CS580

Introduction to Diffusion Models

Jumin Lee

Advisor : Sung-Eui Yoon

SGVR Lab
KAIST

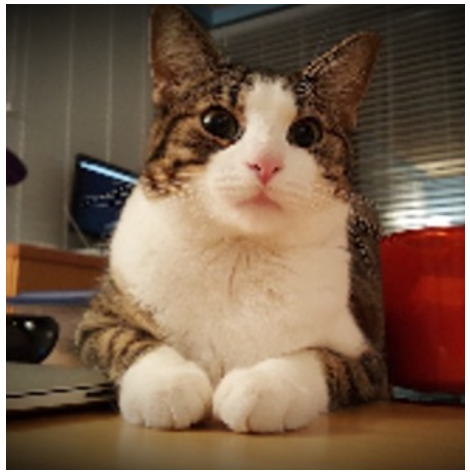
Diffusion Model

2020.06 DDPM

2022.04 DALLE2

2022.05 Imagen

2022.06 Stable Diffusion



Diffusion Model for Conditional Generation

2020.06 DDPM

2022.04 DALLE2

2022.05 Imagen

2022.06 Stable Diffusion

- Conditional Generation
  - **Inpainting**
  - Outpainting
  - Image to Image Generation
  - Text to Image Generation

Diffusion Model for Conditional Generation

2020.06 DDPM

2022.04 DALLE2

2022.05 Imagen

2022.06 Stable Diffusion

- Conditional Generation
  - Inpainting
  - **Outpainting**
  - Image to Image Generation
  - Text to Image Generation

Diffusion Model for Conditional Generation

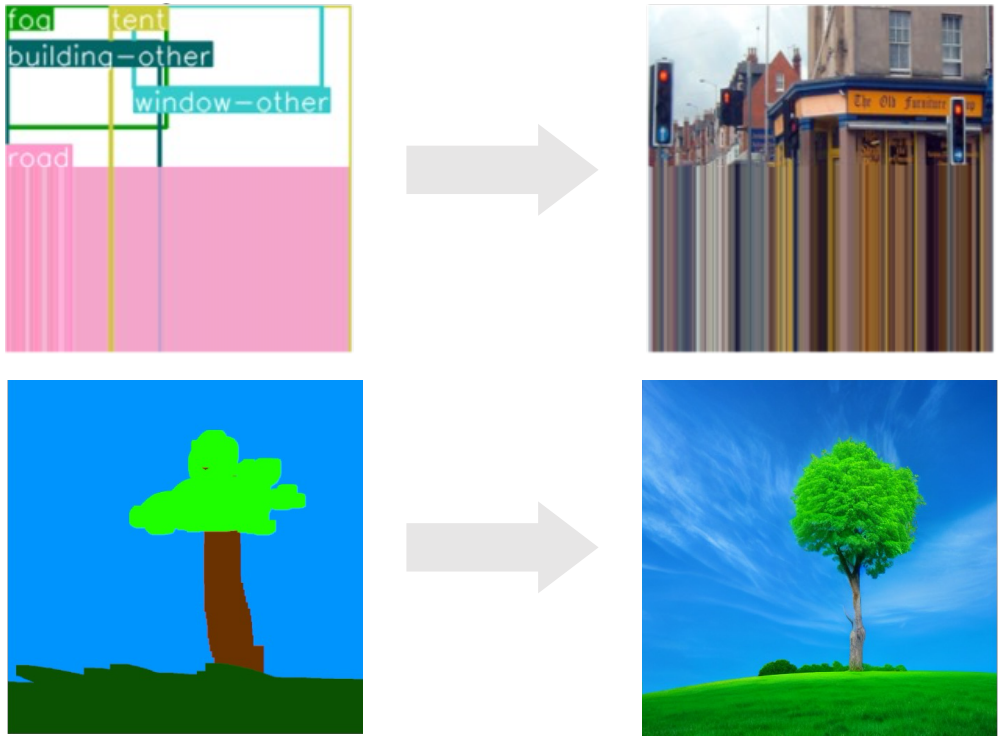
2020.06 DDPM

2022.04 DALLE2

2022.05 Imagen

2022.06 Stable Diffusion

- Conditional Generation
  - Inpainting
  - Outpainting
  - **Image to Image Generation**
  - Text to Image Generation



Diffusion Model for Conditional Generation

2020.06 DDPM

2022.04 DALLE2

2022.05 Imagen

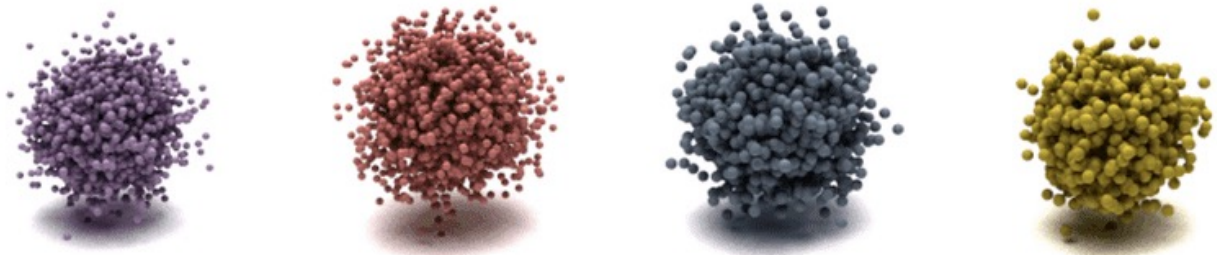
2022.06 Stable Diffusion

- Conditional Generation
  - Inpainting
  - Outpainting
  - Image to Image Generation
  - **Text to Image Generation**

Diffusion Model

2021~
3D Diffusion

- A 3D diffusion process can be used to generate an object from point clouds, meshes, or latent spaces.



2021 PVD

2021
Text2Mesh

2023
Dreamfusion

2023
Magic3D

2023
ProlificDreamer

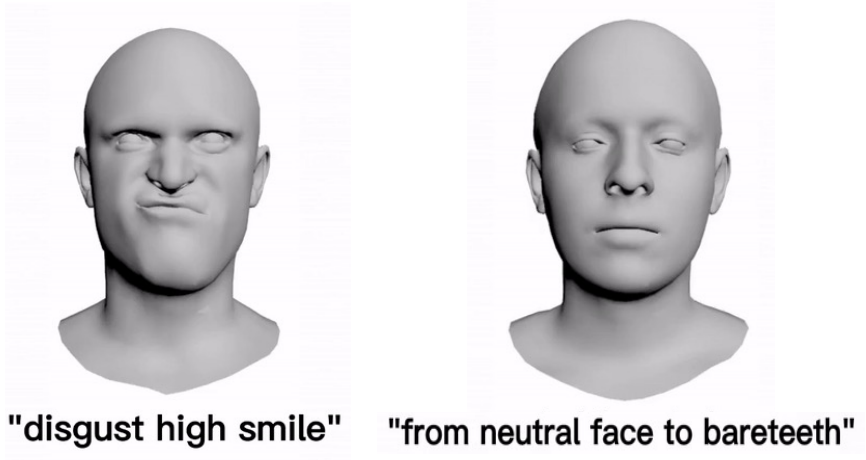
2023
MVdream

Diffusion Model

2021~
3D Diffusion

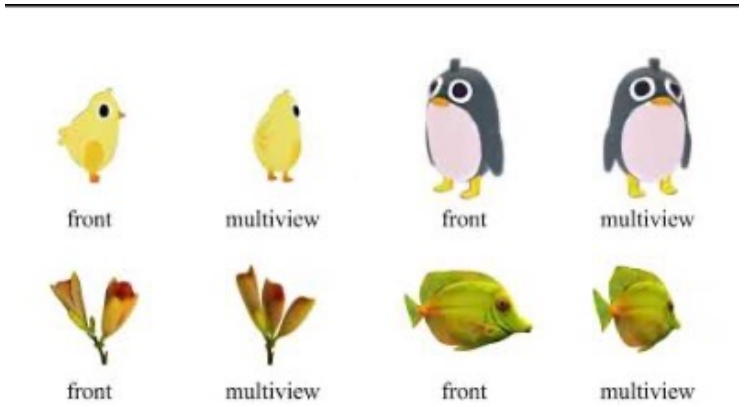
2023~
4D Diffusion

- Extend the diffusion process domain to 4D, including space and time.



2023
4D Facial Expression

2023
Align Your Gaussian

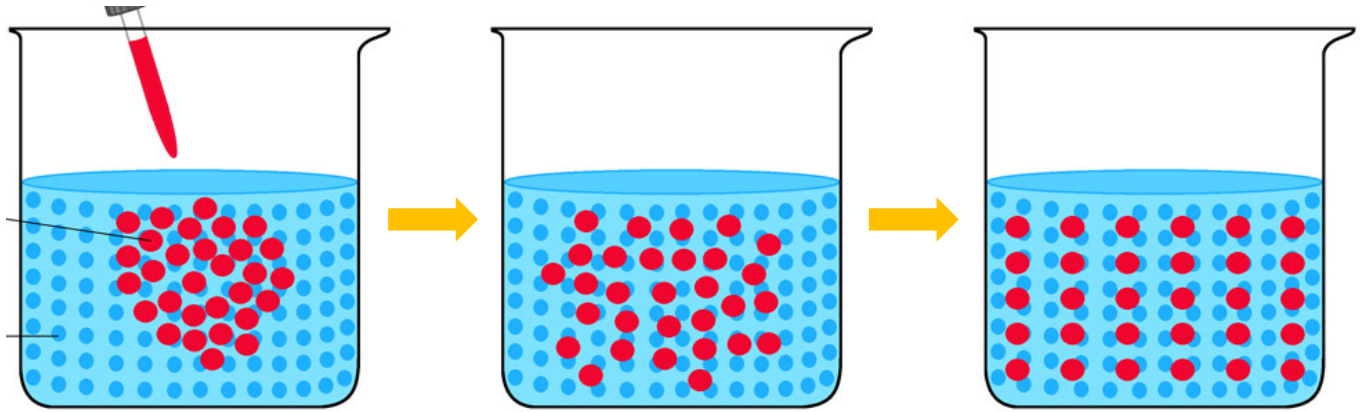


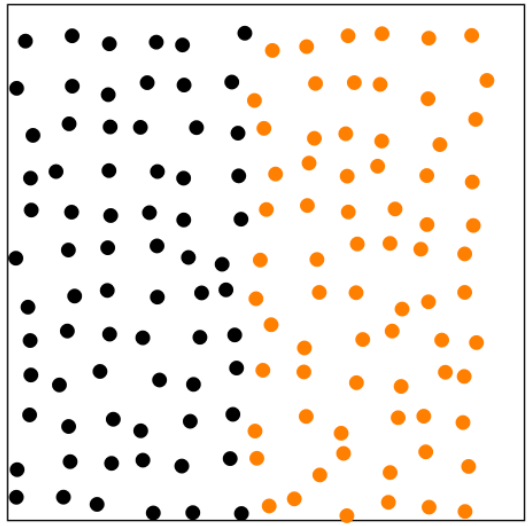
2023
4DGen

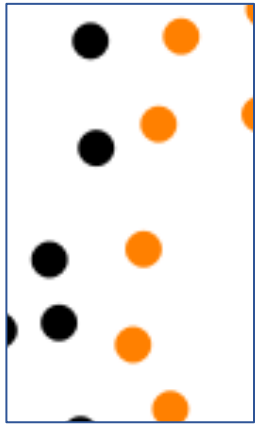
Background

Diffusion Process

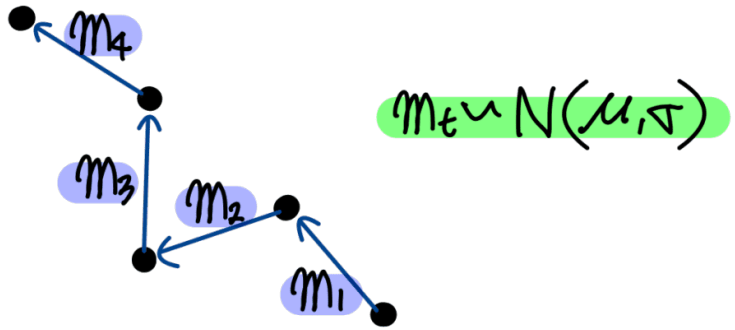
- Diffusion models are inspired by non-equilibrium thermodynamics.
- For a small fraction of the time, it is difficult to determine whether particles are moving in the direction of mixing or in the opposite direction.



