ICLR 2024 3D Generation

ICLR2024 paper list on 3D generation with brief introduction about each paper

Oral

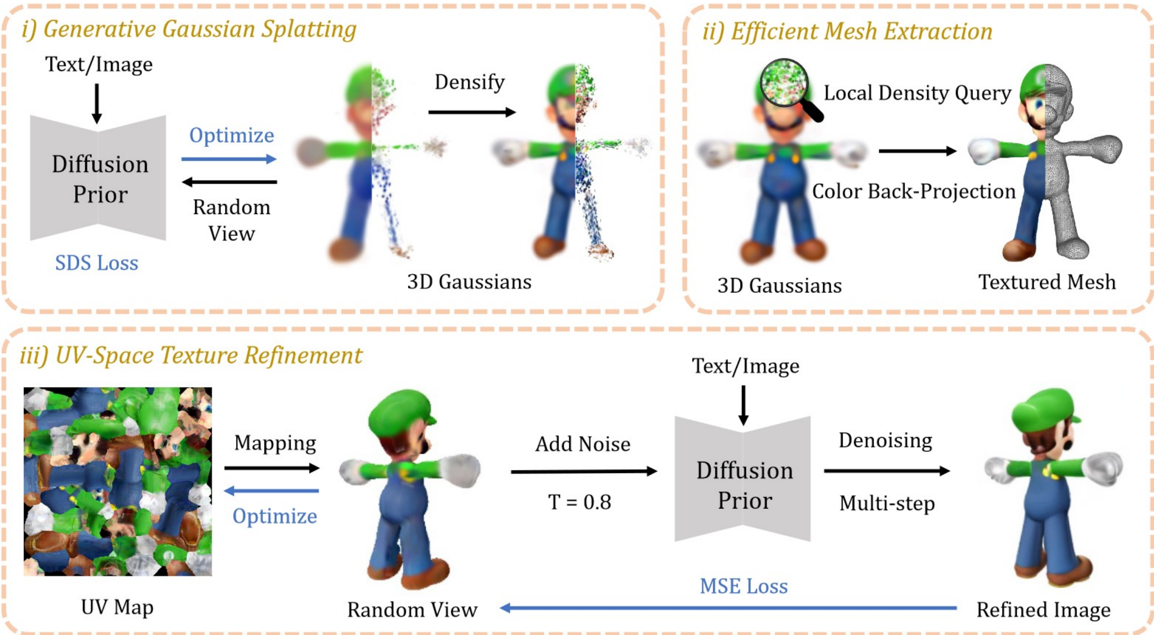
DreamGaussian: Generative Gaussian Splatting for Efficient 3D Content Creation (8 8 8 10)

