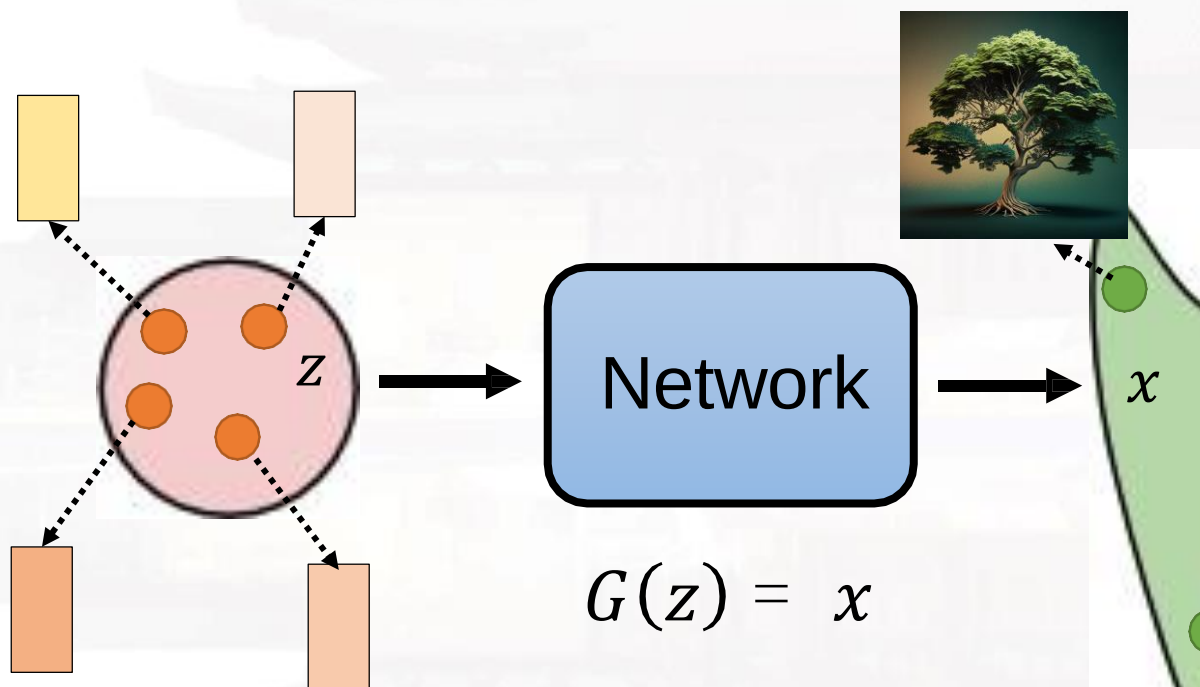


What are deep generative models?



Deep generative models are a class of machine learning models that are capable of generating new data samples that resemble a given dataset. They learn the underlying distribution of the data and use this knowledge to create new instances that are similar to the original data but not identical to any specific training example ●

Generative Models



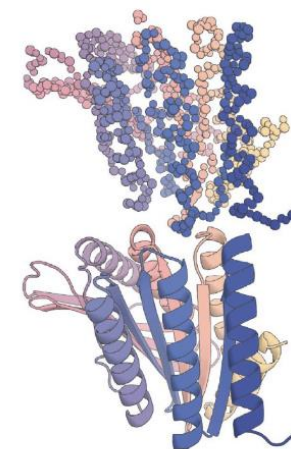
Chatbot



Image generation

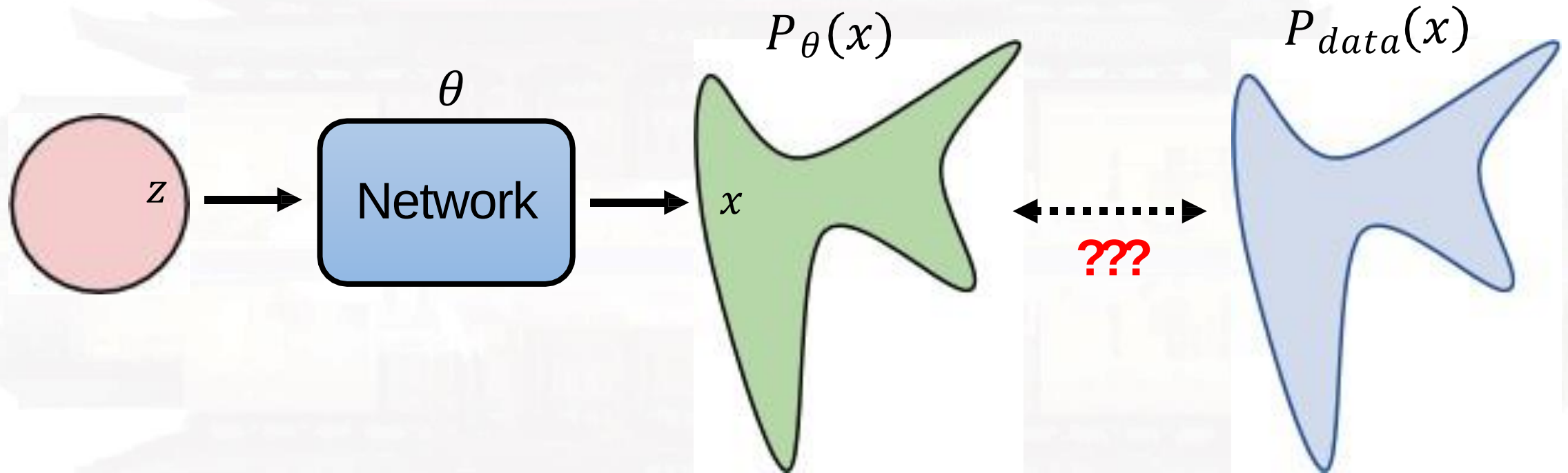


Video generation

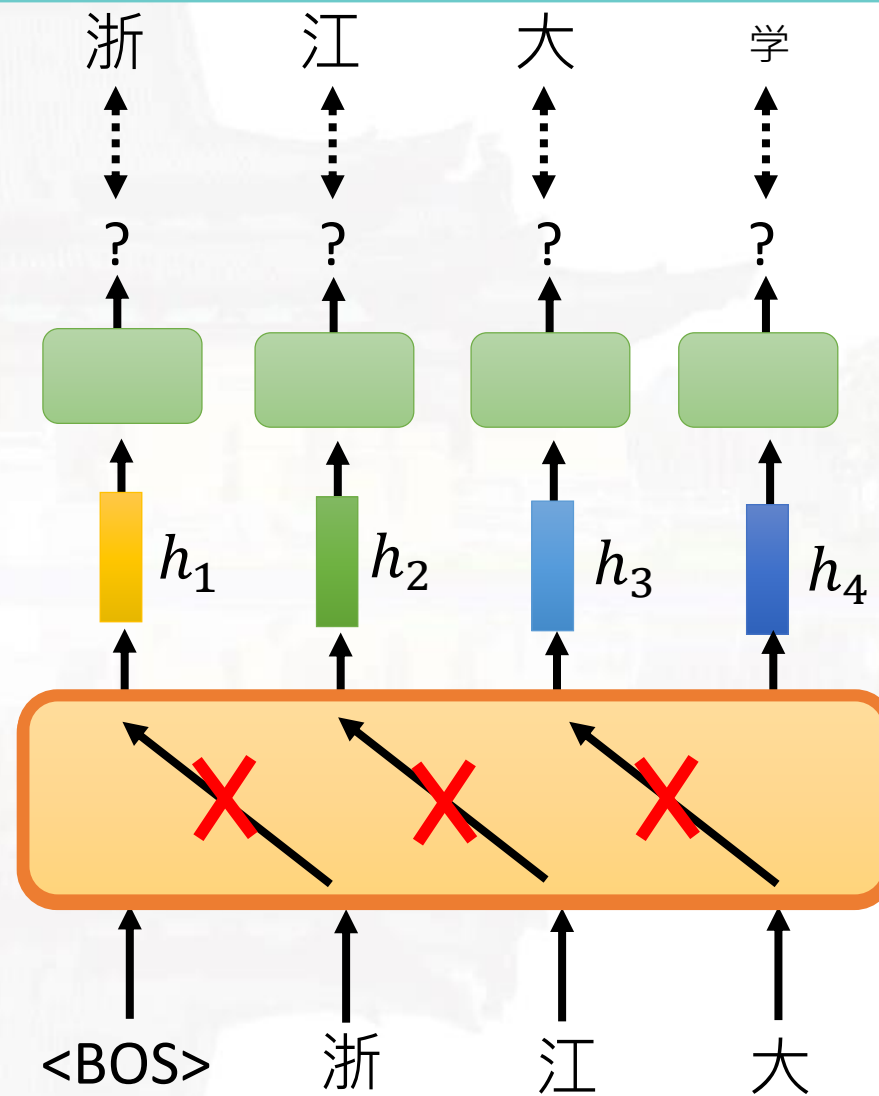
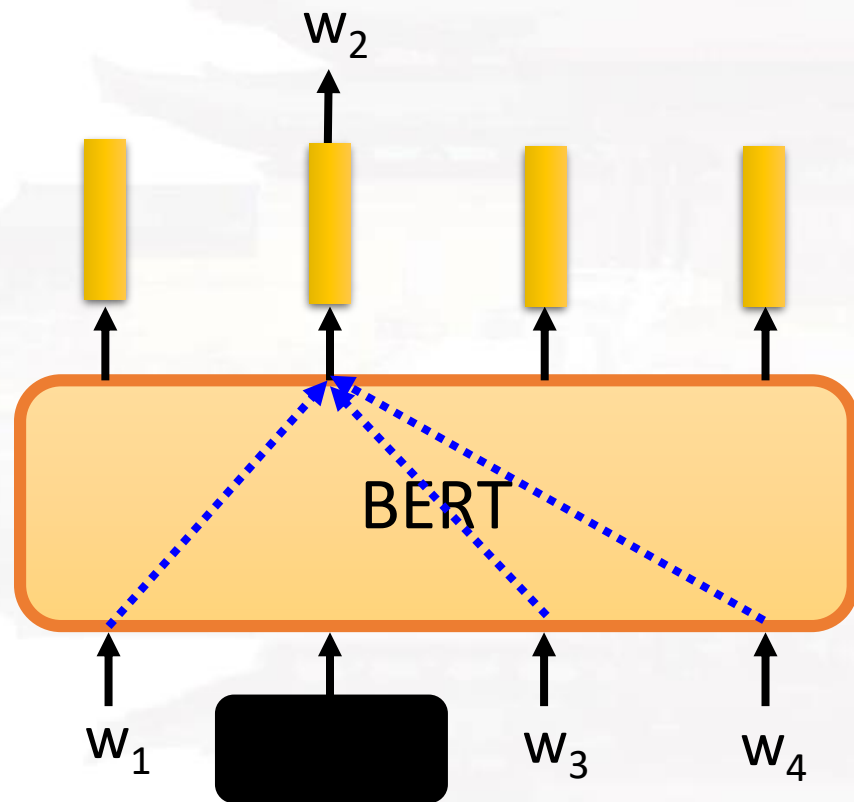
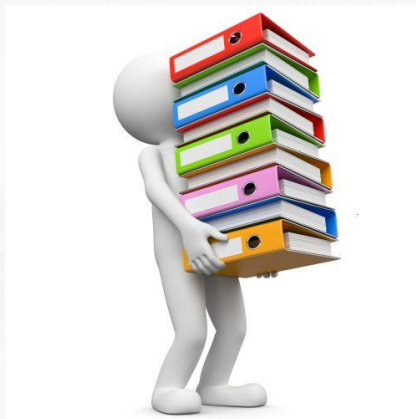