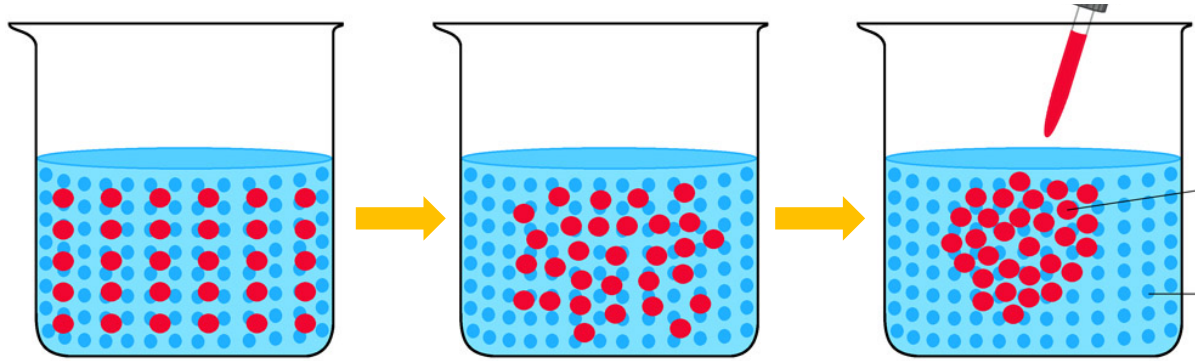




Diffusion Process

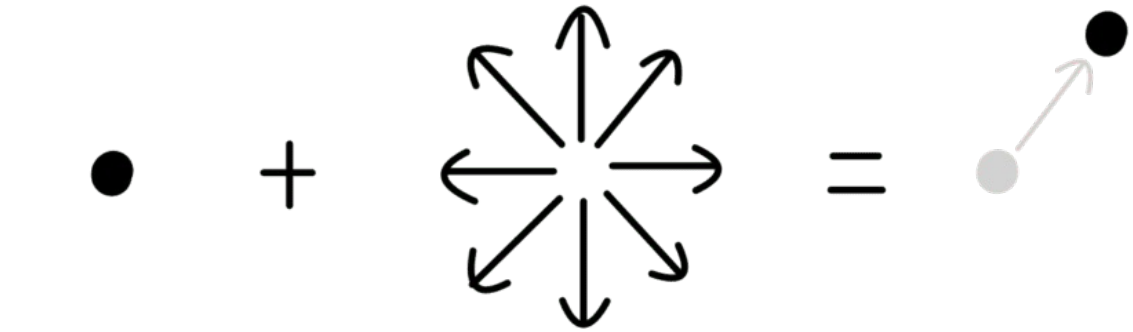


- If we look at the movement of a single molecule on a very short time scale, it follows a Gaussian distribution.
- Since the direction of mixing and the opposite direction are the same in a very short time, the opposite direction also follows a Gaussian distribution.



Diffusion Process

- Just as we viewed the molecule's motion as a Gaussian-distributed noise, we add a Gaussian-distributed noise to the pixel.



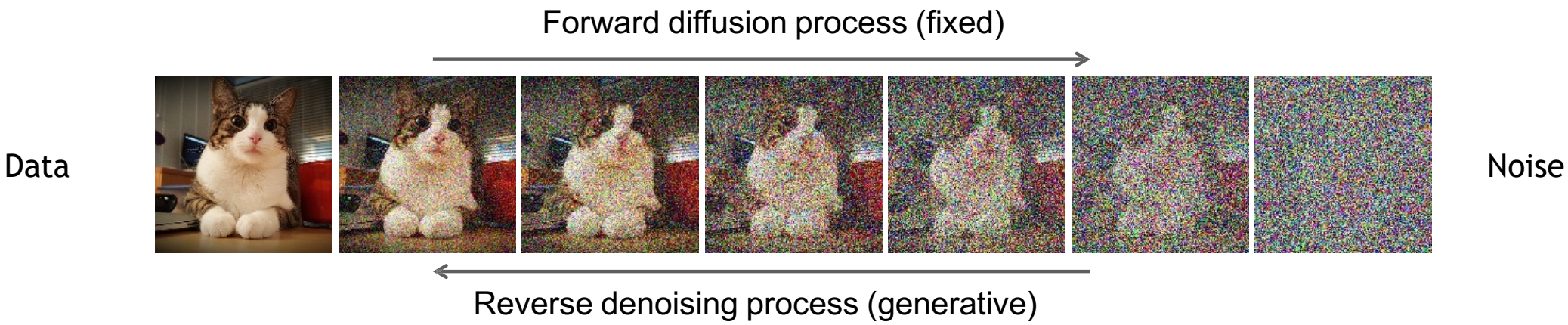
Molecule Noise movement Movement

Pixel Noise Pixel

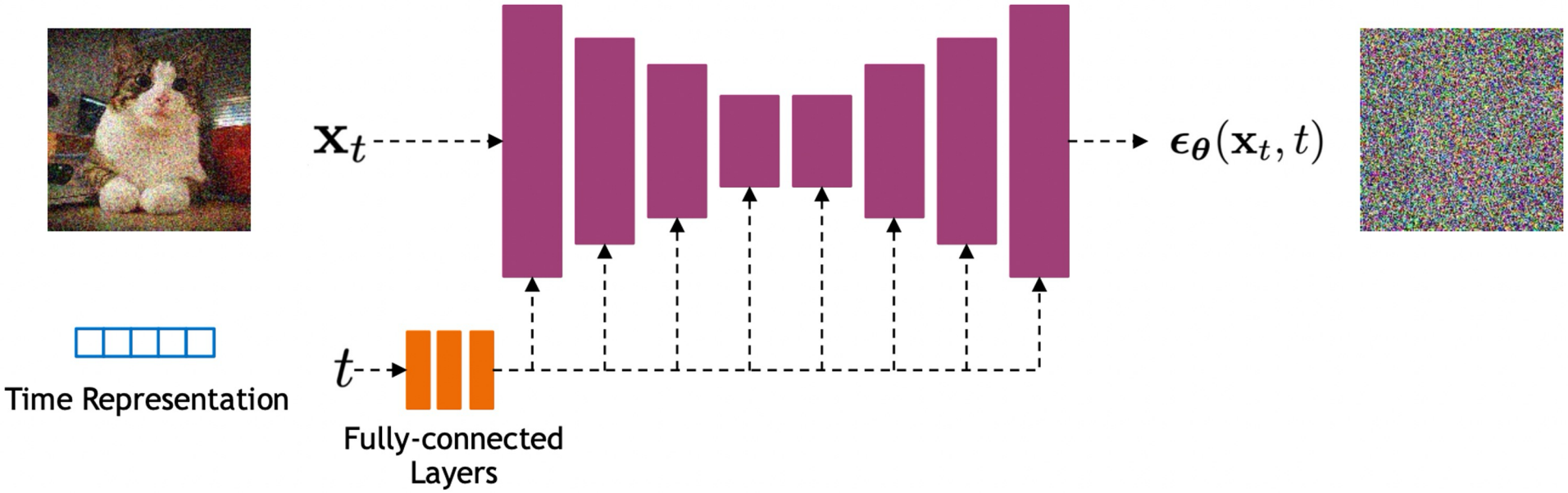
Denoising Diffusion Models

Denoising diffusion models consist of two processes:

- Forward diffusion process that gradually adds noise to input
- Reverse denoising process that learns to generate data by denoising



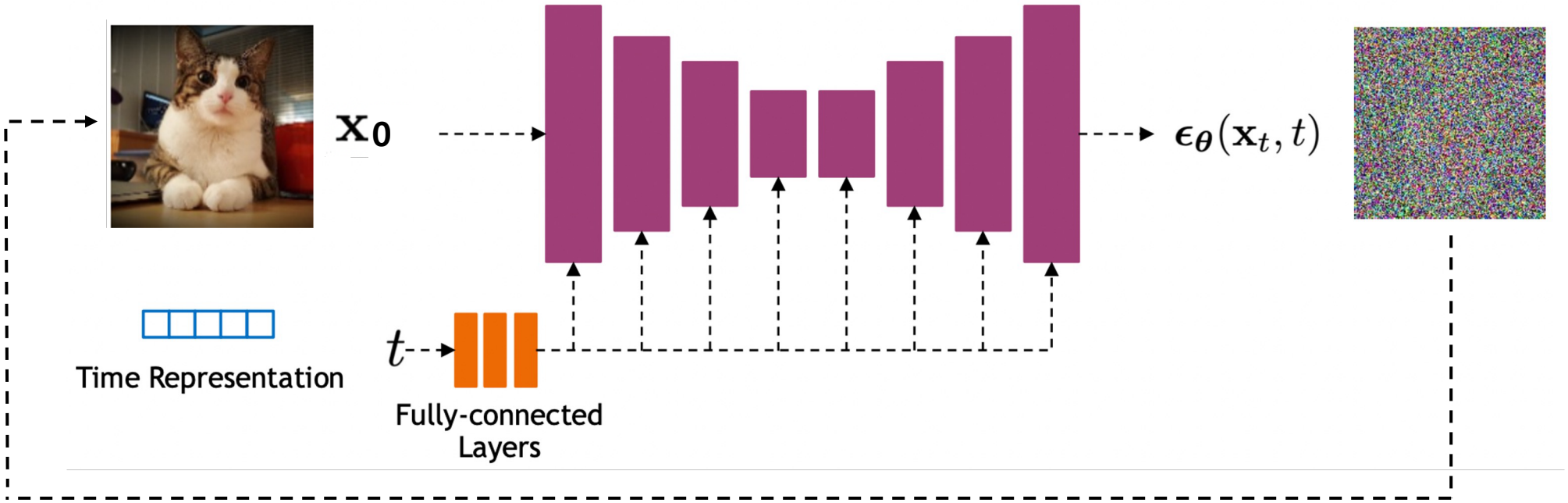
Denoising Diffusion Models : Training



Algorithm 1 Training

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

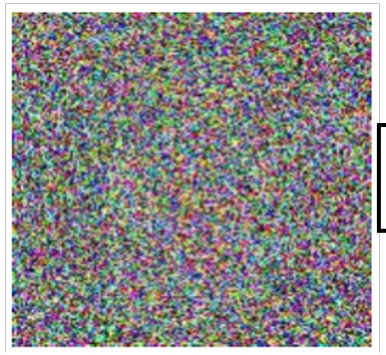
Denoising Diffusion Models : Sampling



Algorithm 2 Sampling

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

Results



Diffusion

Diffusion Model

- Pros
  - Intuitive Understanding: Diffusion in pixel space directly affects image pixels, making the changes visually easy to understand.

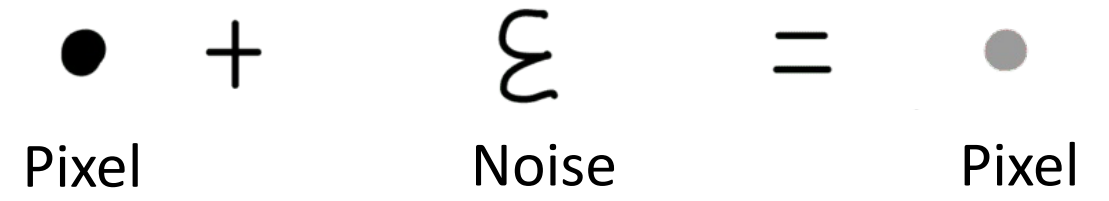
- Cons
  - Computational Cost

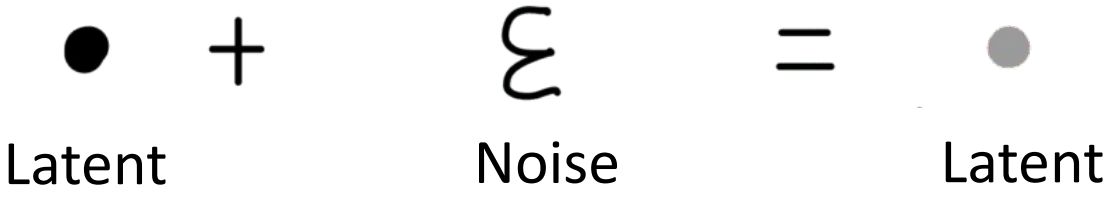
  : The larger the number of pixels, the greater the computation.
  - Memory Usage

  : Handling high-resolution images requires substantial memory.