Authors: Jiaxiang Tang , Jiawei Ren, Hang Zhou , Ziwei Liu , Gang Zeng

▶ **Abstract**

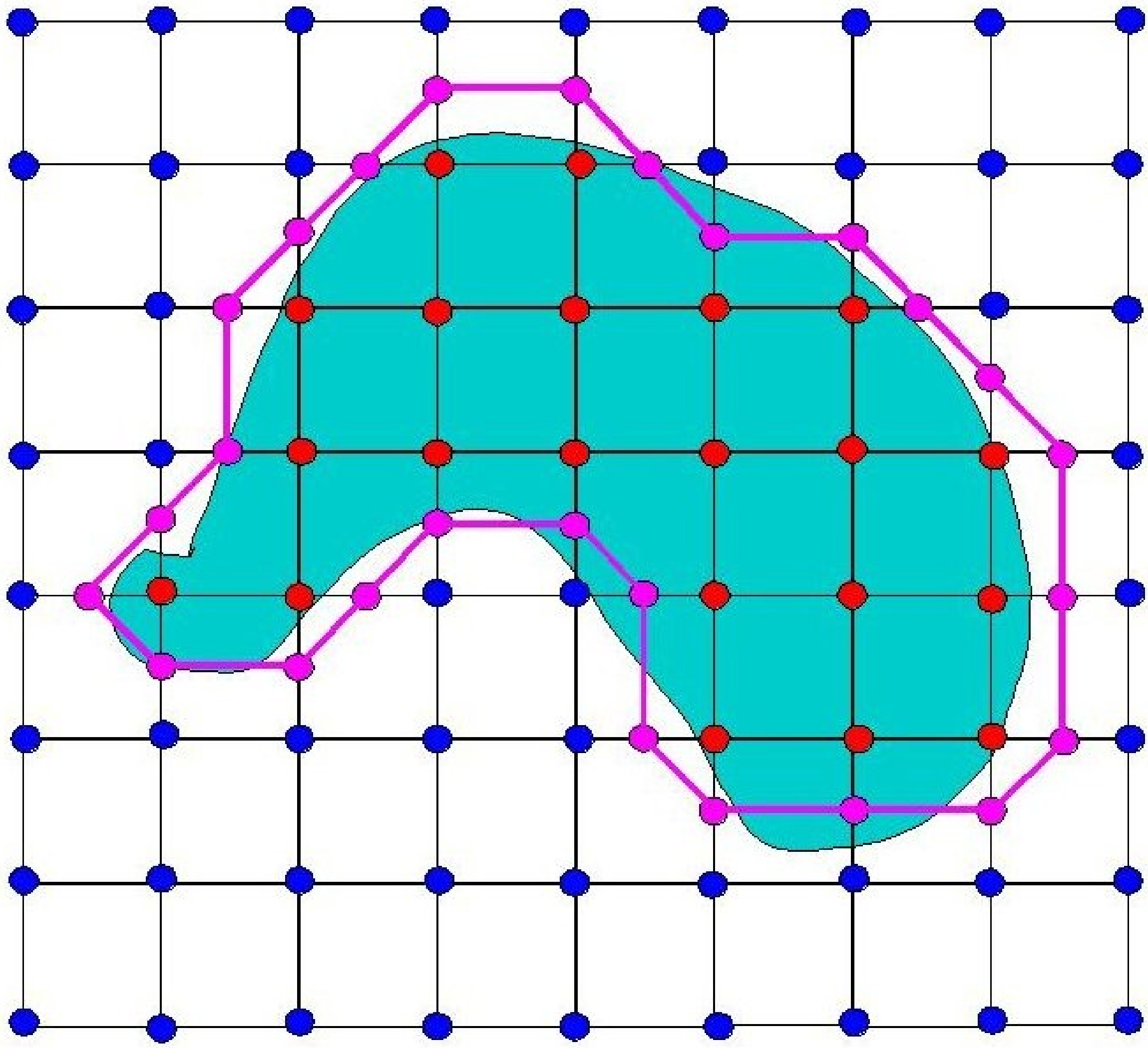
📄 Paper 🌐 Project 💻 Code #object_generation #texture_refinement #diffusion #3DGS #SDS