Protein generation

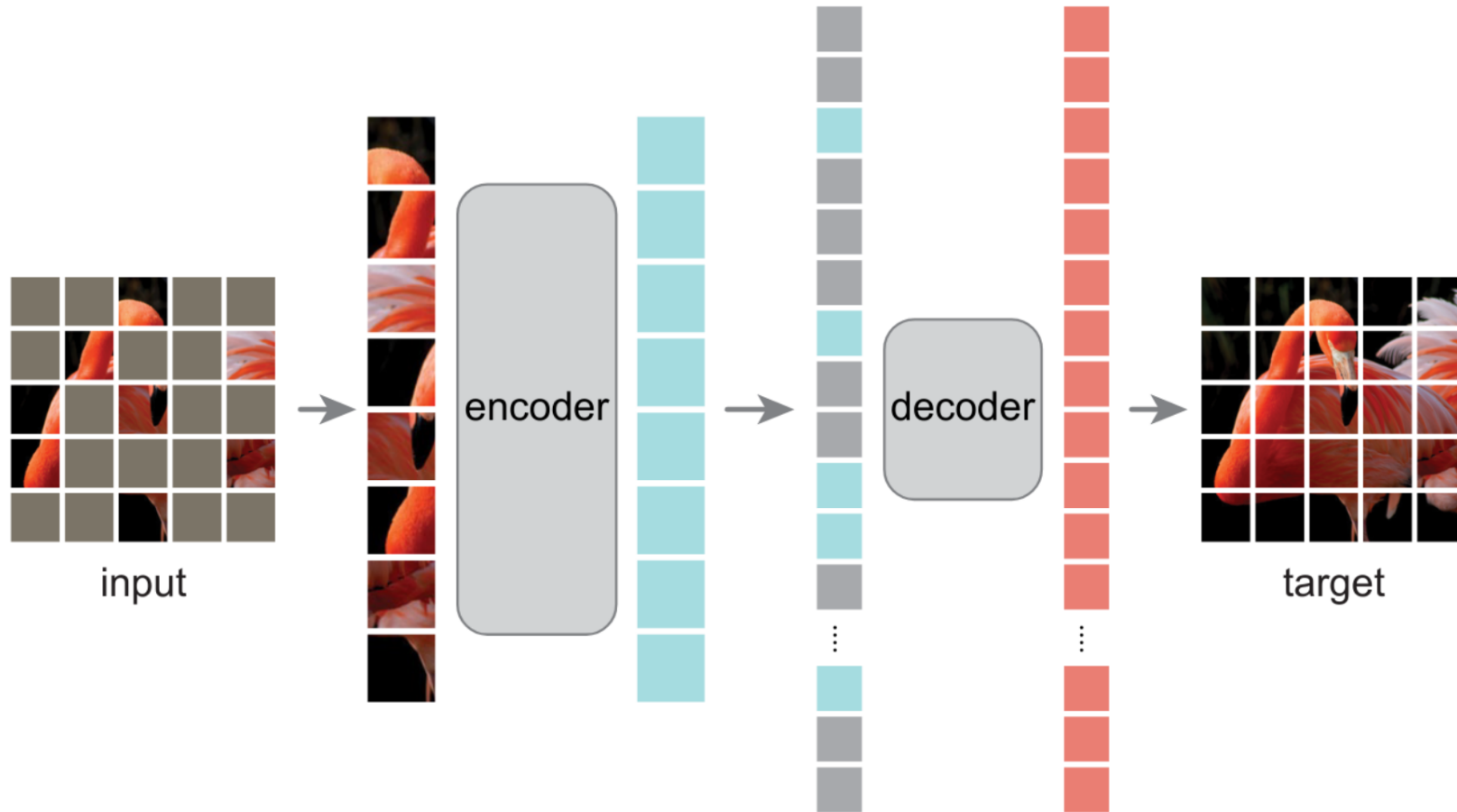
Generative Models



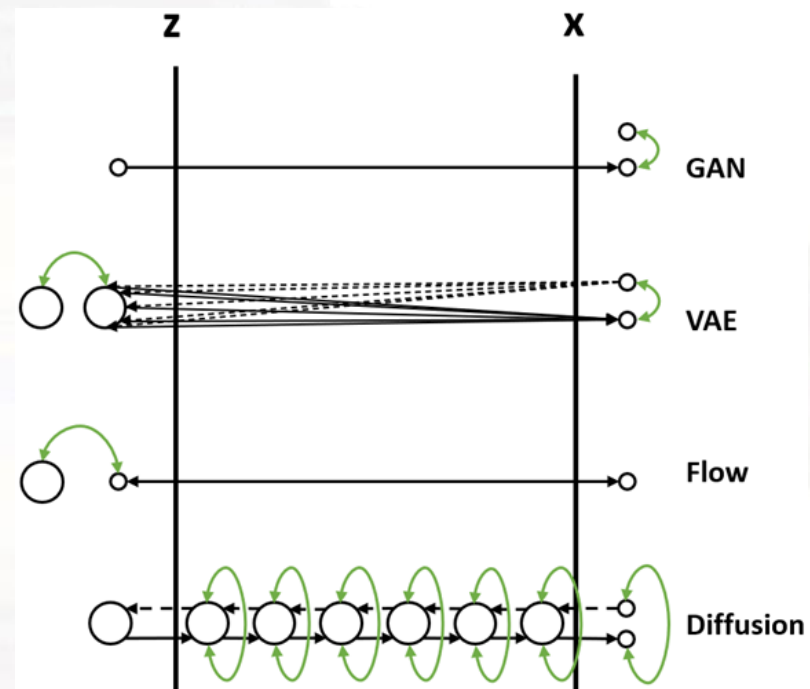
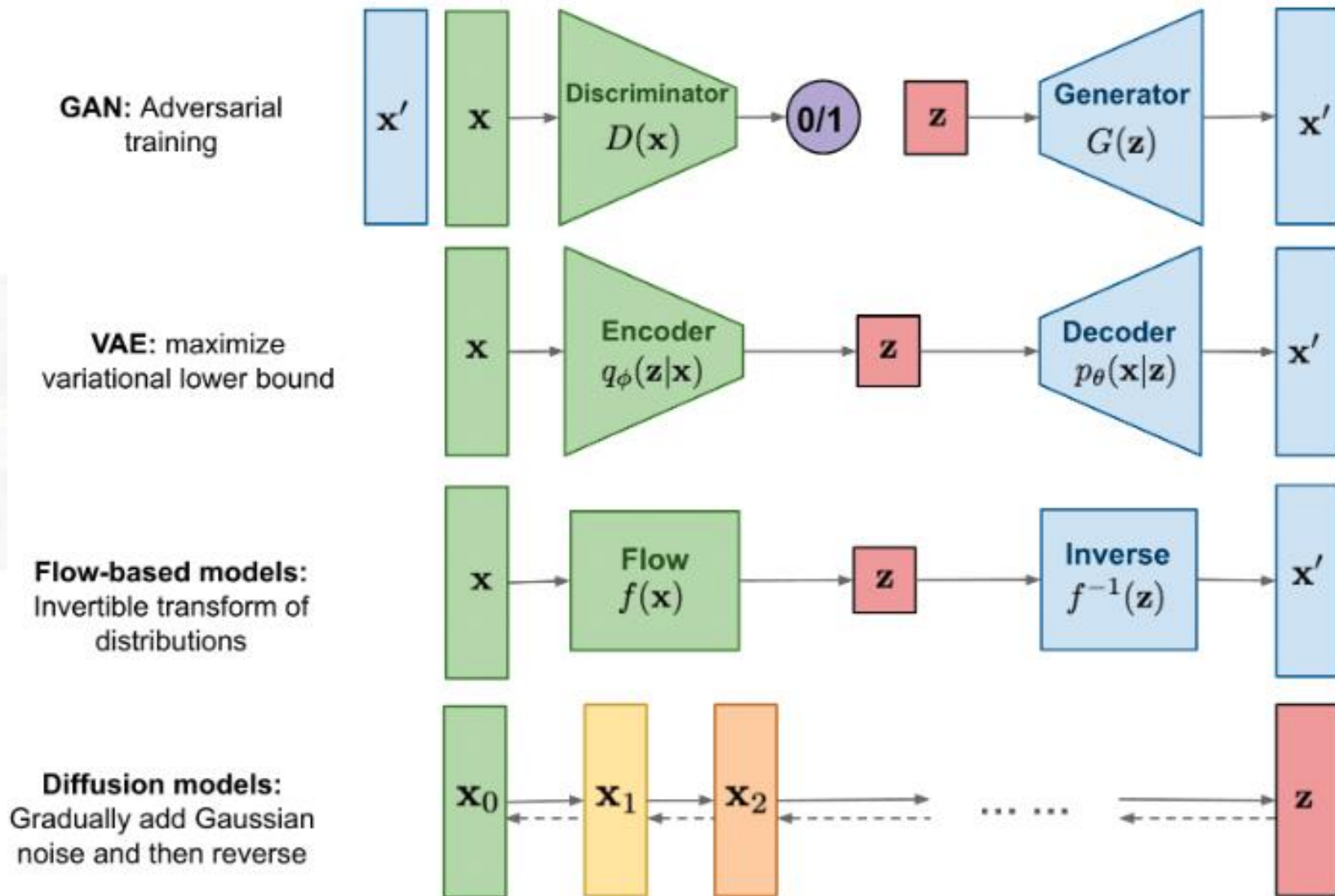
How: Self-supervised learning