Latent Diffusion Model

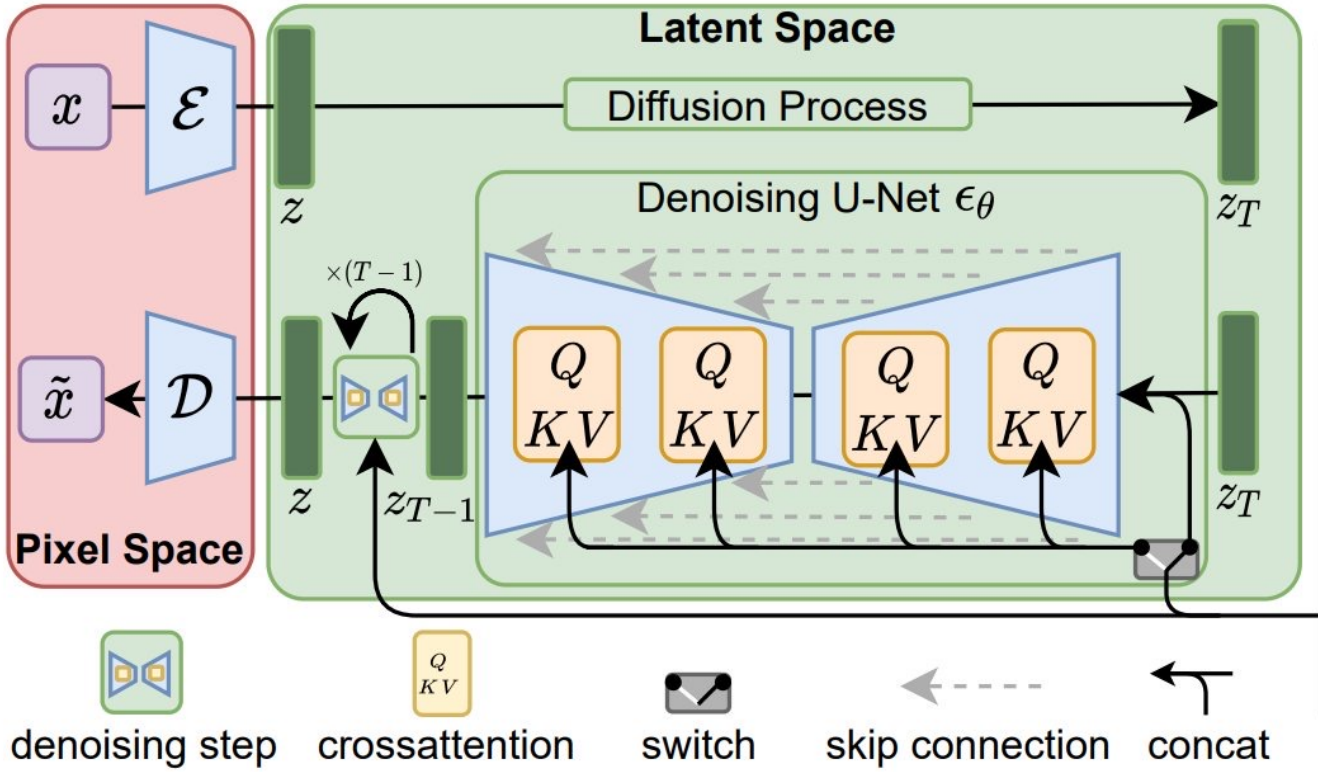
- Latent spaces typically have lower dimensions than pixel spaces, resulting in lower computational costs.
  - Pixel Space >> Latent Space





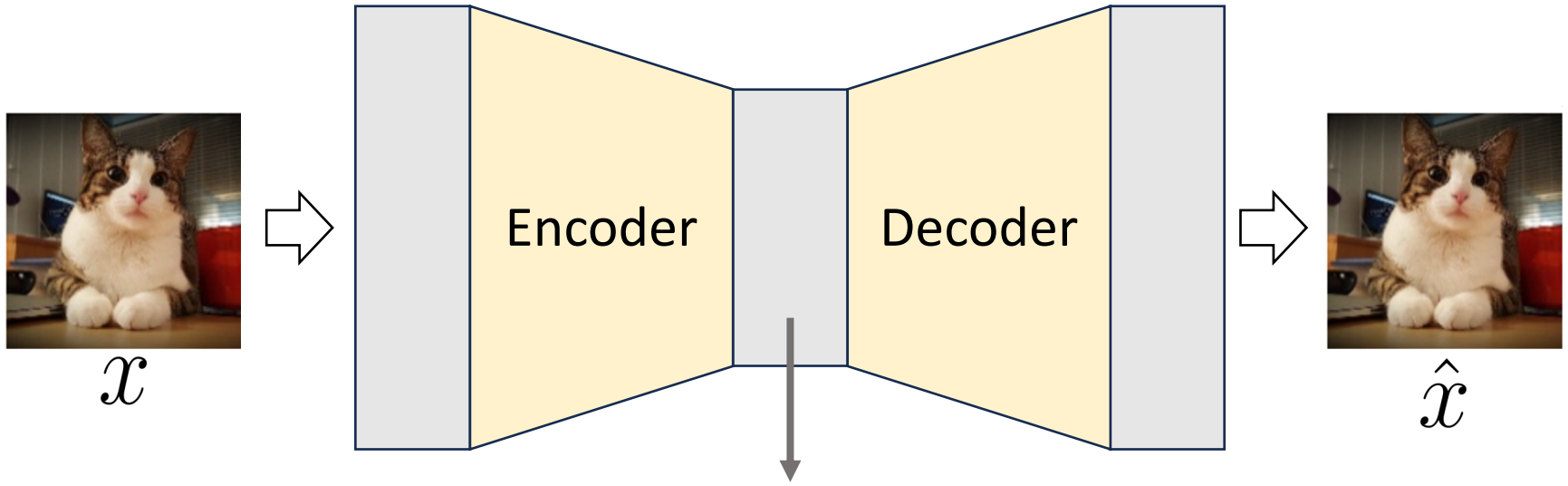
Latent Diffusion Model

- Runs the diffusion process in the latent space instead of pixel space
- 2 Stage Training : Auto-Encoder + Latent Diffusion



Latent Diffusion Model

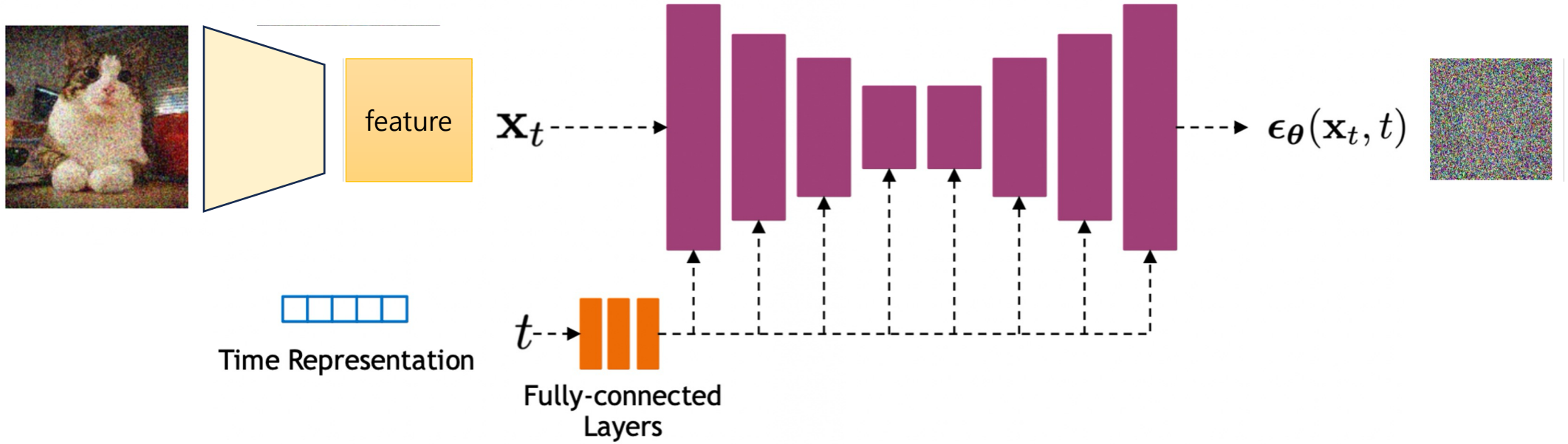
- Autoencoders can be particularly valuable as they enable a compressed yet remaining semantic and conceptual meaning of an image.



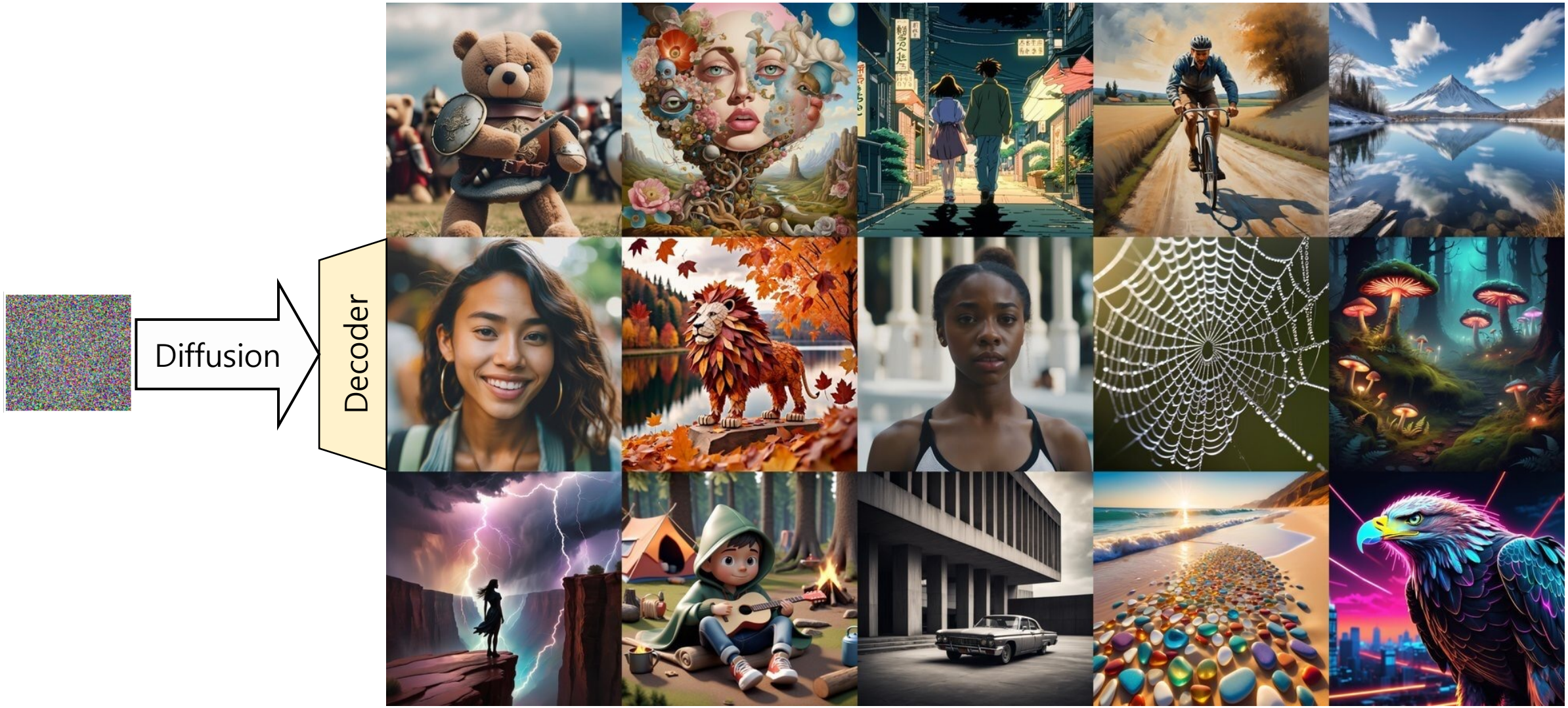
$$loss = \frac{1}{n}\sum_{i=0}^{n}(x_i - \hat{x}_i)^2$$

Latent Diffusion Model

- Runs the diffusion process in the latent space instead of pixel space
- 2 Stage Training : Auto-Encoder + Latent Diffusion