Pipeline

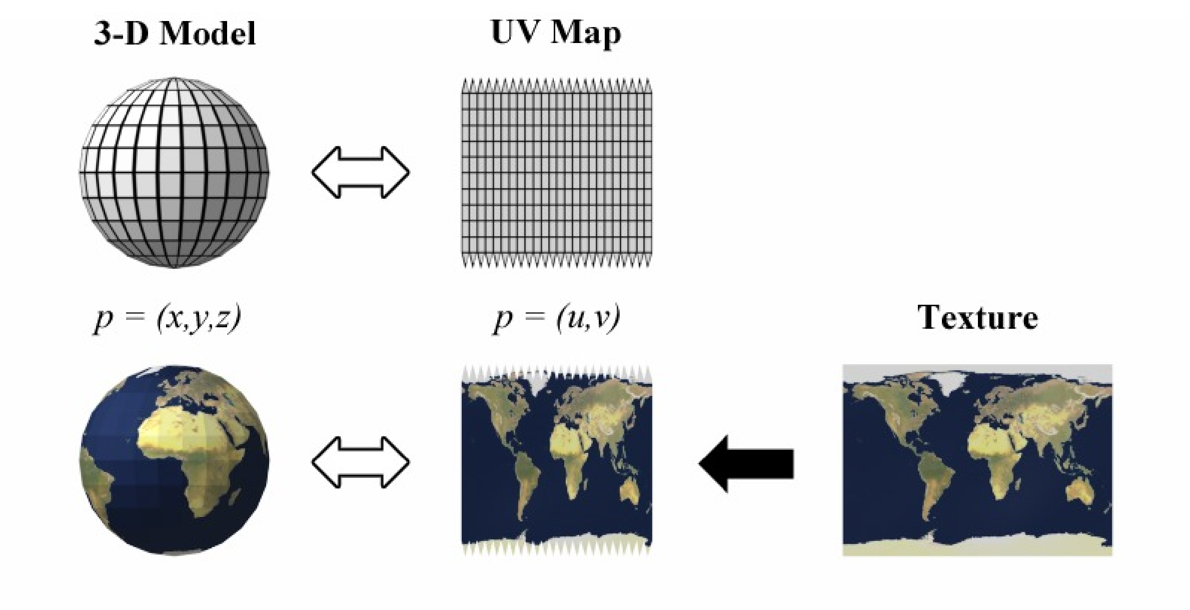


Mesh Extraction

Marching cubes (2D case as an example)



UV Mapping



LRM: Large Reconstruction Model for Single Image to 3D (8 8 8 10)

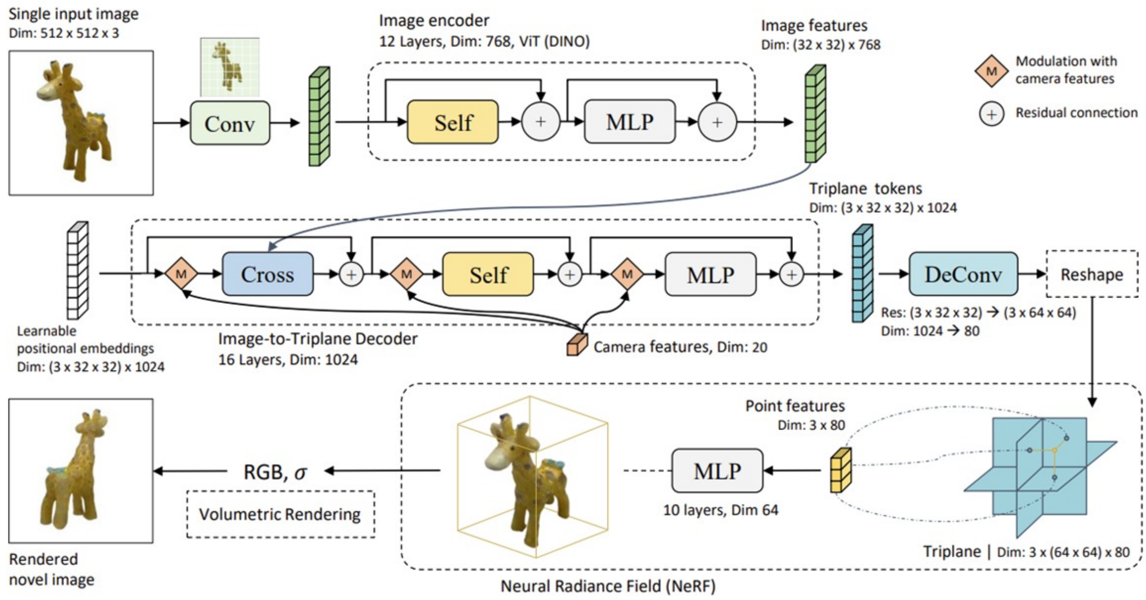
Authors: Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, Hao Tan

▶ **Abstract**

Paper Project #object_generation #triplane #NeRF

Pipeline

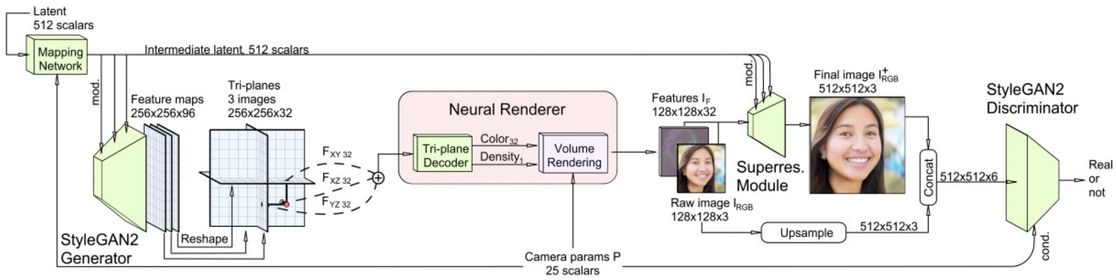
Image → Image feature → Triplane tokens (Triplane Nerf)