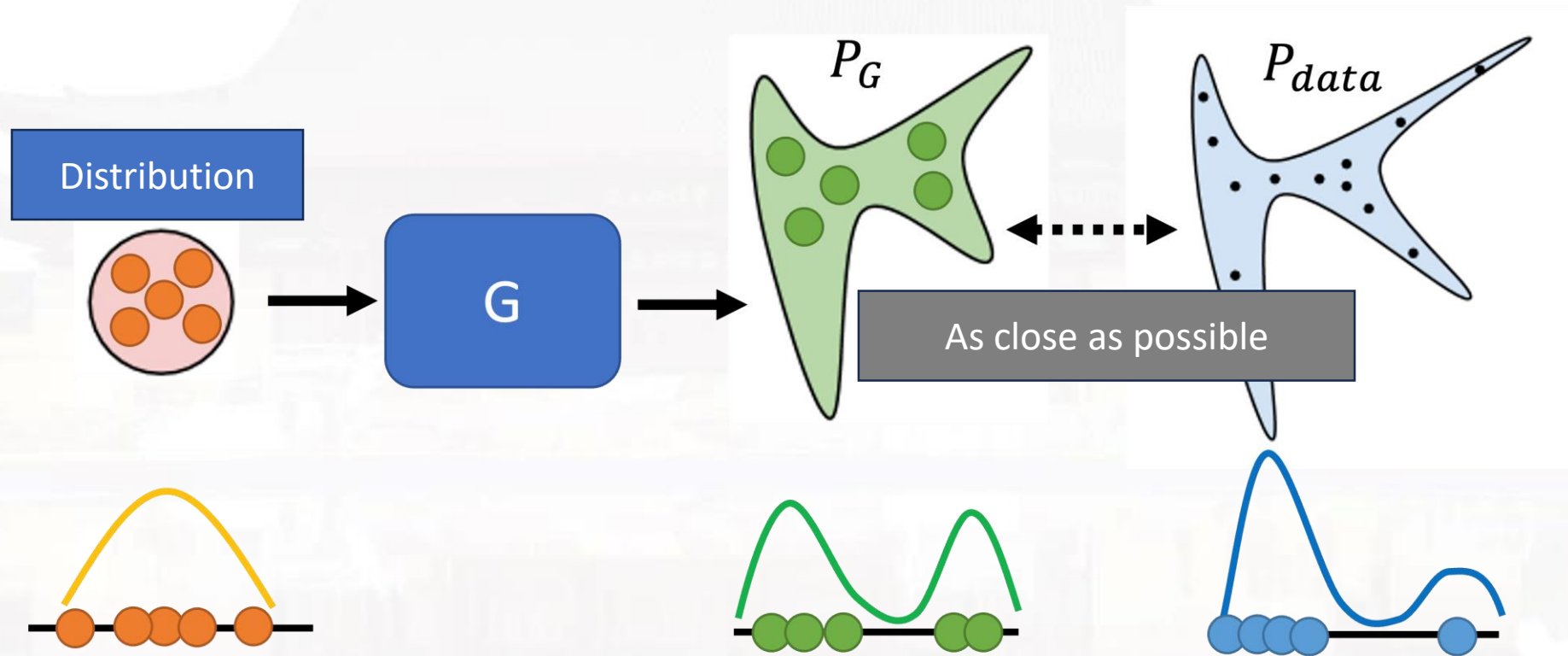
How: Self-supervised learning



Generative Models



GAN



$$G^* = \arg \min_G \underline{Div}(P_G, P_{data})$$

Difference between P_G and P_{data}

GAN



Drag Your GAN: Interactive Point-based Manipulation on the Generative Image Manifold

Xingang Pan^{1,2} Ayush Tewari³ Thomas Leimkühler¹ Lingjie Liu^{1,4} Abhimitra Meka⁵ Christian Theobalt^{1,2}

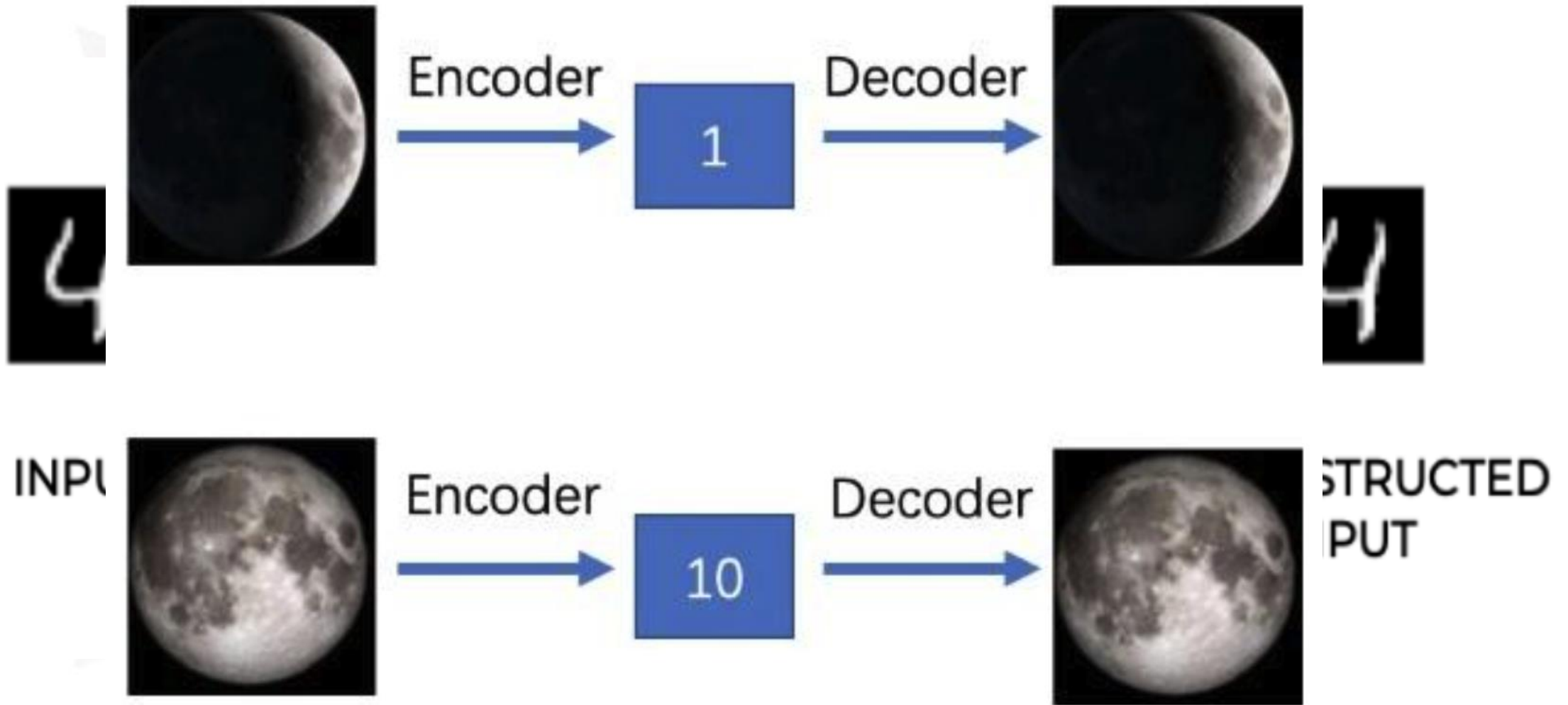
¹Max Planck Institute for Informatics ²Saarbrücken Research Center for Visual Computing, Interaction and AI ³MIT ⁴University of Pennsylvania
⁵Google AR/VR

SIGGRAPH 2023 Conference Proceedings

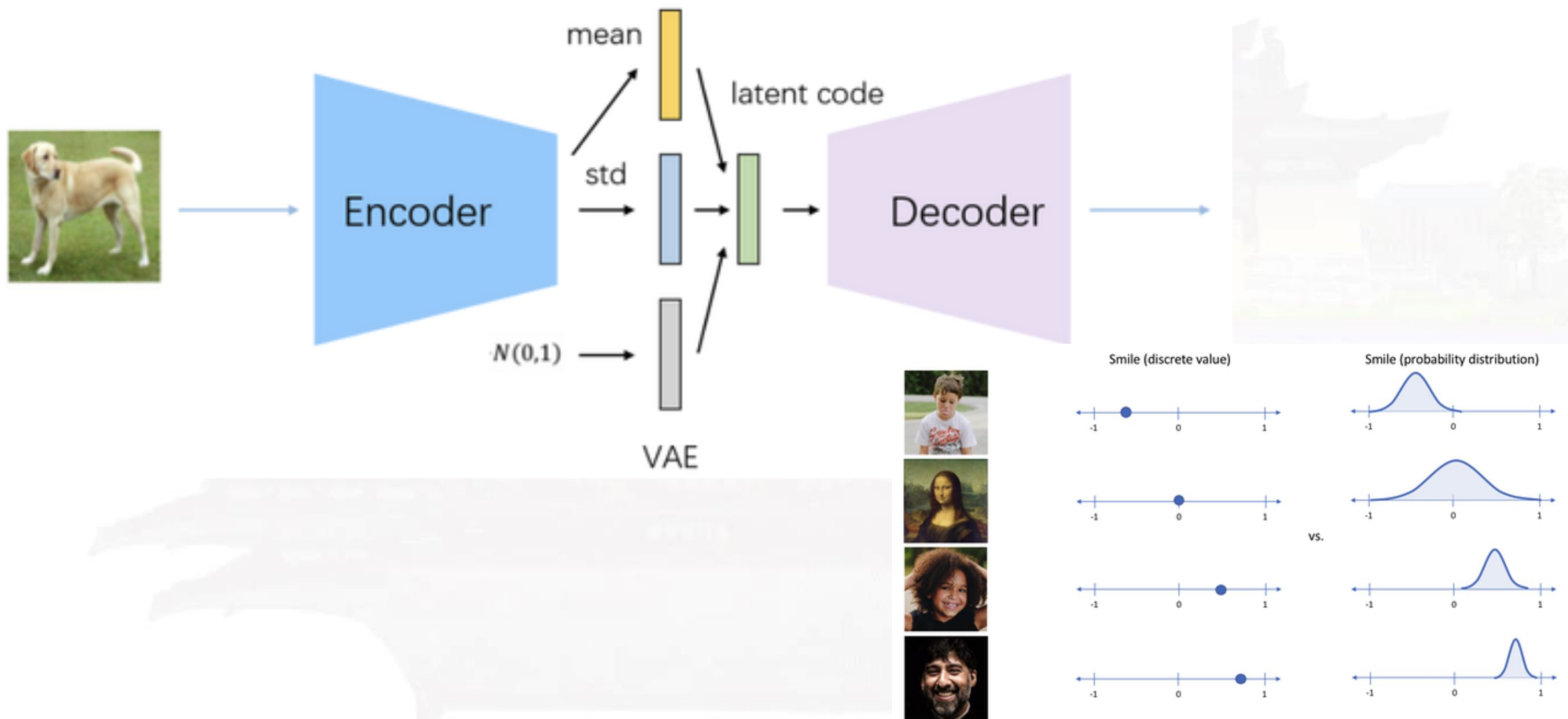
Image + User input (1st Edit) Result 2nd Edit Result

The screenshot shows the website for 'Drag Your GAN'. At the top, there are logos for the Max Planck Institute for Informatics, Saarbrücken Research Center for Visual Computing, MIT, University of Pennsylvania, and Google. The title 'Drag Your GAN: Interactive Point-based Manipulation on the Generative Image Manifold' is prominently displayed. Below the title, the authors' names and affiliations are listed. The main content area shows a sequence of four images: a lion, a lion with a red dot on its nose, a lion with a red dot on its nose and a blue dot on its ear, and a lion with a red dot on its nose. This is followed by a sequence of four images of a cat: a cat with a red dot on its nose, a cat with a red dot on its nose and a blue dot on its ear, a cat with a red dot on its nose and a blue dot on its ear, and a cat with a red dot on its nose. The bottom of the screenshot shows a Windows taskbar with various application icons.