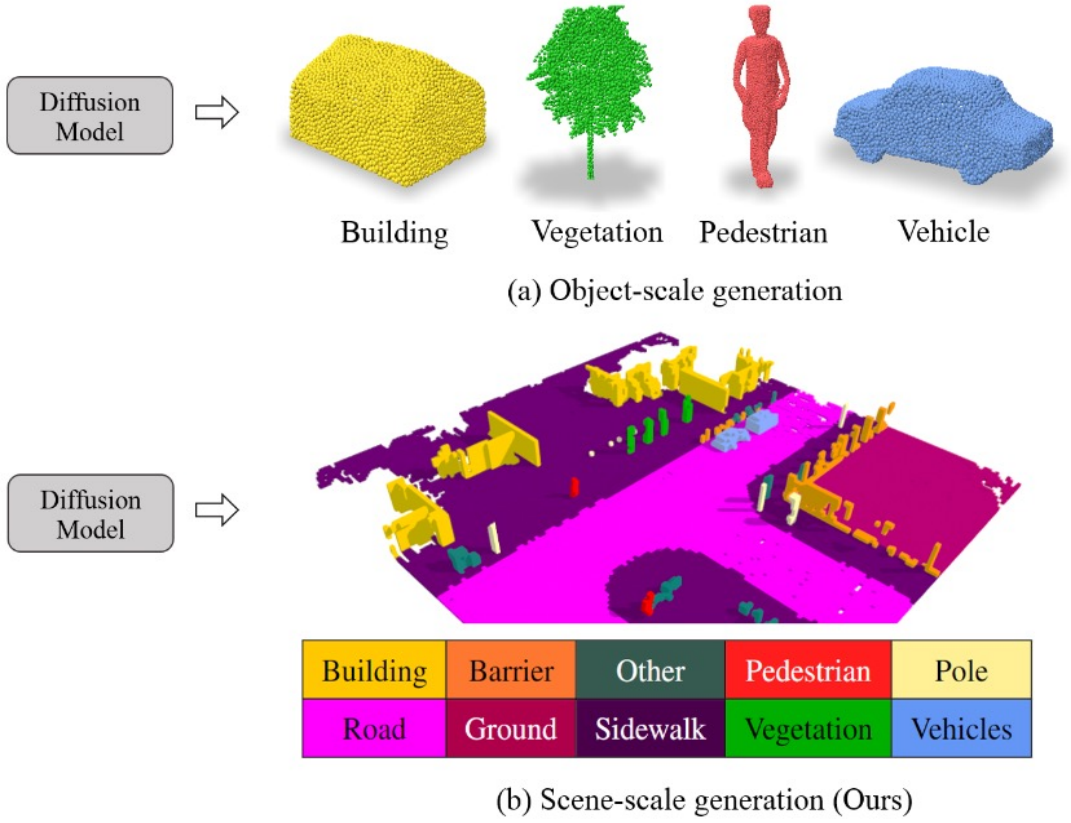
Results



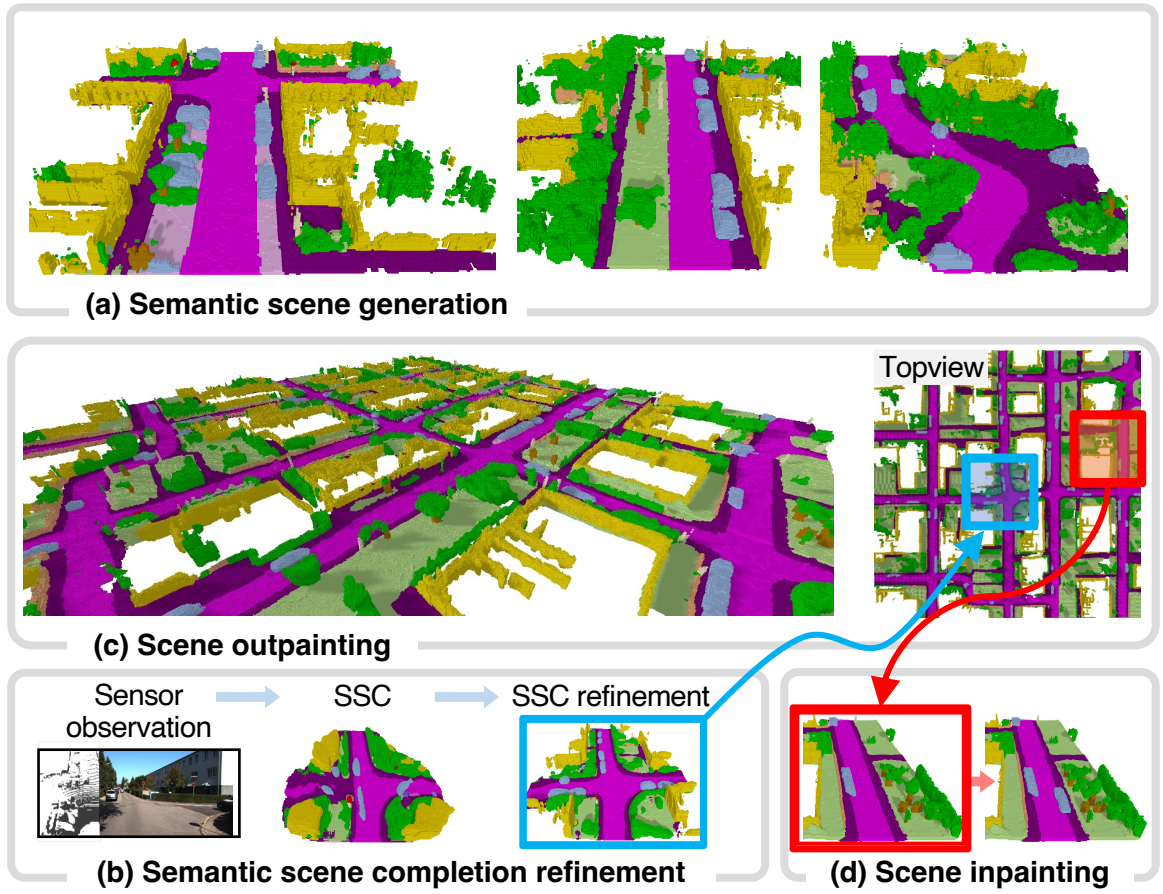
Our Goal

road sidewalk parking ground building traffic-sign car
truck bicycle motorcycle vehicle vegetation motorcyclist pole
terrain person bicyclist trunk fence empty (air)

Our Goal

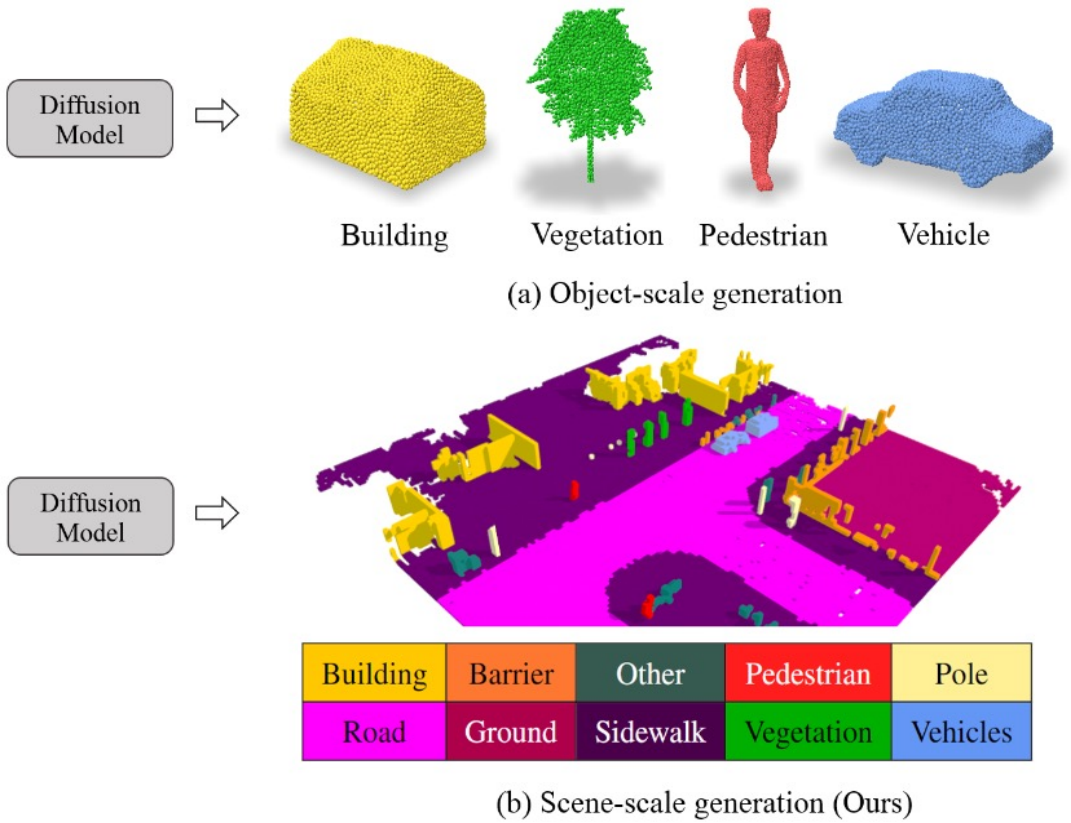


Jumin Lee, Woobin Im, Sebin Lee, Sung-Eui Yoon, *Diffusion Probabilistic Models for Scene-Scale 3D Categorical Data,* IPIU 2023 (grand prize)



Jumin Lee*, Sebin Lee*, Changho Jo, Woobin Im, Ju-Hyeong Seon, Sung-Eui Yoon, *SemCity: Semantic Scene Generation with Triplane Diffusion*, CVPR 2024

3D Scene-level Generation



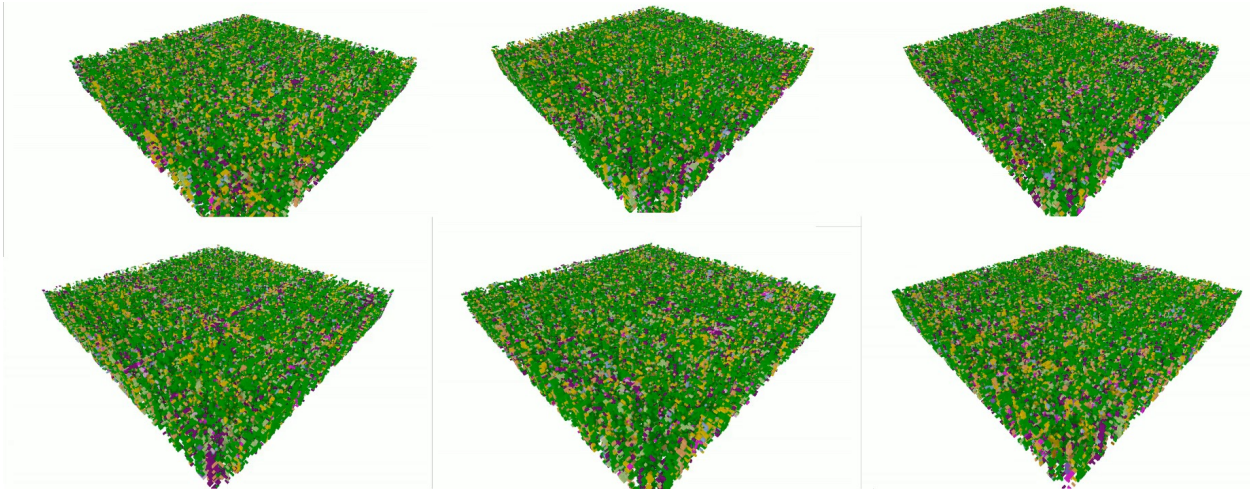
Jumin Lee, Woobin Im, Sebin Lee, Sung-Eui Yoon, *Diffusion Probabilistic Models for Scene-Scale 3D Categorical Data,* IPIU 2023 (grand prize)