A fully trained large transformer decoder can convert a single image to its corresponding triplane

Ralated works:

- TensoRF: Tensorial Radiance Fields (ECCV2022, Triplane NeRF) PAPER
- EG3D: Efficient Geometry-aware 3D Generative Adversarial Networks (CVPR2022, stylegan generator → image feature → triplane feature → volume rendering → stylegan discriminator) PAPER!



Spotlight

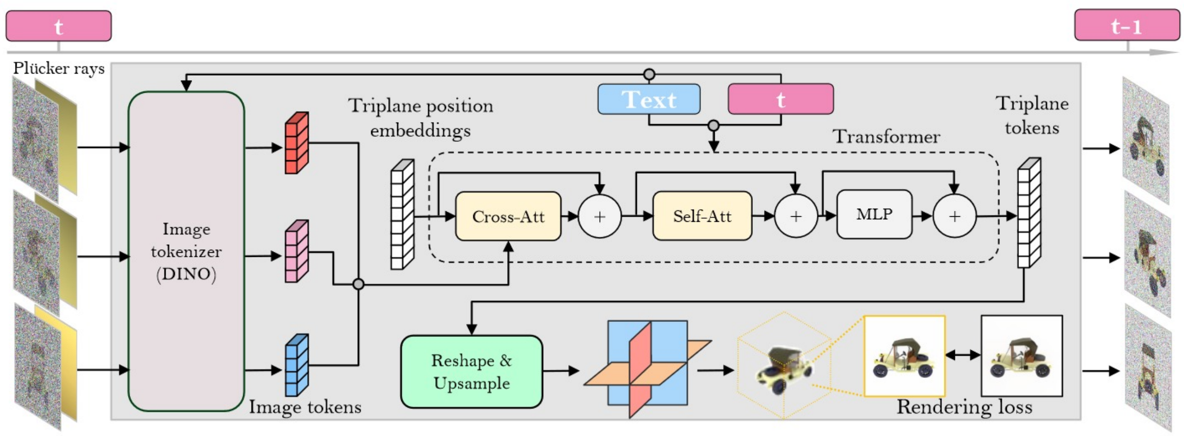
DMV3D: Denoising Multi-view Diffusion Using 3D Large Reconstruction Model (6 8 8 10)

Authors: Yinghao Xu, Hao Tan, Fujun Luan, Sai Bi, Peng Wang, Jiahao Li, Zifan Shi, Kalyan Sunkavalli, Gordon Wetzstein, Zexiang Xu, Kai Zhang

▶ **Abstract**

📄 Paper 🌐 Project #object_generation #triplane #diffusion #multi-view_diffusion #viewpoint_information #NeRF

Pipeline



Use LRM to replace the UNet of diffusion model.
End-to-end training, during inference stage, once the multi-view images are fully denoised, our model offers a clean triplane NeRF, enabling 3D generation.

- Multi-view images as input, and add noise on different images with the same schedule $\mathcal{I} = \{\mathbf{I}_1, \ldots, \mathbf{I}_N\}, \mathcal{I}_t = \{\sqrt{\bar{\alpha}_t}\mathbf{I} + \sqrt{1-\bar{\alpha}_t}\epsilon_{\mathbf{I}} \mid \mathbf{I} \in \mathcal{I}\}$
- Use LRM decoder to convert noisy multi-view images into triplane tokens
- Rendering denoised multi-view images from the triplane NeRF