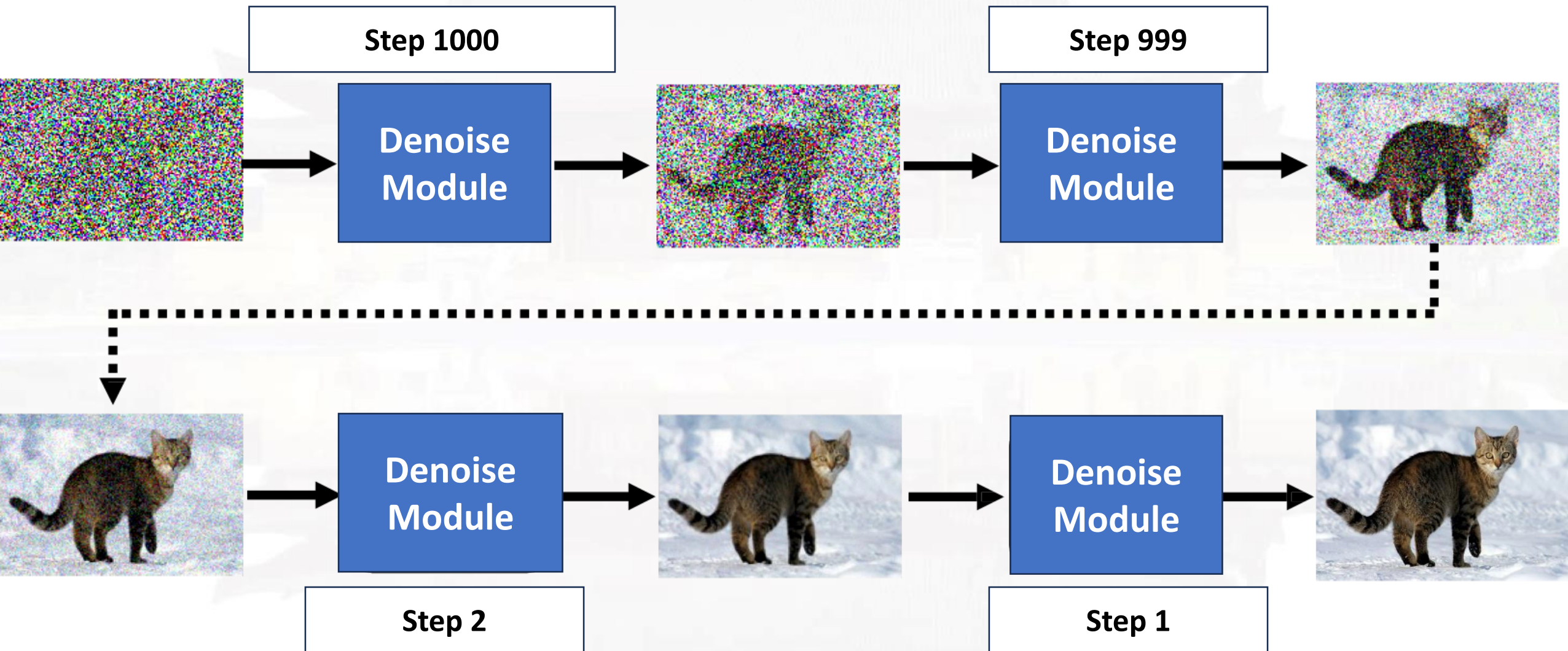
VAE



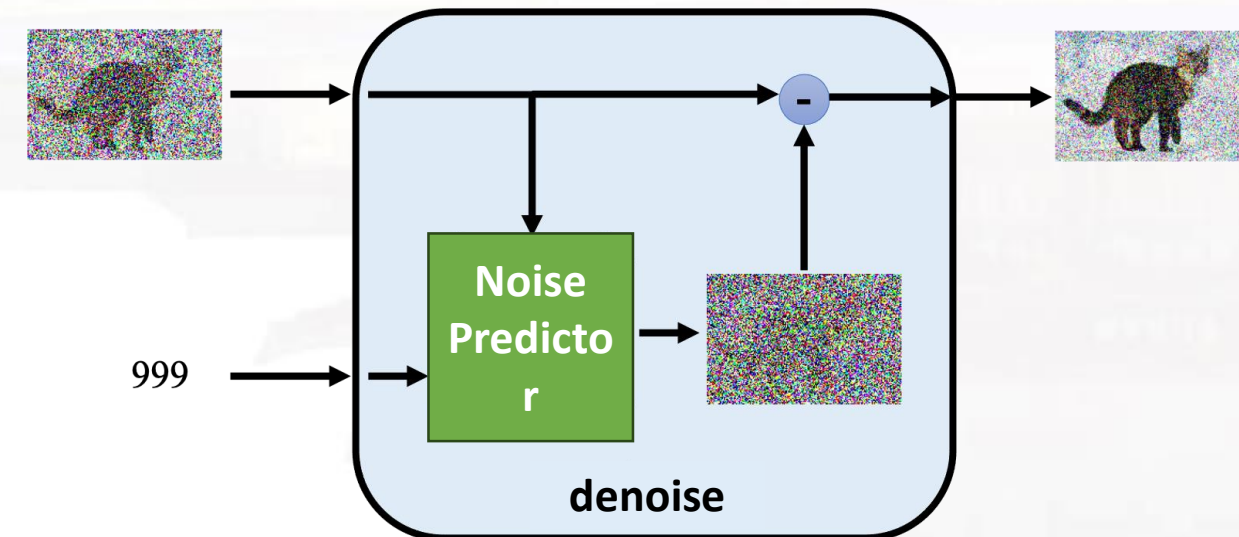
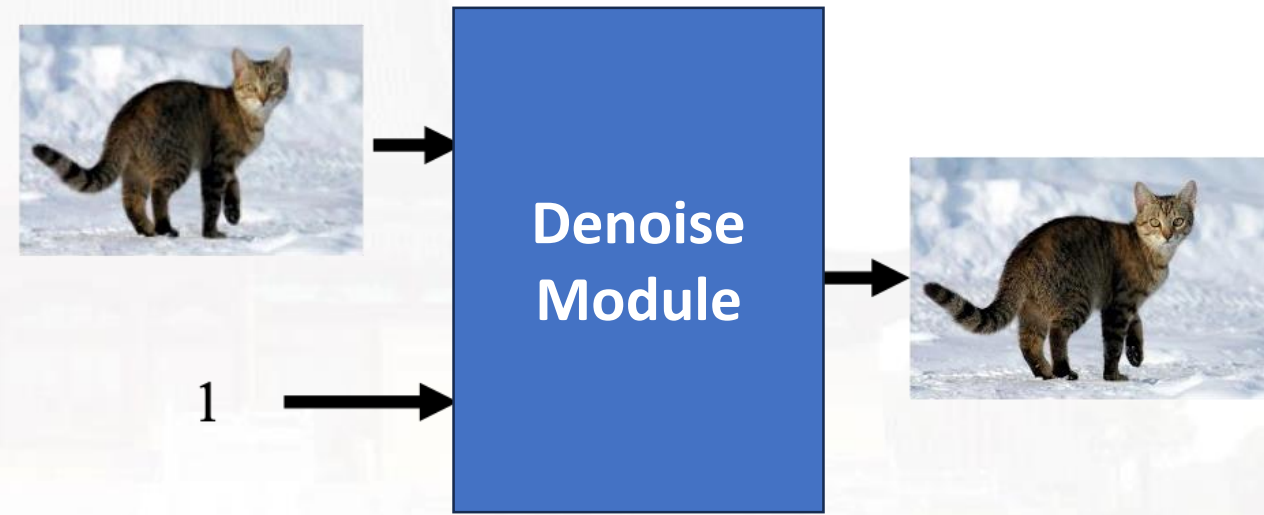
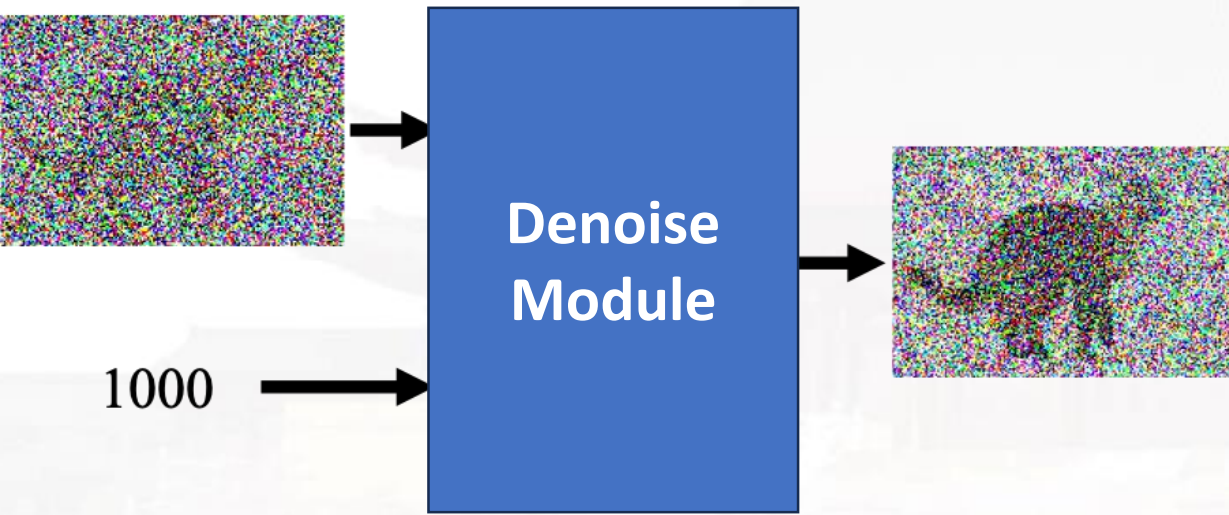
VAE