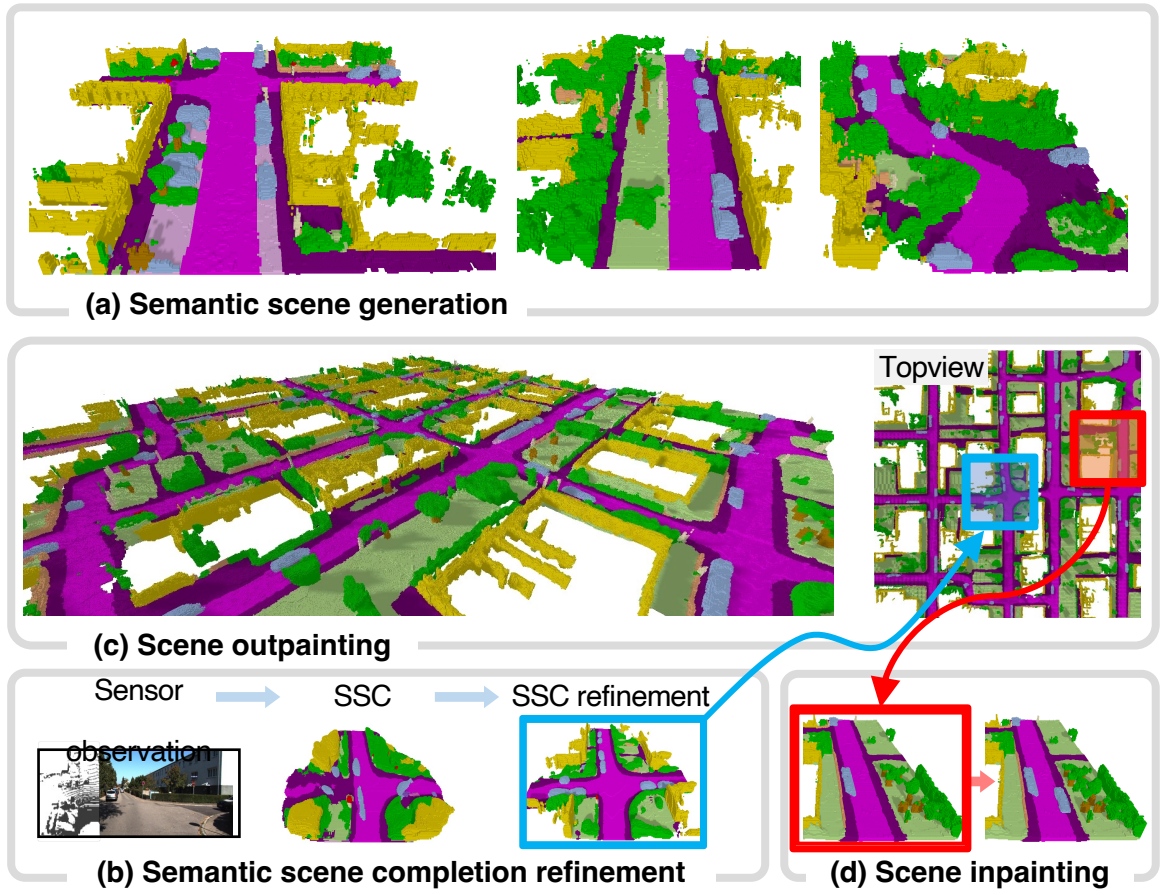
- Firstly apply the diffusion model at the 3D scene level not at the 3D object level.
- Show meaningful results.

road sidewalk parking ground building traffic-sign car
truck bicycle motorcycle vehicle vegetation motorcyclist pole
terrain person bicyclist trunk fence empty (air)

3D Scene-level Generation

- Enhance generation power.
- Extend our model with several applications (inpainting, outpainting, semantic scene completion refinement), as in the image domain.





Jumin Lee*, Sebin Lee*, Changho Jo, Woobin Im, Ju-Hyeong Seon, Sung-Eui Yoon, *SemCity: Semantic Scene Generation with Triplane Diffusion*, CVPR 2024

CS580

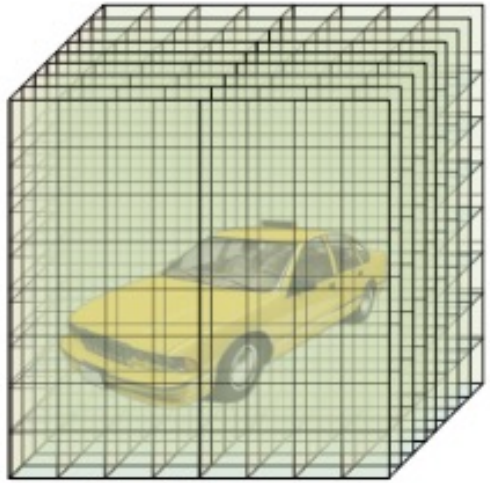
SemCity: Semantic Scene Generation with Triplane Diffusion

Jumin Lee, Sebin Lee, Changho Jo, Woobin Im, Ju-Hyeong Seon and Sung-Eui Yoon, *SemCity: Semantic Scene Generation with Triplane Diffusion, CVPR 2024*

SGVR Lab KAIST

Ideas

- Decompose a scene into 3 orthogonal 2D planes.
- Utilized in 3D object reconstruction.

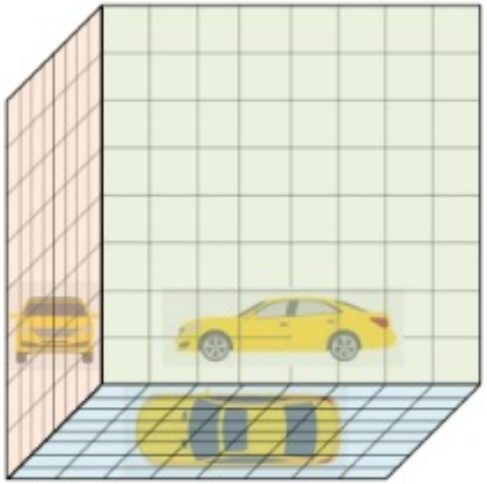


Voxel

Expressive

Bird's-Eye View

Efficient



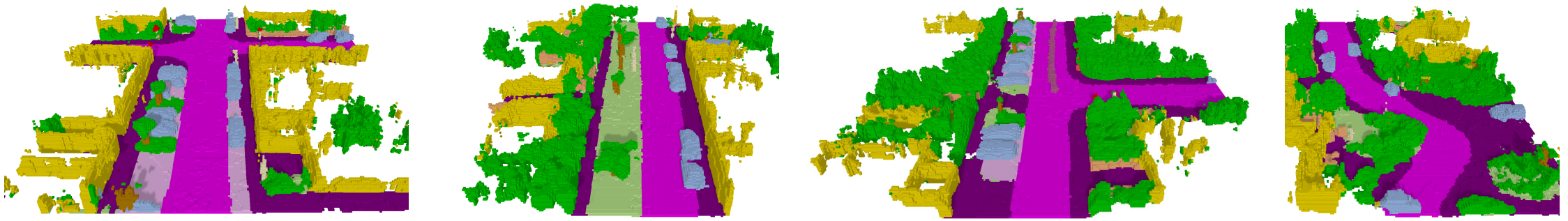
Triplane

Expressive & Efficient

parking ground building traffic-sign car
motorcycle vehicle vegetation motorcyclist pole
bicyclist trunk fence empty (air) road
terrain sidewalk bicycle person truck

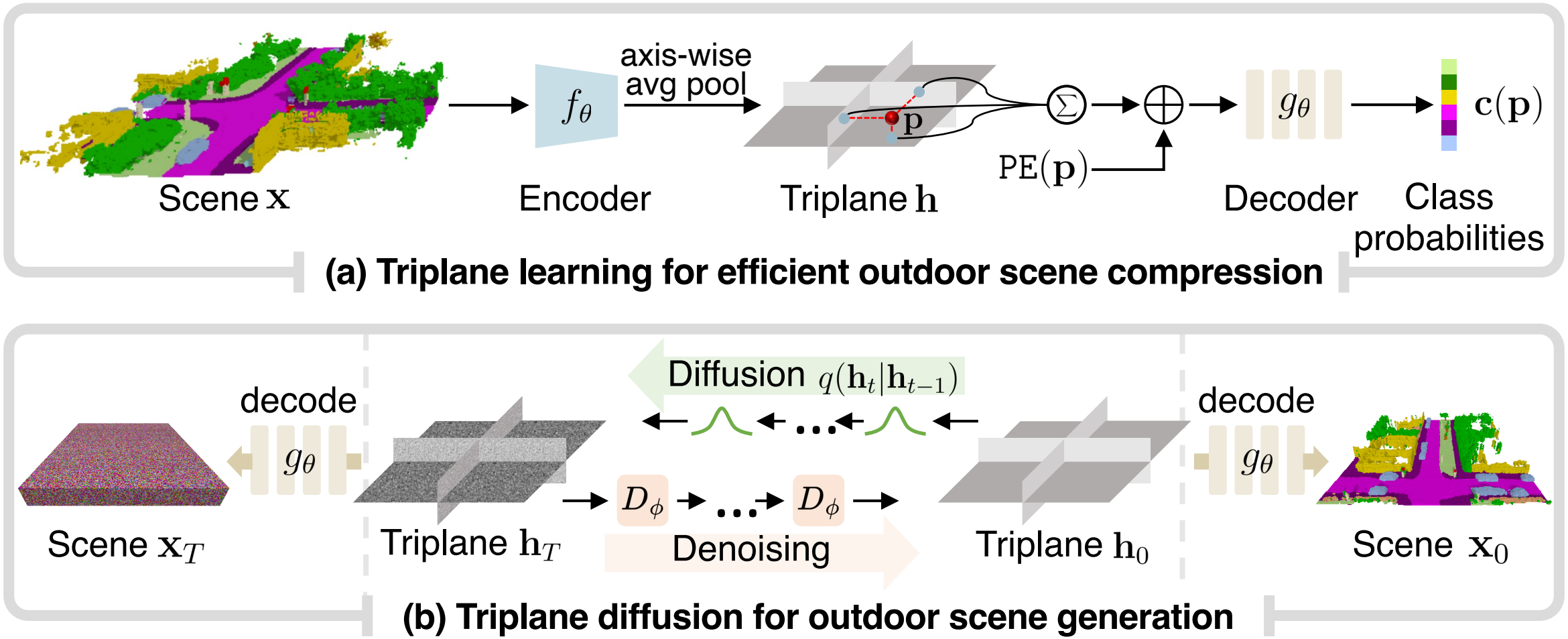
Ideas

- Leverage the triplane representation for the generation of real outdoor scenes.
  - Efficient and expressive.
  - Better focus on objects rather than empty region.
  - Spatial awareness representation helps capture semantic and geometric complexity within a scene.

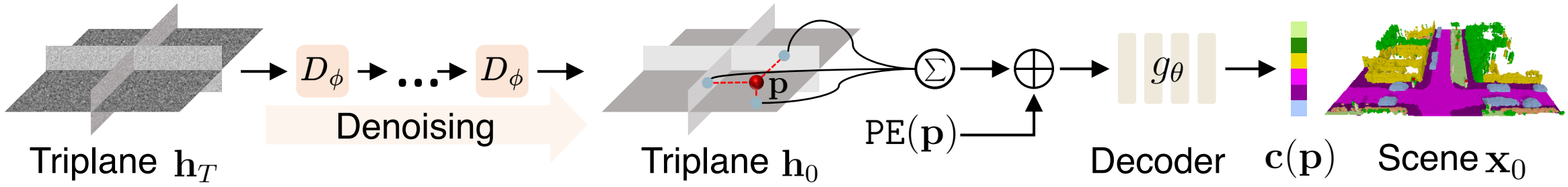


Scene generation

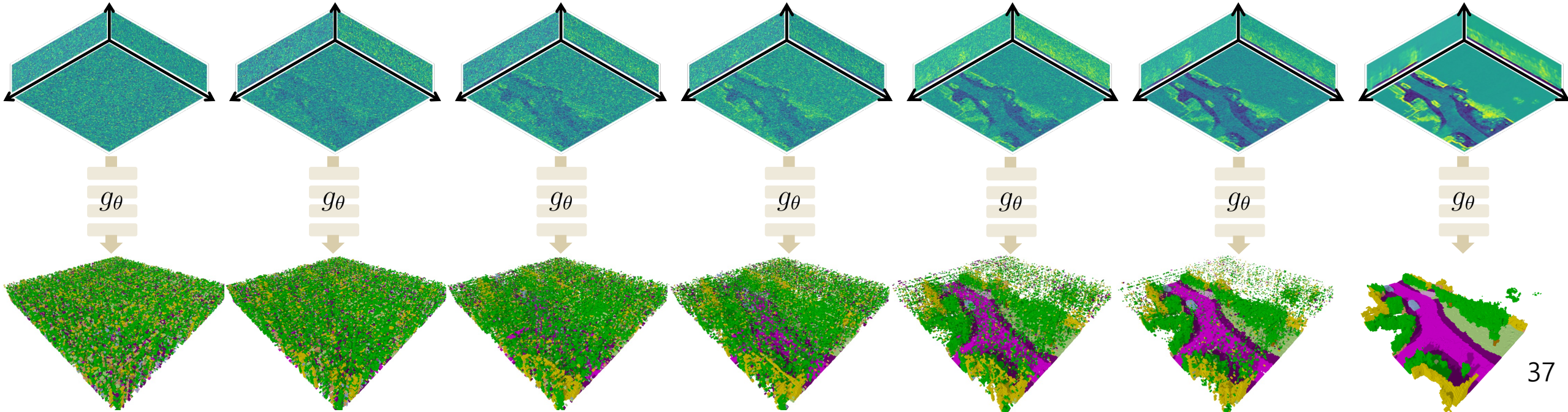
Method : Training



Method : Sampling



Triplane diffusion for outdoor scene generation



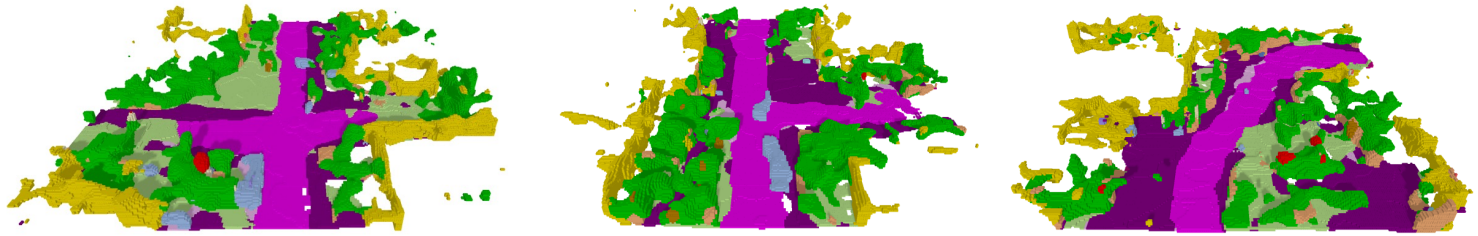
road
sidewalk
parking
ground
building
traffic-sign
car
truck
bicycle
motorcycle
vehicle
vegetation
motorcyclist
pole
terrain
person
bicyclist
trunk
fence
empty (air)

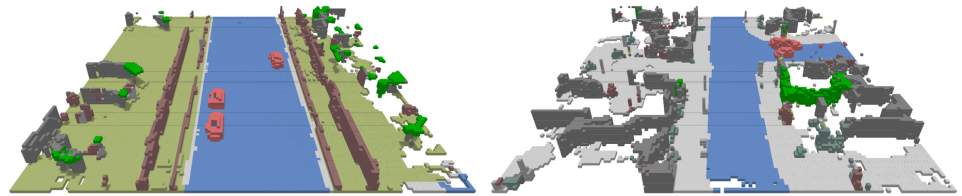
Generation Results

SemanticKITTI

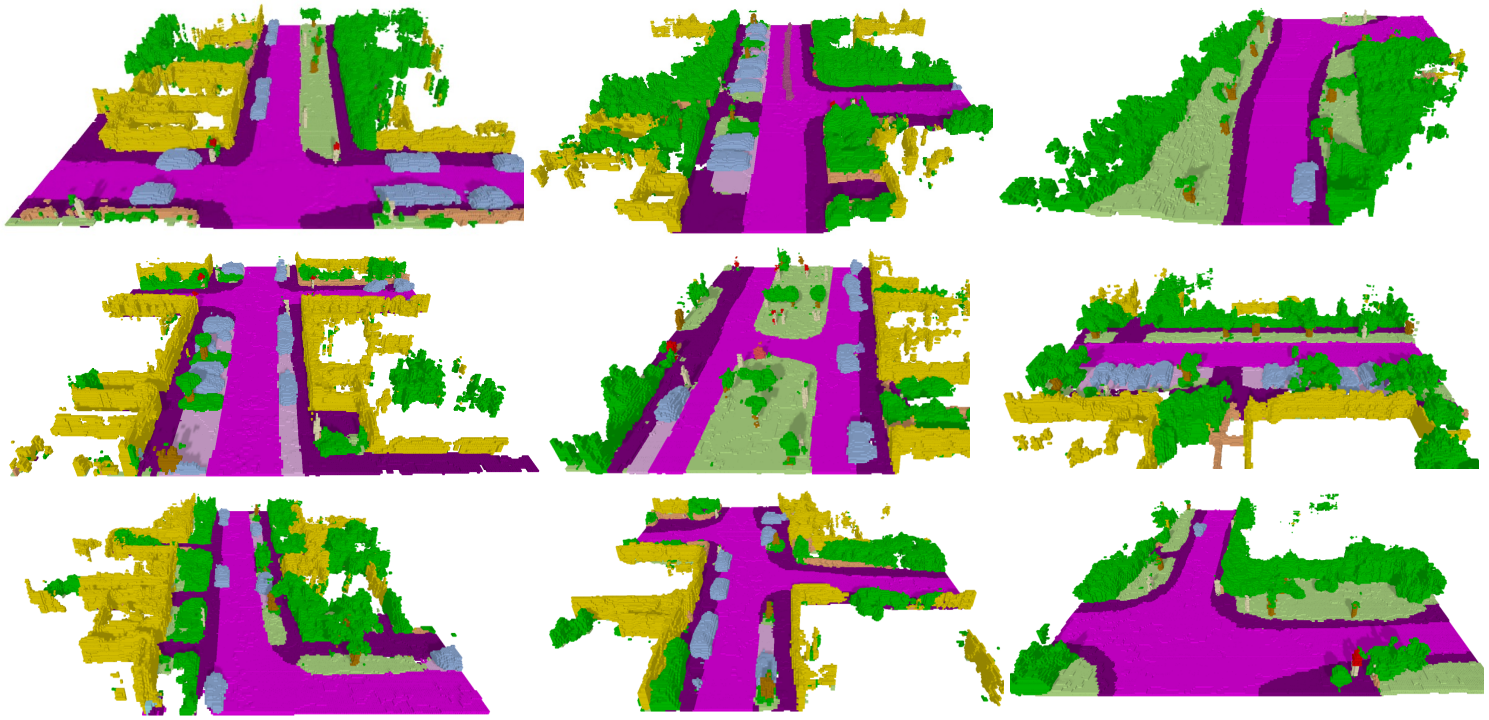
CarlaSC

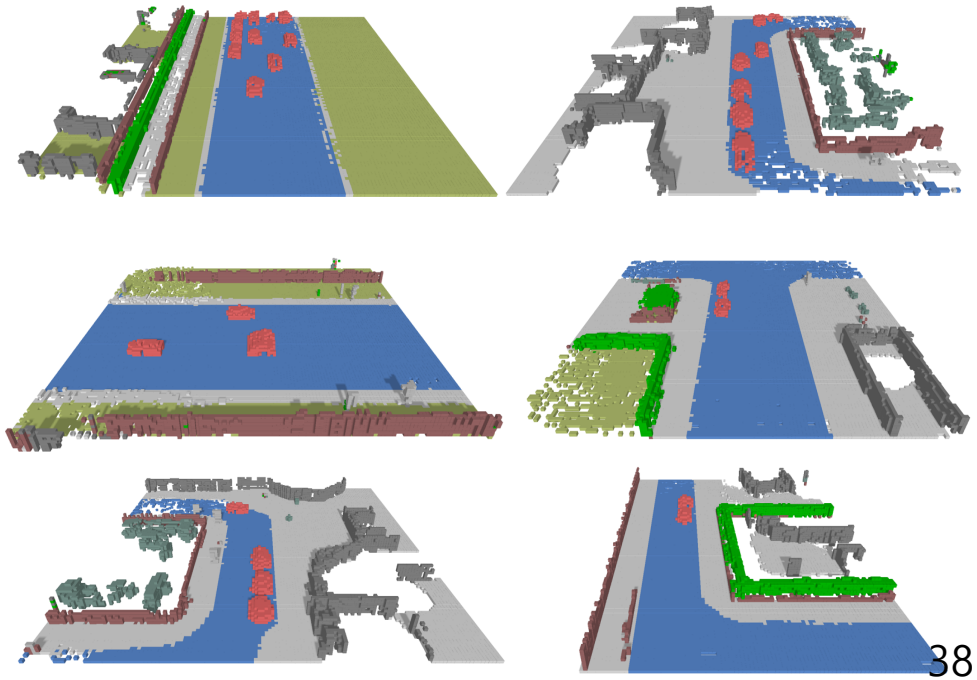
SSD





Ours





Generation Results

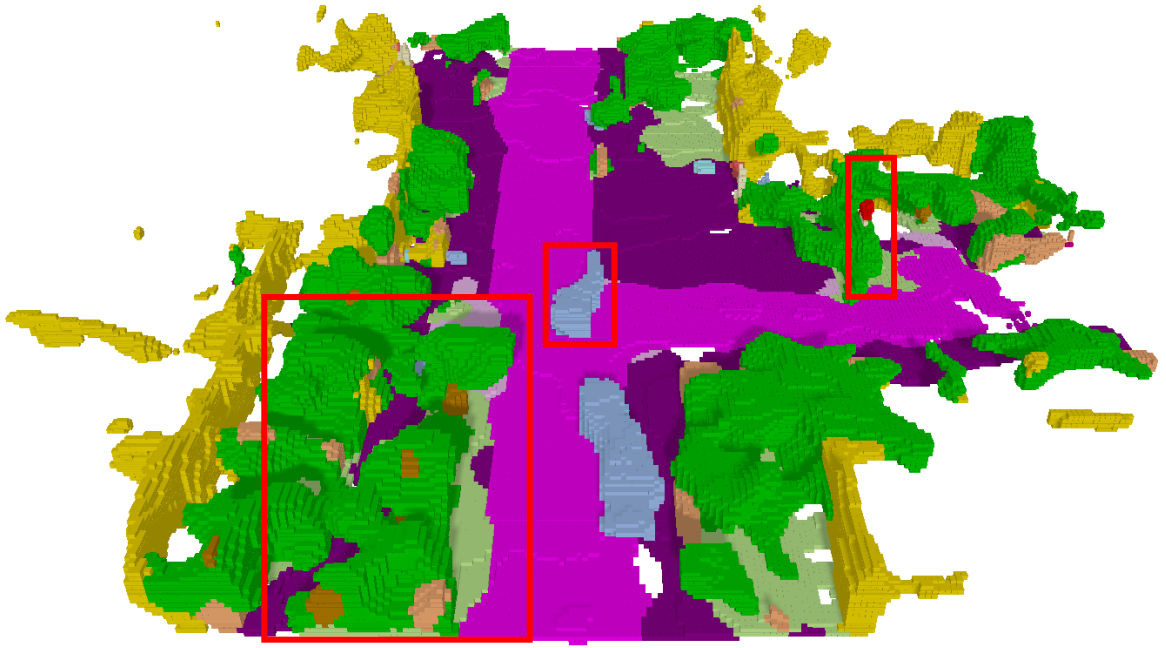
	Diversity & Fidelity		Fidelity		Diversity
Model	FID ↓	KID ↓	IS ↑	Prec ↑	Rec ↑
SemanticKITTI [6]					
SSD [24]	112.82	0.12	2.23	0.01	0.08
SemCity (Ours)	56.55	0.04	3.25	0.39	0.32
CarlaSC [50]					
SSD [24]	87.39	0.09	2.44	0.14	0.07
SemCity (Ours)	40.63	0.02	3.51	0.31	0.09