Diffusion Model



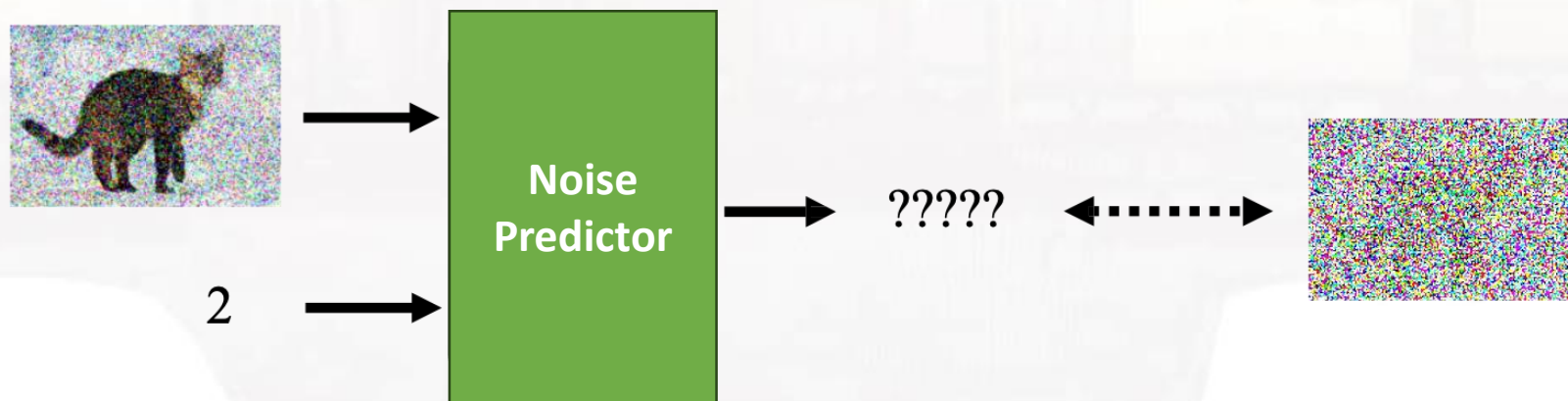
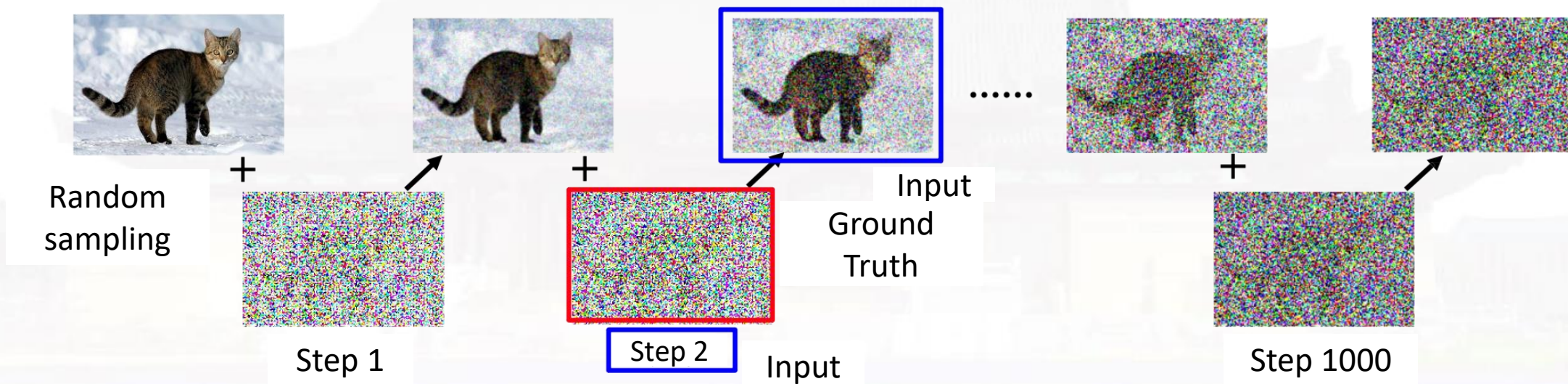
Diffusion Model



Q: Why not directly use an end-to-end model, where the input is the image to be denoised and the output is directly the denoised result?

A: It is possible to do so. However, most current papers still choose to use a noise predictor because the difficulty of generating an image is different from that of generating noise. If a denoising model can generate a noisy cat, it can almost be said that it already knows how to draw a cat. Therefore, the difficulty of generating a noisy cat is different from generating the noise contained within an image. Thus, directly using a noise predictor might be relatively simpler, whereas using an end-to-end model to directly produce the denoised result is relatively difficult.

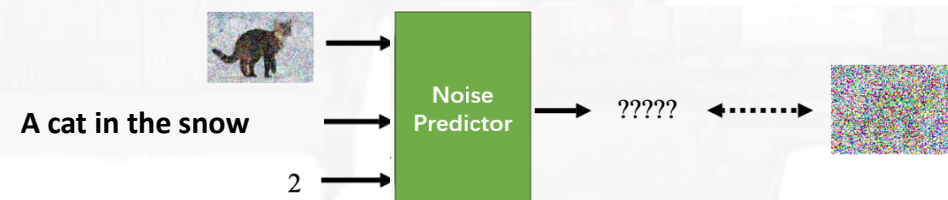
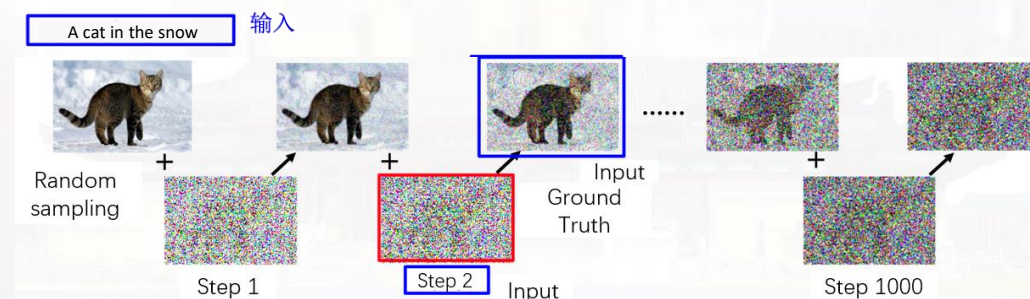
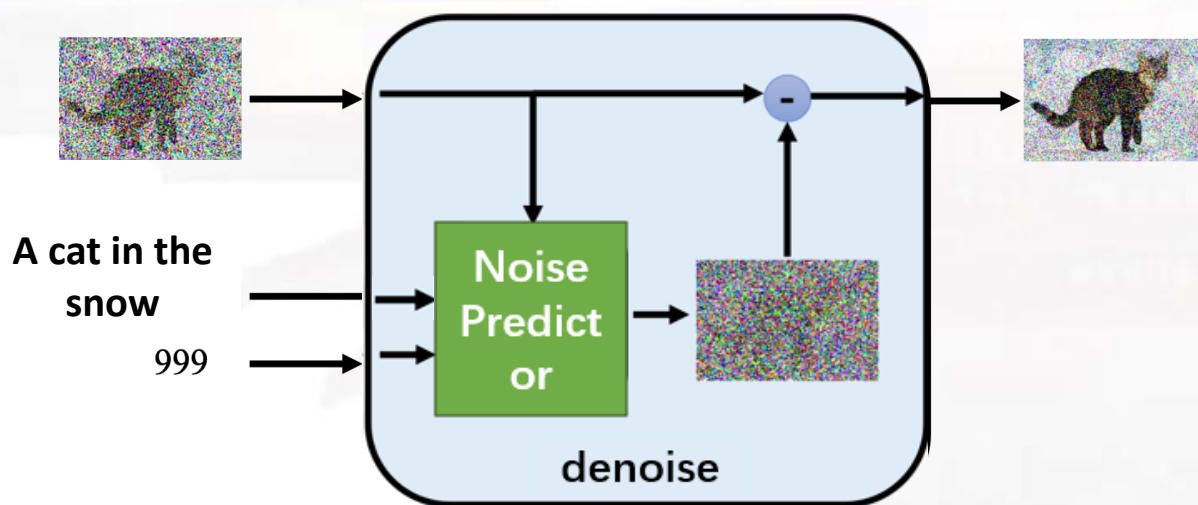
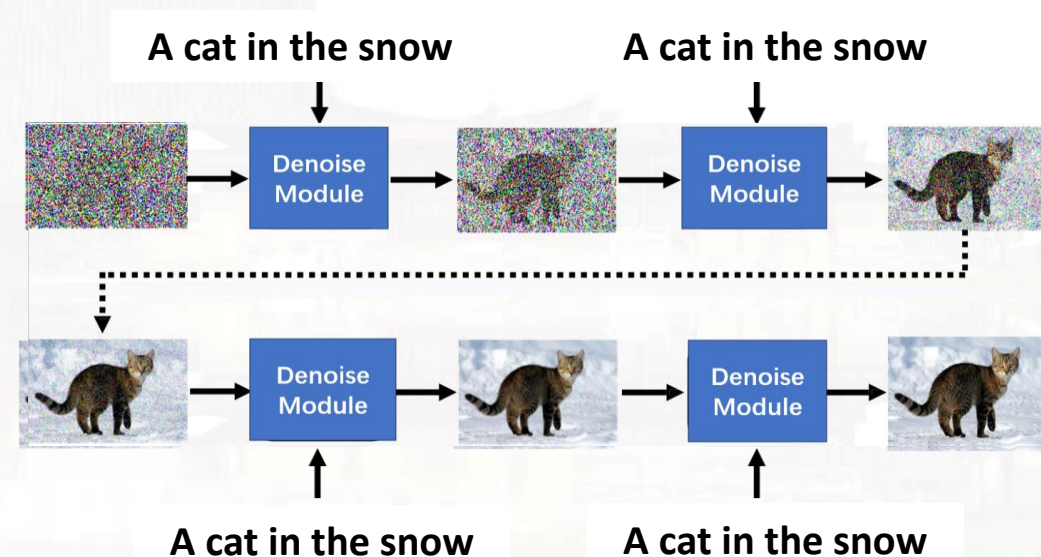
Diffusion Model