Quantitative results of semantic scene generation

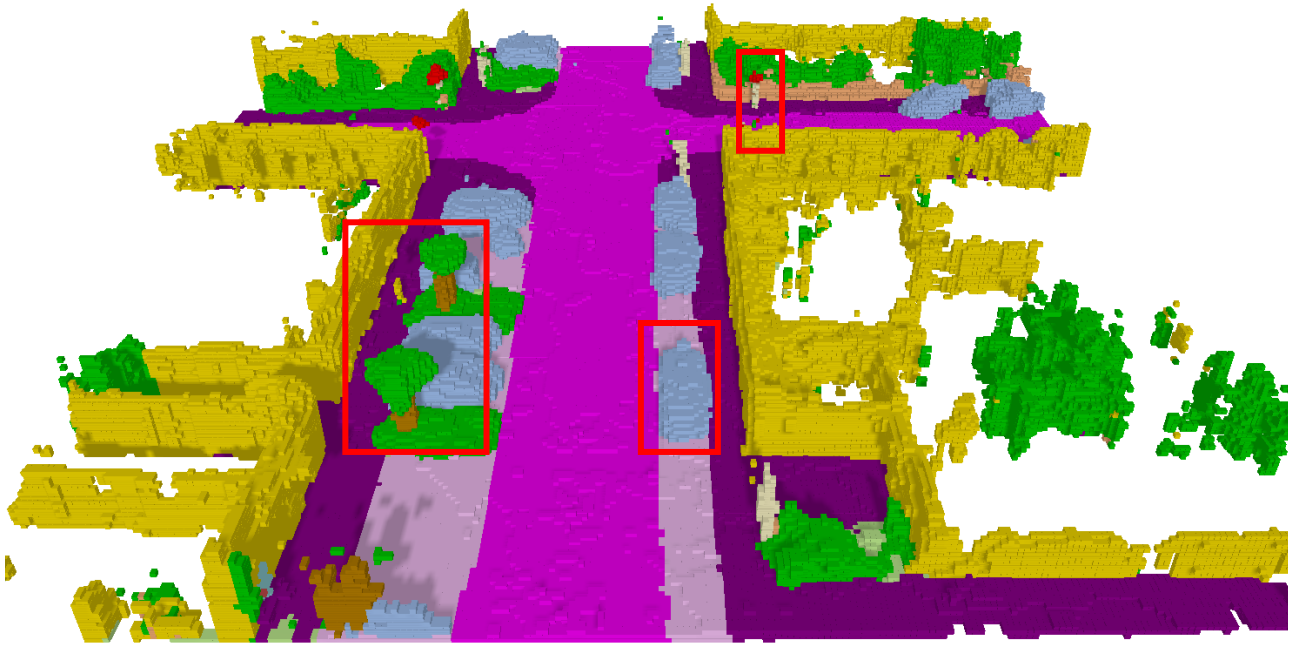
road sidewalk parking ground building traffic-sign car
truck bicycle motorcycle vehicle vegetation motorcyclist pole
terrain person bicyclist trunk fence empty (air)

Generation Results : Comparison

SSD



SemCity

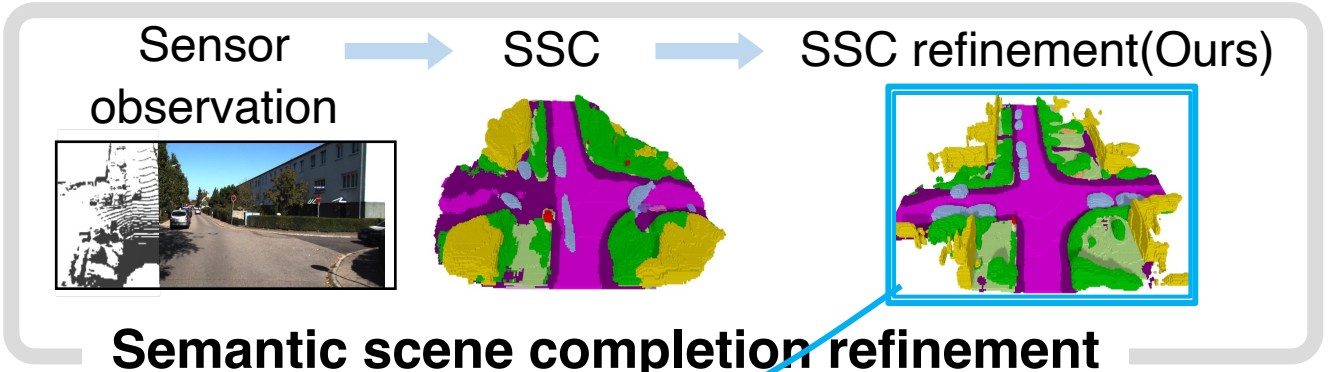


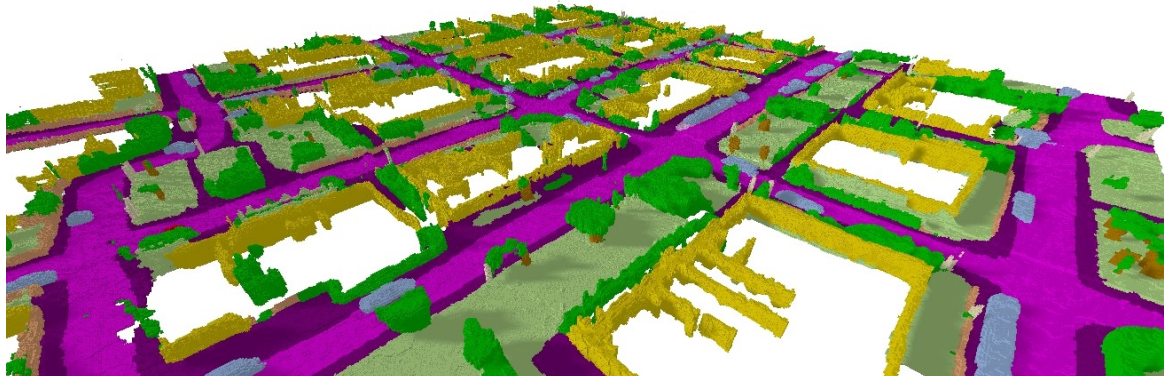
- Overall contours : road, building
- Finer structures : trunk and leave, traffic light and pole, car

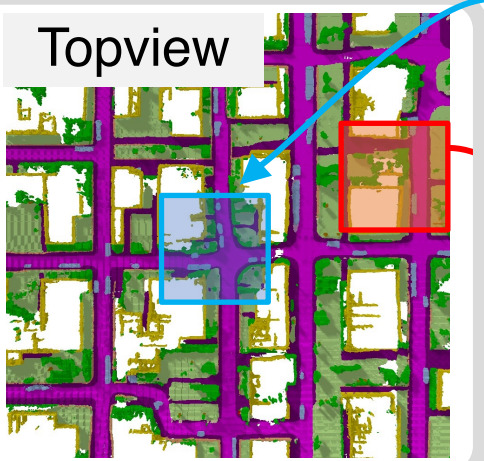
road sidewalk parking ground building traffic-sign car
truck bicycle motorcycle vehicle vegetation motorcyclist pole
terrain person bicyclist trunk fence empty (air)

Conditional Generation

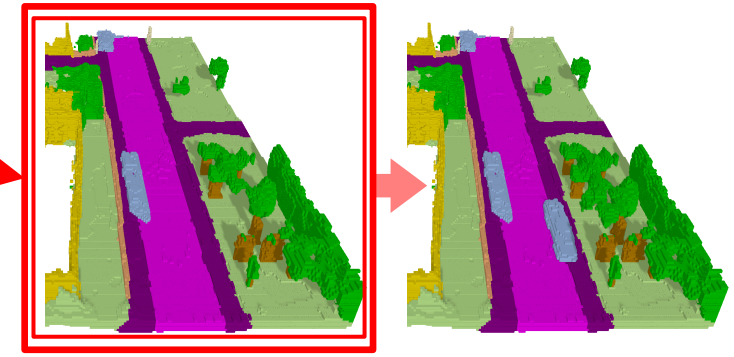
- We extend our model to refine the predictions of SSC models.







Scene outpainting



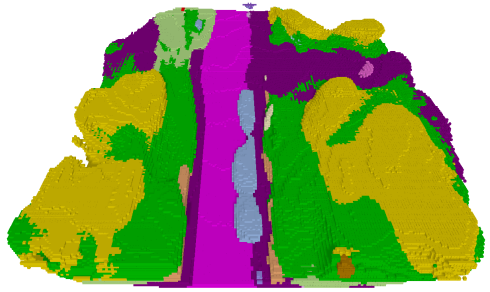
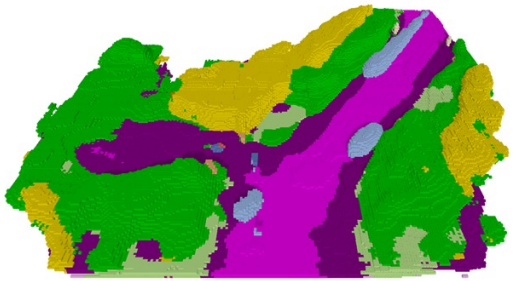
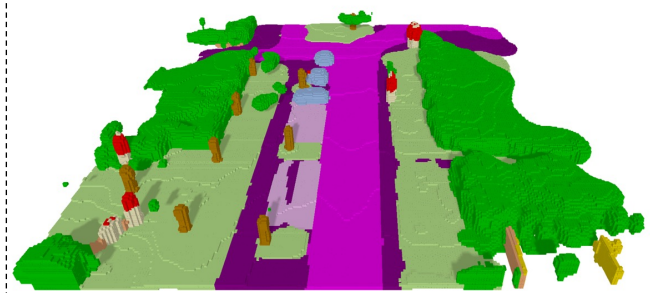
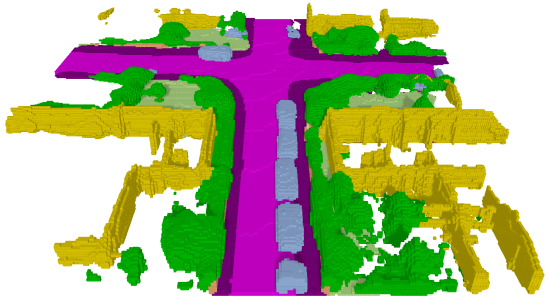
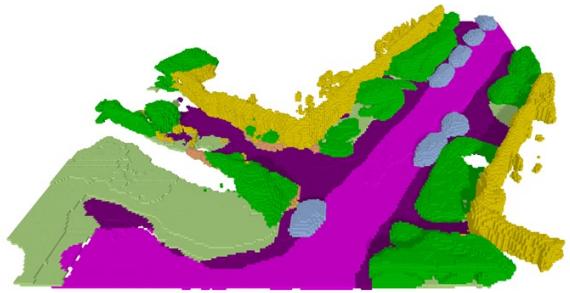
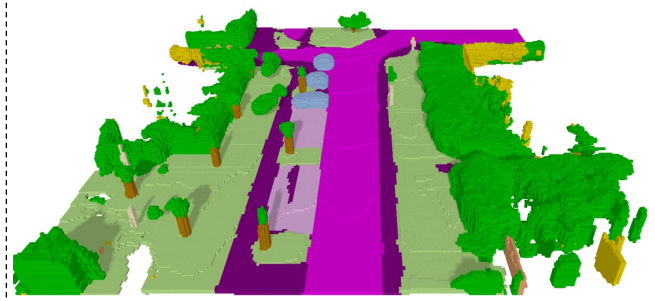
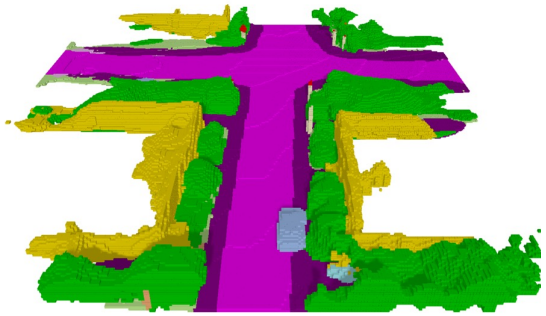
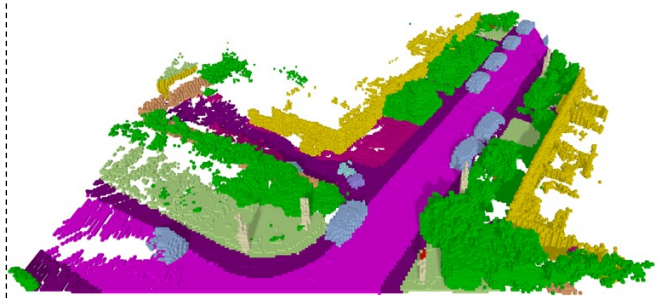
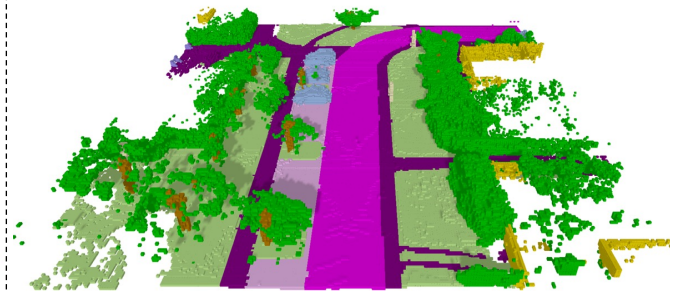
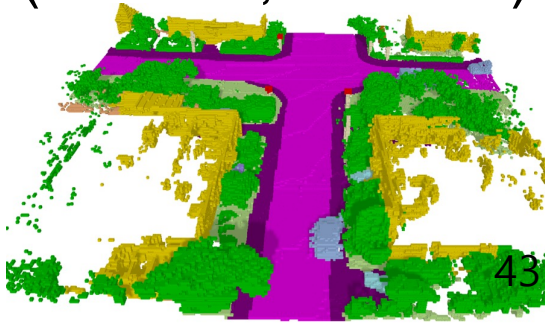
Scene inpainting

- We propose to manipulate triplane features during our diffusion process for scene outpainting and inpainting.

road sidewalk parking ground building traffic-sign car
truck bicycle motorcycle vehicle vegetation motorcyclist pole
terrain person bicyclist trunk fence empty (air)

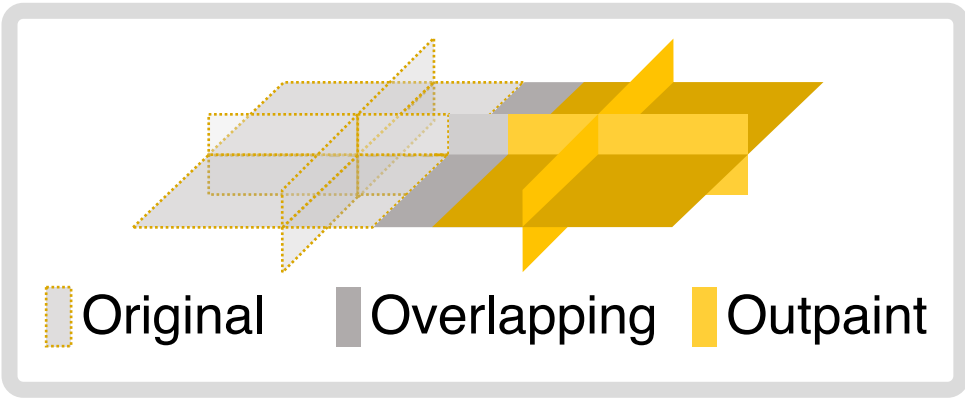
Semantic Scene Completion Refinement

(·, ·) : IoU, mIoU

	MonoScene	OccDepth	SSA-SC	SCPNet
SSC	 (33.95%, 12.37%)	 (44.89%, 13.96%)	 (64.02%, 37.22%)	 (39.81%, 28.97%)
Ours	 (51.57%, 21.45%)	 (56.98%, 22.67%)	 (67.07%, 45.36%)	 (54.55%, 33.46%)
GT				

Scene Outpainting

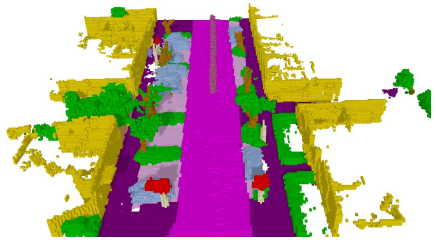
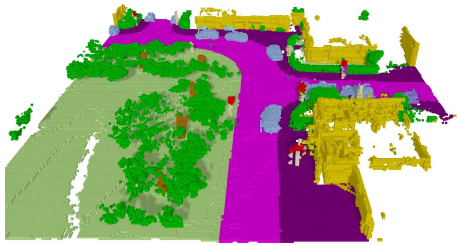
256 x 256 x 32 → 1792 x 2816 x 32

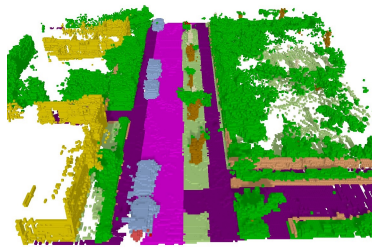


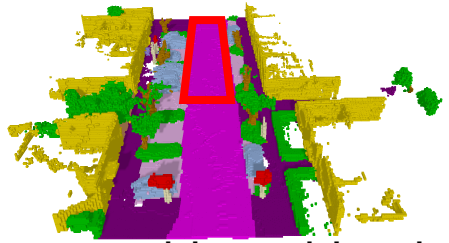
Scene Outpainting

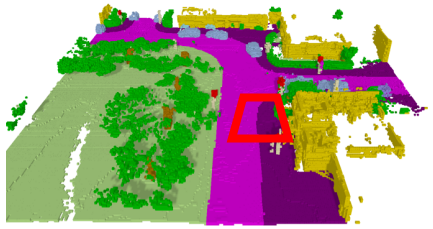
Scene Inpainting

Given scenes

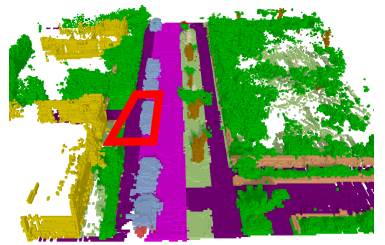






Remove object : bicyclelist


Remove object : car

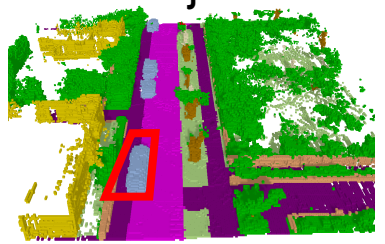

Add object : truck

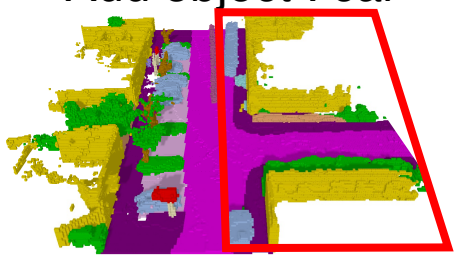

Add object : car

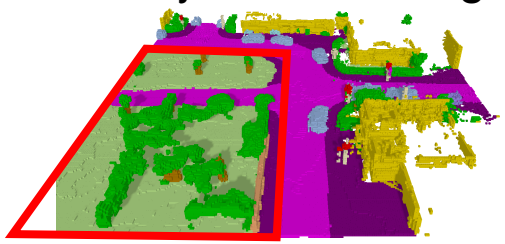

Add object : car

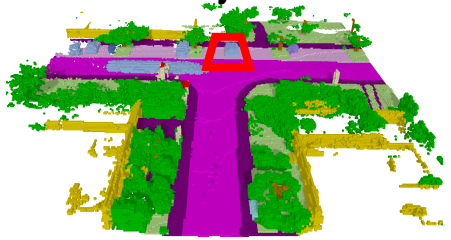

Add object : traffic sign


Add object : car


Modify object : car → car


Modify scene


Modify scene


Modify object : truck → car

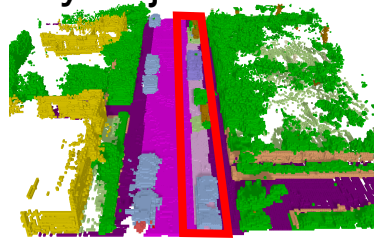
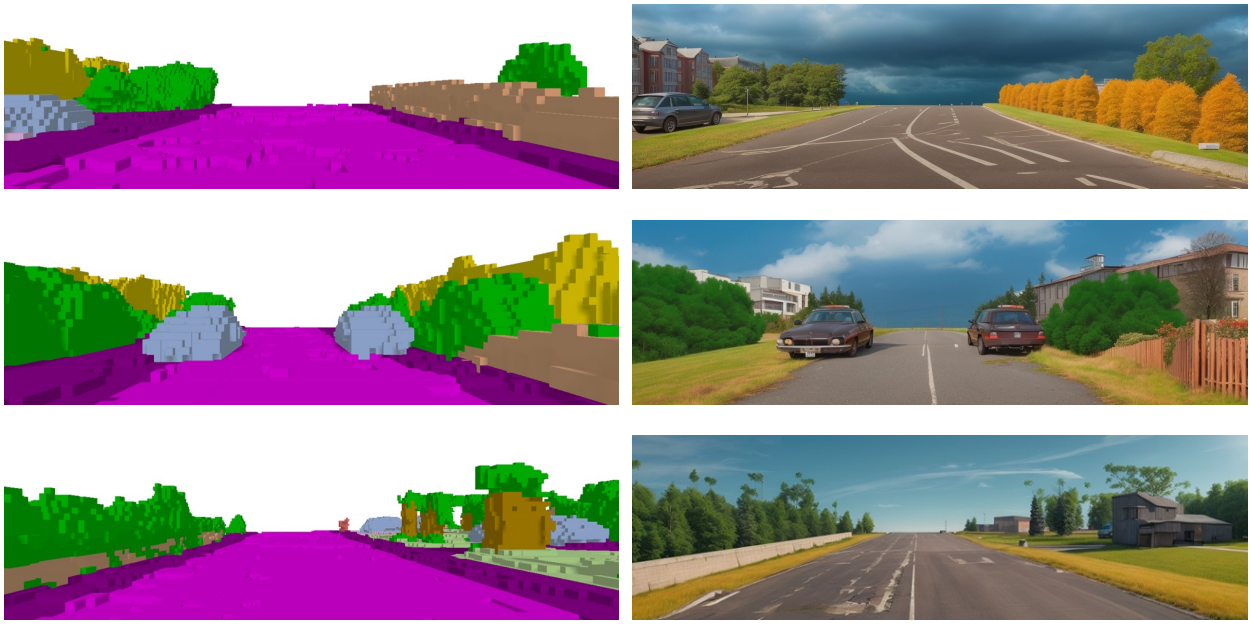
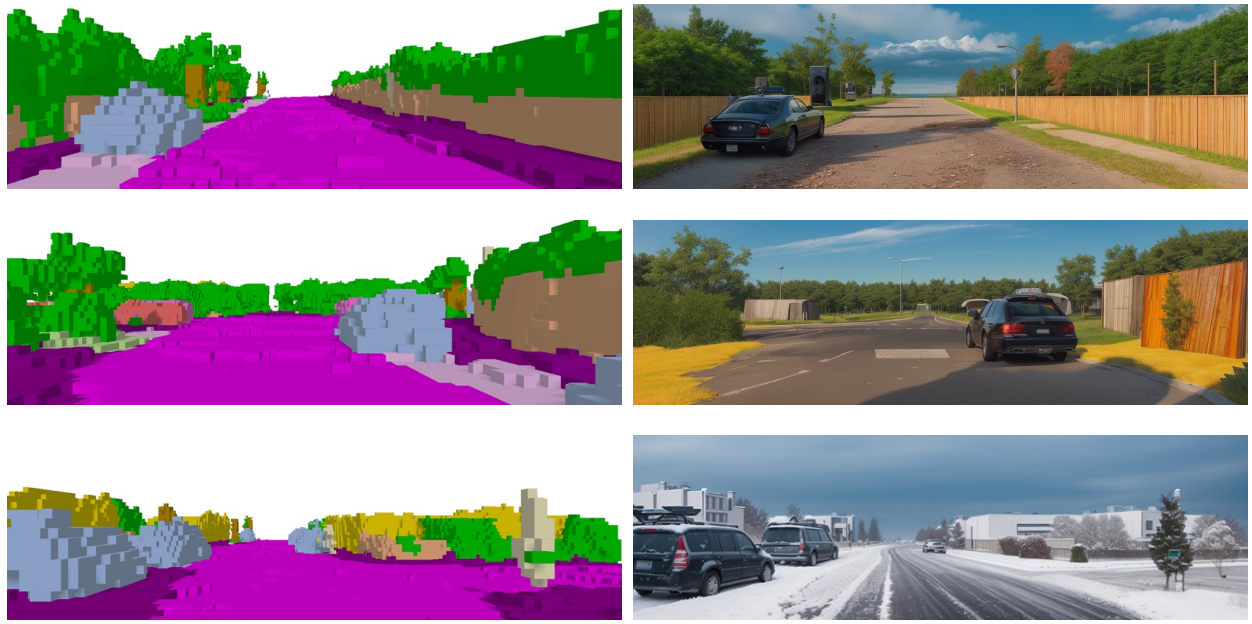

Modify scene

Image to Image Generation

- Exploit ControlNet to generate RGB images by conditioning semantic and depth maps rendered from our generated scene.



Generated scene Generated image



Generated scene Generated image

Conclusion

SemCity Public

[CVPR 2024] The official implementation for "SemCity: Semantic Scene Generation with Triplane Diffusion"

Python 195 16

scene-scale-diffusion Public

The official implementation for "Diffusion Probabilistic Models for Scene-Scale 3D Categorical Data"

Python 175 9

Diffusion Model for Scene-level Generation

- Firstly utilized the diffusion model on a 3D outdoor dataset.
- Enhancing outdoor scenes generation through a triplane representation.
- By manipulating triplane, our model can both inpaint and outpaint scenes.
- Our model can refine the outcomes of existing semantic scene completion model by utilizing learned 3D scene prior.

Thank you.