Diffusion Model



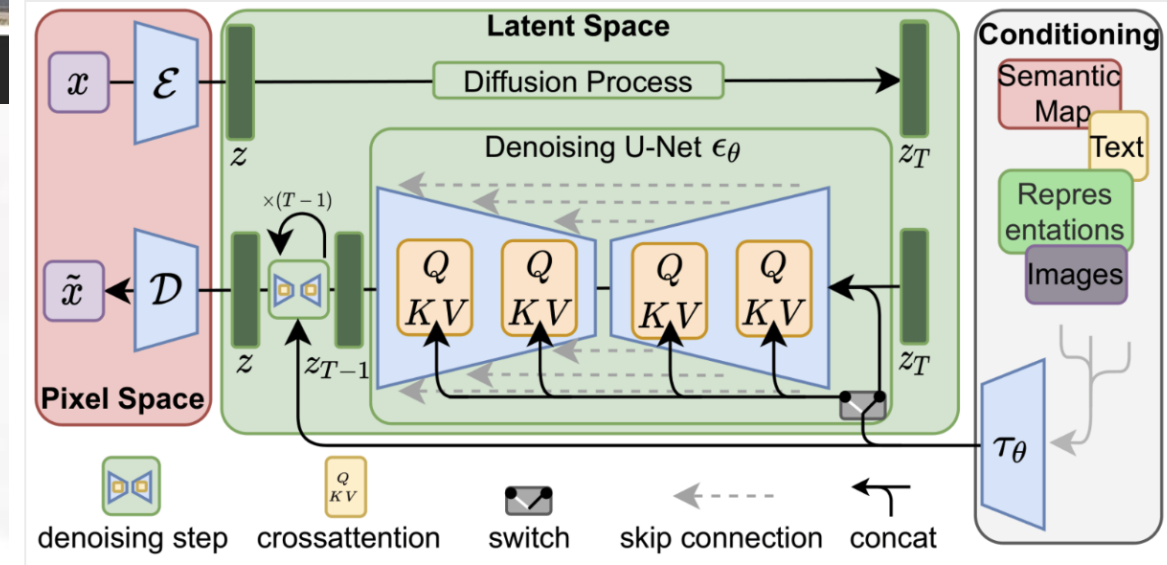
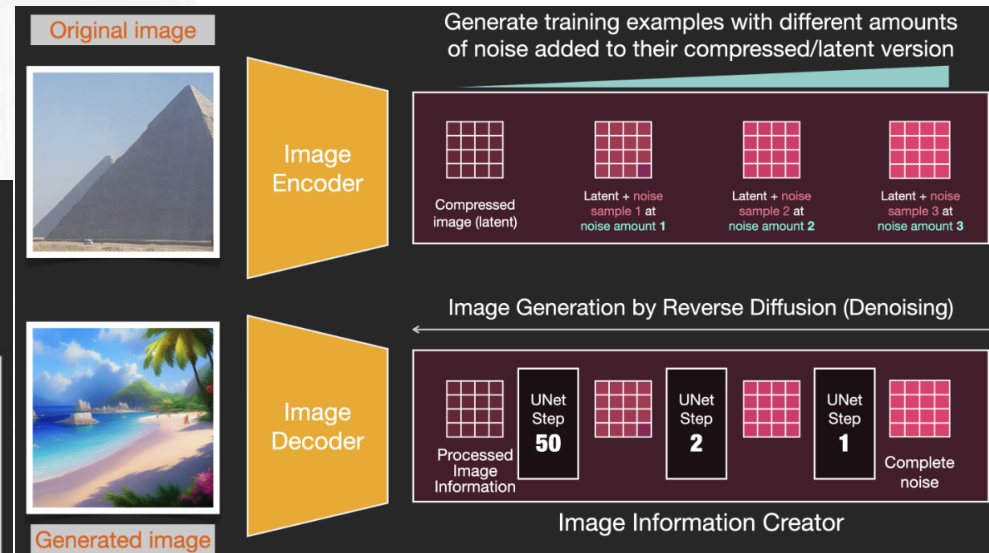
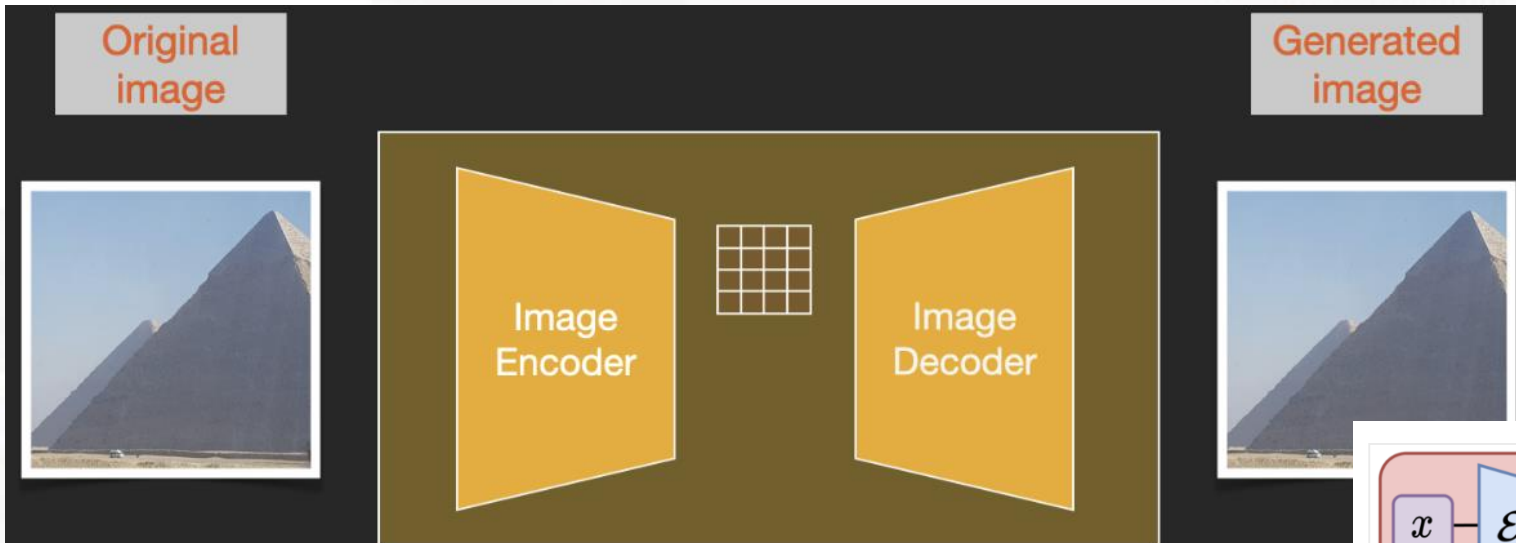
■ Text to Image



Diffusion Model



■ LDM





DIFFUSION EXPLAINER

Visual Explanation for Text-to-image Stable Diffusion

Seongmin Lee, Ben Hoover, Hendrik Strobelt, Zijie J. Wang, Anthony Peng,
Austin Wright, Kevin Li, Haekyu Park, Alex Yang, Polo Chau

Diffusion Model

■ ControlNet



Input Canny edge



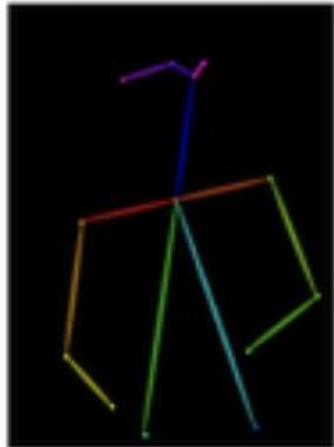
Default



"masterpiece of fairy tale, giant deer, golden antlers"



"..., quaint city Galie"



Input human pose



Default



"chef in kitchen"

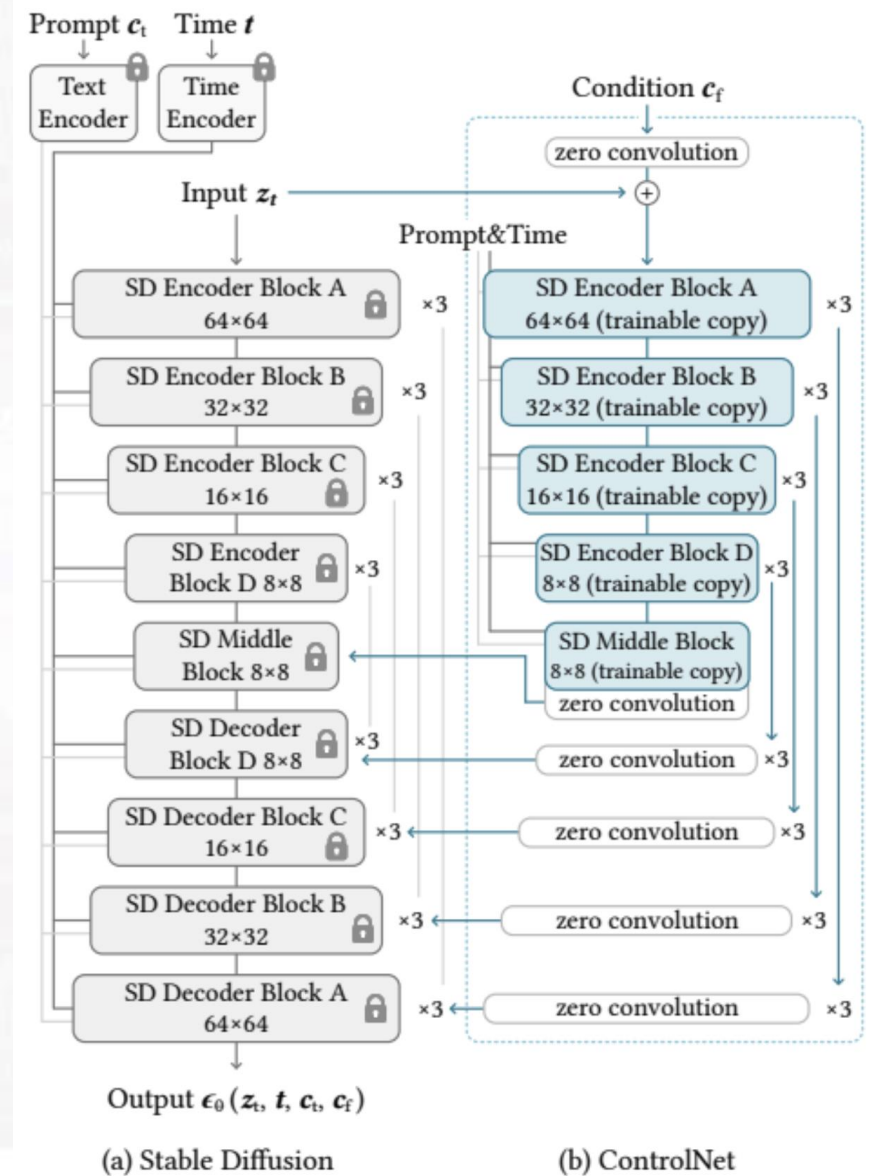
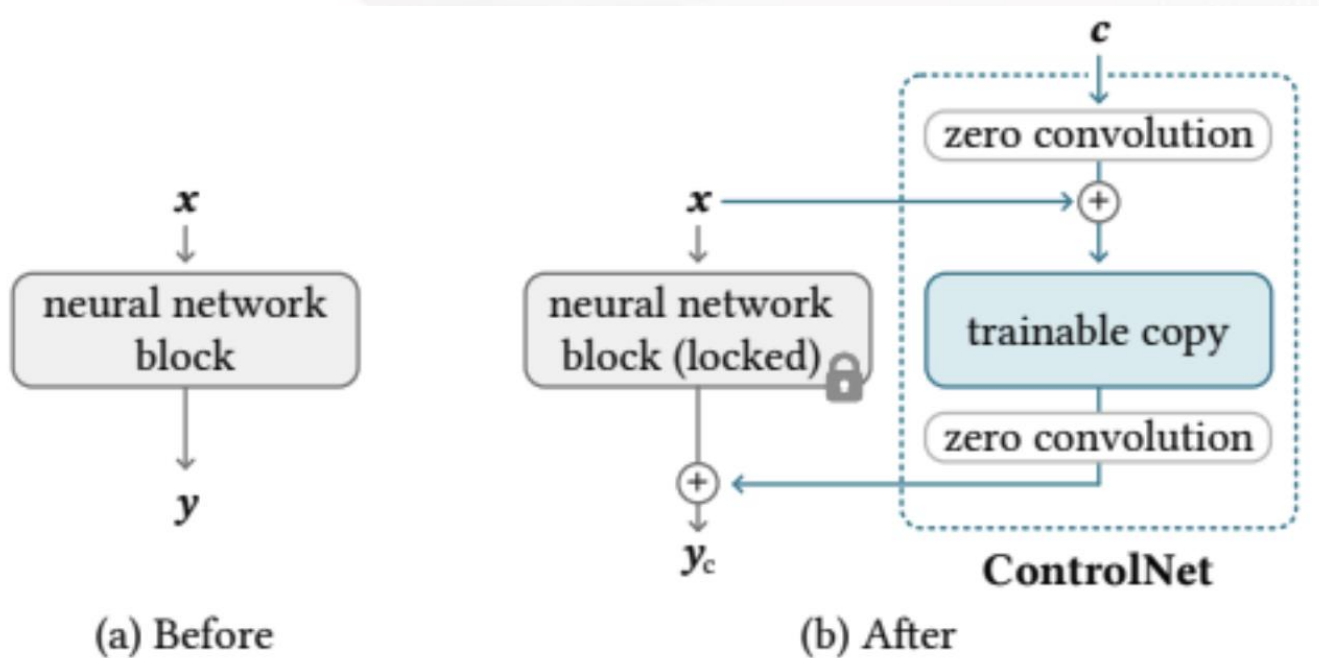


"Lincoln statue"

Diffusion Model



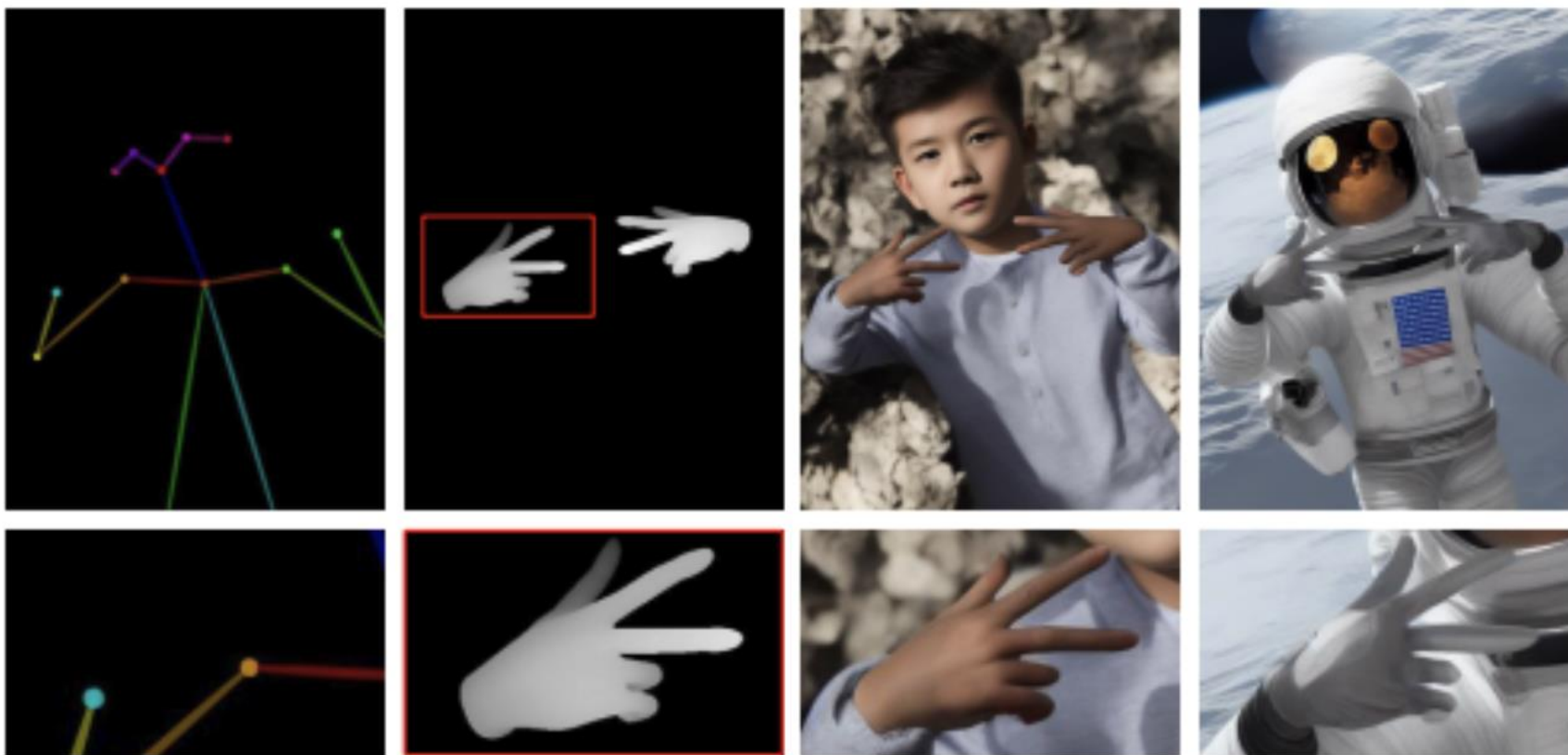
■ ControlNet



Diffusion Model



■ ControlNet



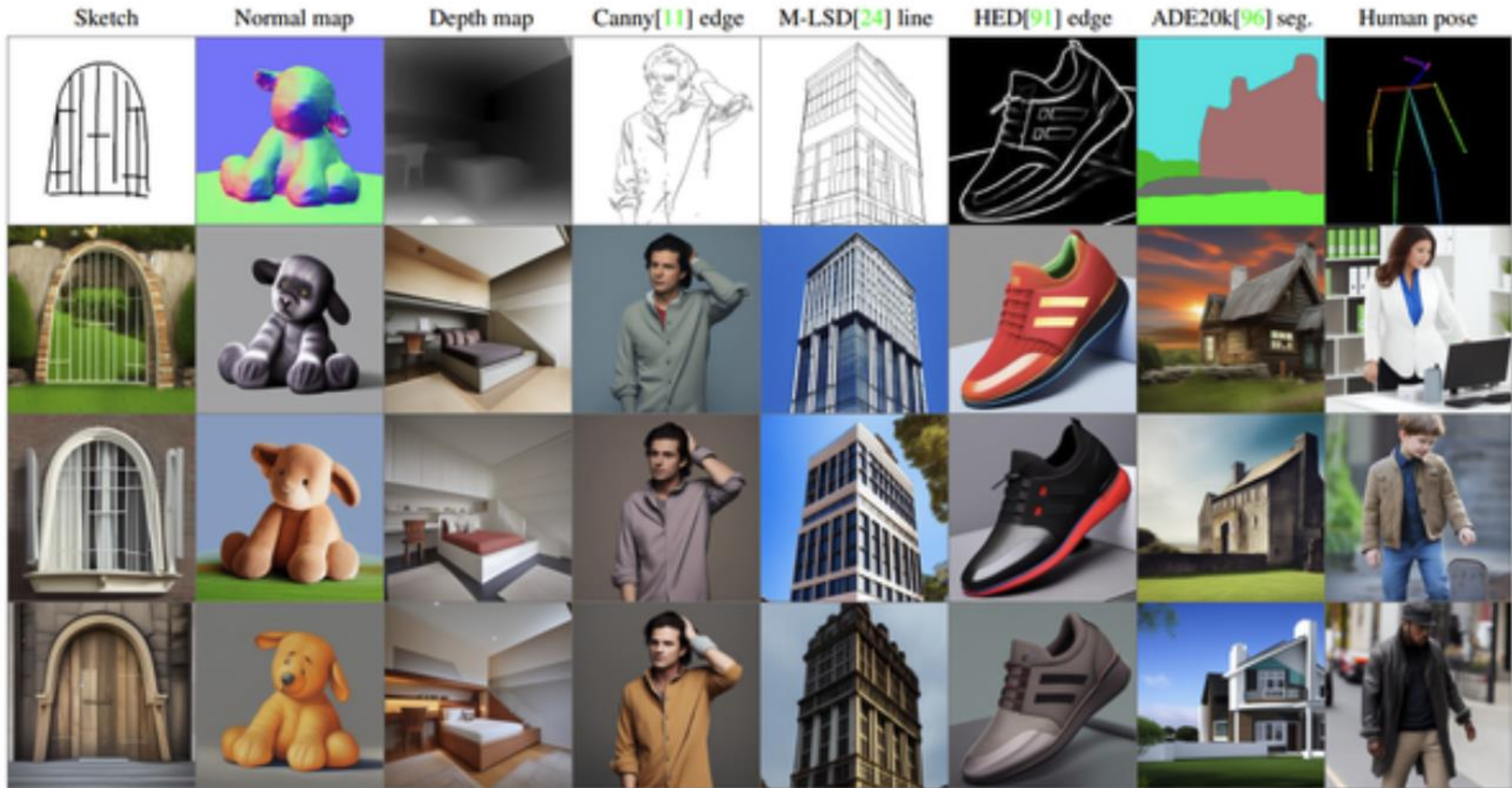
Multiple condition (pose&depth)

“boy”

“astronaut”

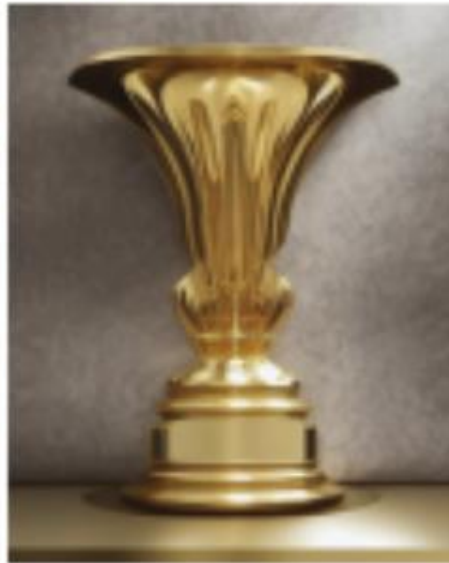
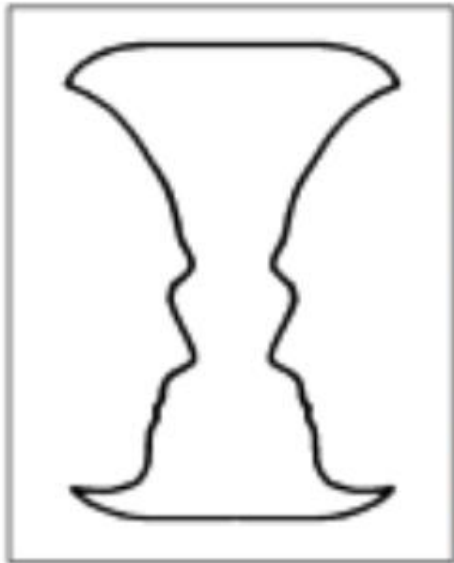
Diffusion Model

■ ControlNet



Diffusion Model

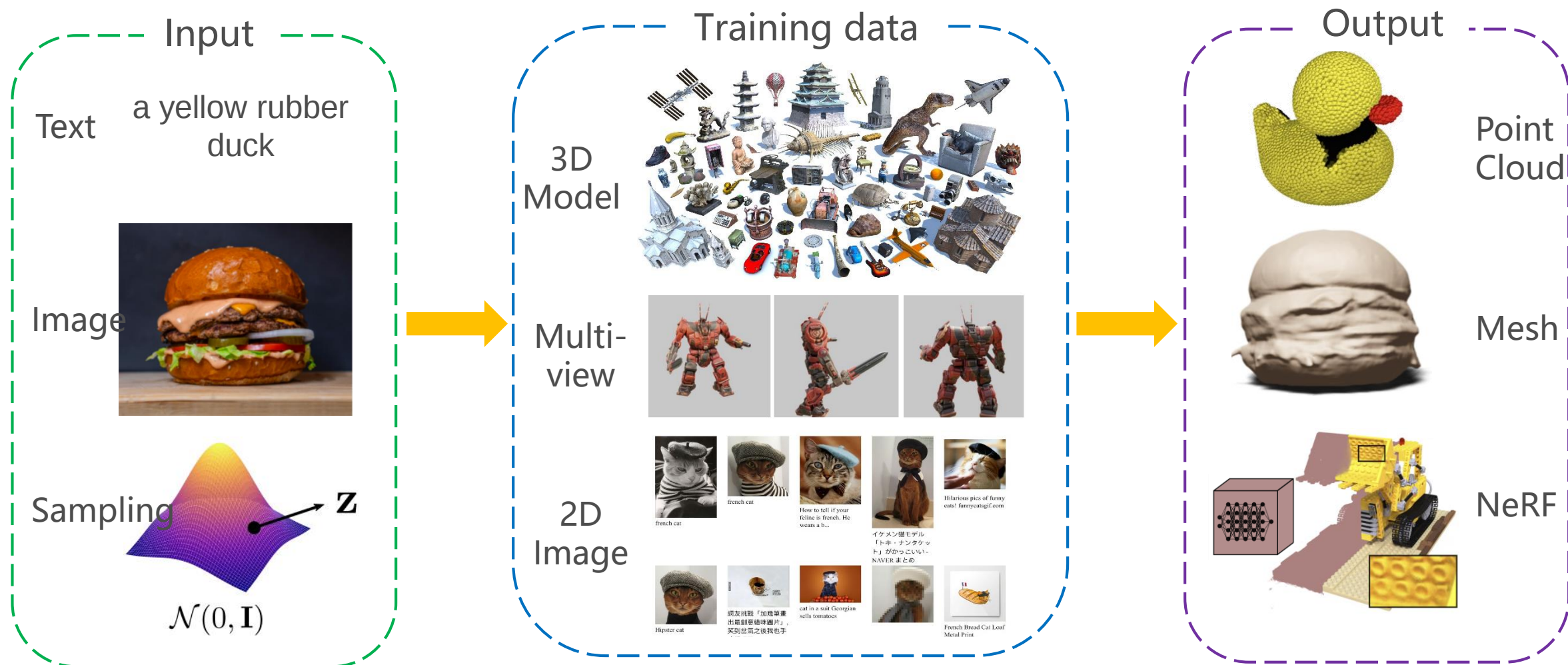
■ ControlNet



Input

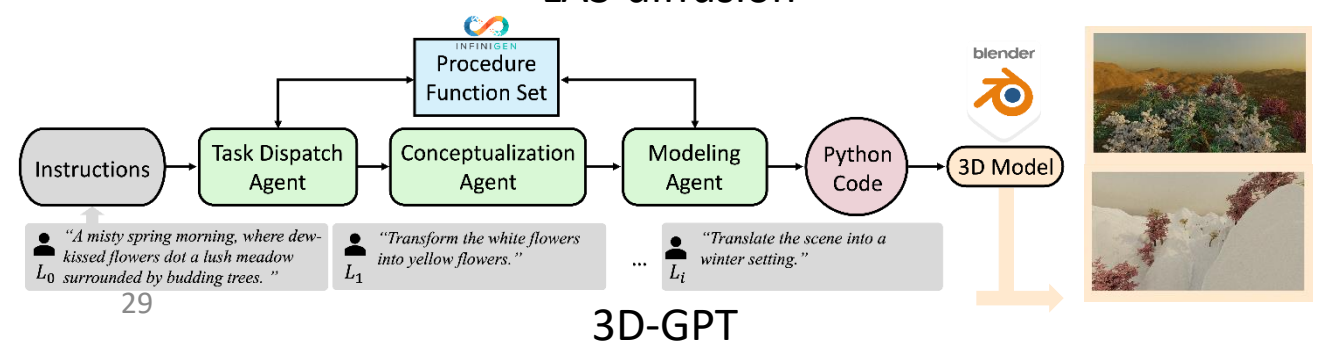
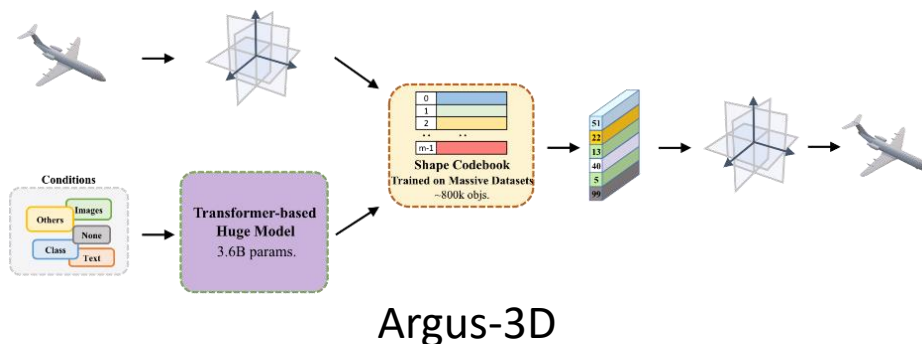
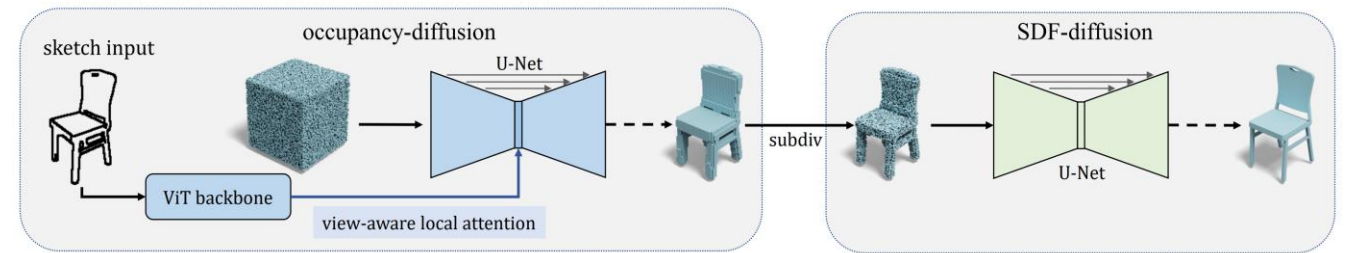
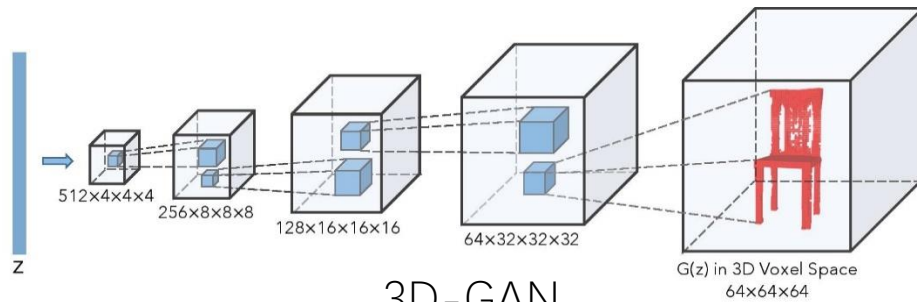
“a high-quality and extremely detailed image”

3D AIGC



3D AIGC

- 3D generation from 3D data
 - **GAN based:** 3D-GAN, Get3D
 - **Diffusion + Volume:** Rodin, LAS-diffusion...
 - **Autoregression:** Argus-3D, PolyGen
 - **LLM + procedural modeling:** 3D-GPT



3D AIGC

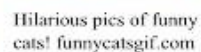
- 3D generation from 3D data



french cat



french cat



Hipster cat



French Bread Cat Loaf
Metal Print



LAION-5B: 5.85 Billion of Images with Text

Objaverse-XL: 10 Million of 3D Objects

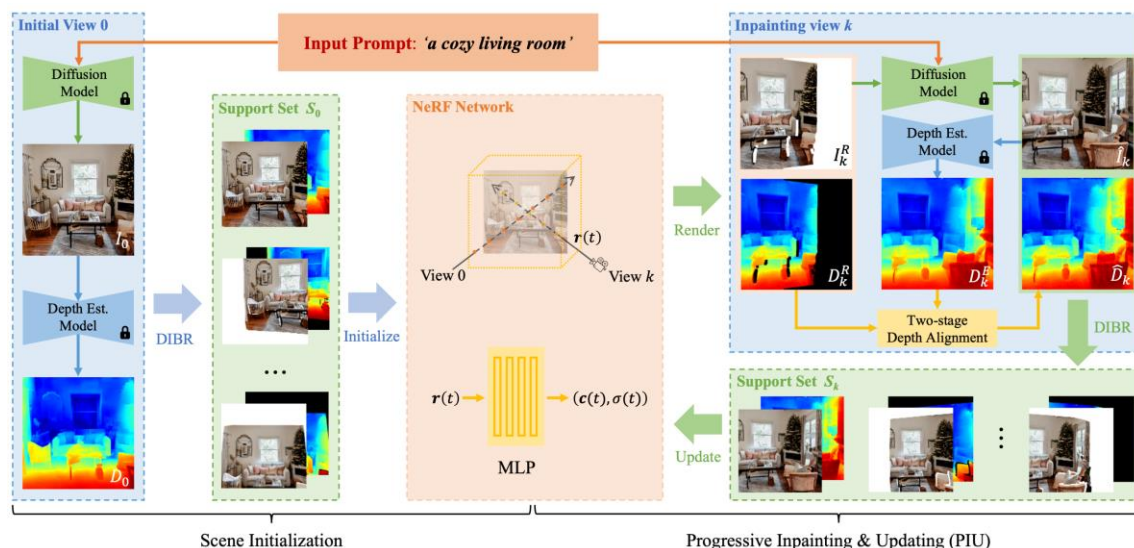
3D AIGC

- 3D Generation from 2D Data
 - **CLIP-guided 3D optimization:** PureCLIPNeRF, CLIP-mesh, Dream3D...
 - **Diffusion-based 3D optimization:** DreamFusion, Magic3D, Fantasia3D, ProlificDreamer...

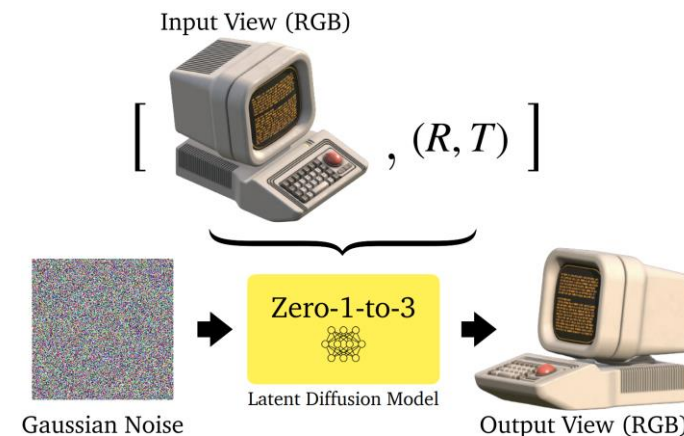


3D AIGC

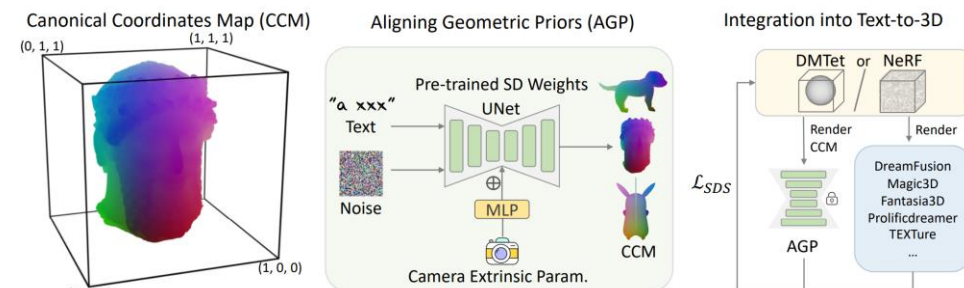
- 3D generation from hybrid data
 - **Progressive Generation:** Text2NeRF, MVDiffusion...
 - **Viewpoint-conditioned Diffusion:** Zero-1-2-3, MVDreamer, SyncDreamer, Wonder3D...
 - **3D-enhanced 2D Diffusion:** SweetDreamer



Text2NeRF



Zero-1-2-3



SweetDreamer