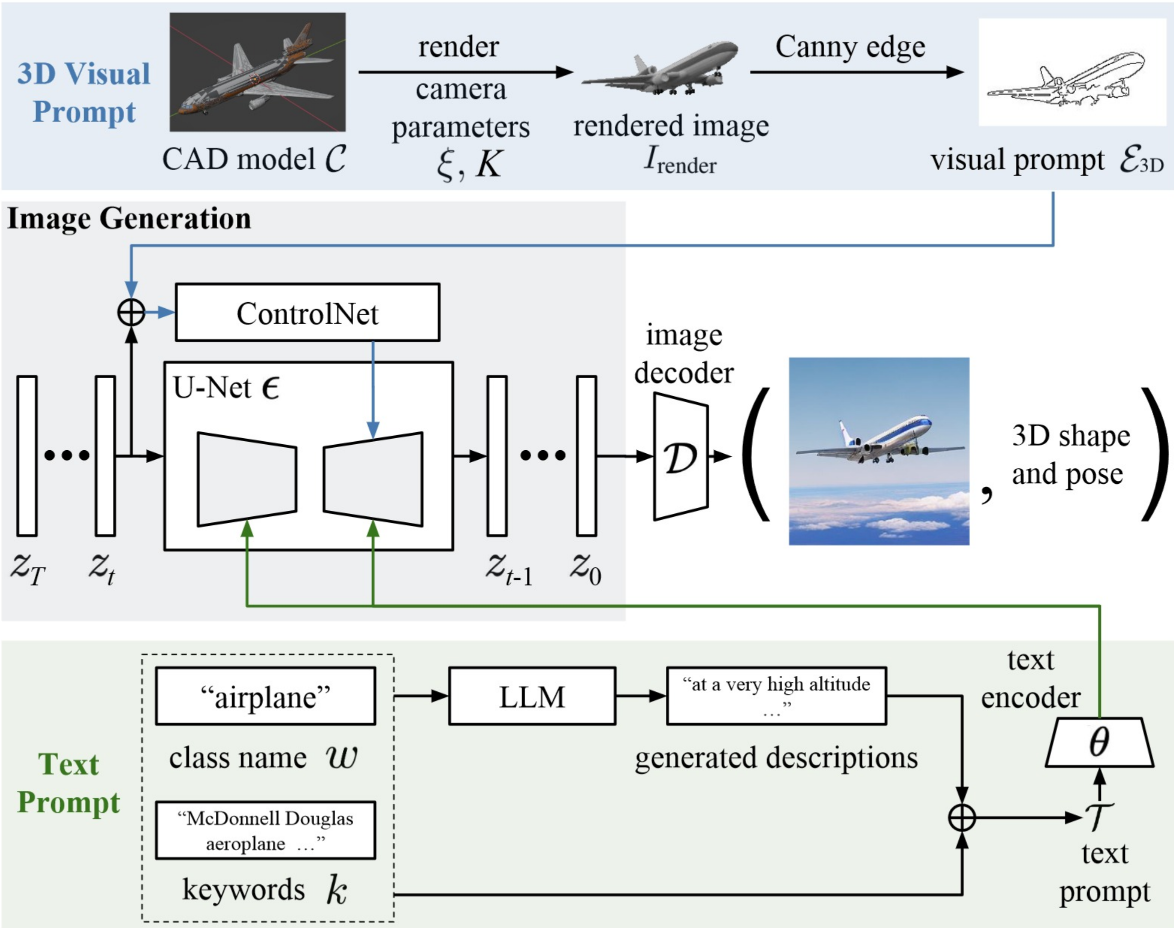
Adding 3D Geometry Control to Diffusion Models (5 5 6 8)

Authors: Wufei Ma, Qihao Liu, Jiahao Wang, Xiaoding Yuan, Angtian Wang, Yi Zhang, Zihao Xiao, Guofeng Zhang, Beijia Lu, Ruxiao Duan, Yongrui Qi, Adam Kortylewski, Yaoyao Liu, Alan Yuille

▶ **Abstract**

📄 Paper #object_generation #diffusion #ControlNet #spatial_information

Pipeline



- Get a CAD model from the 3D shape repository(e.g., ShapeNet and Objaverse)
- Render them from a variety of poses and viewing directions, then get the canny edge $\mathcal{E}_{3D}$ of the rendered image
- Using ControlNet to add 3D geometry information $\mathcal{E}_{3D}$

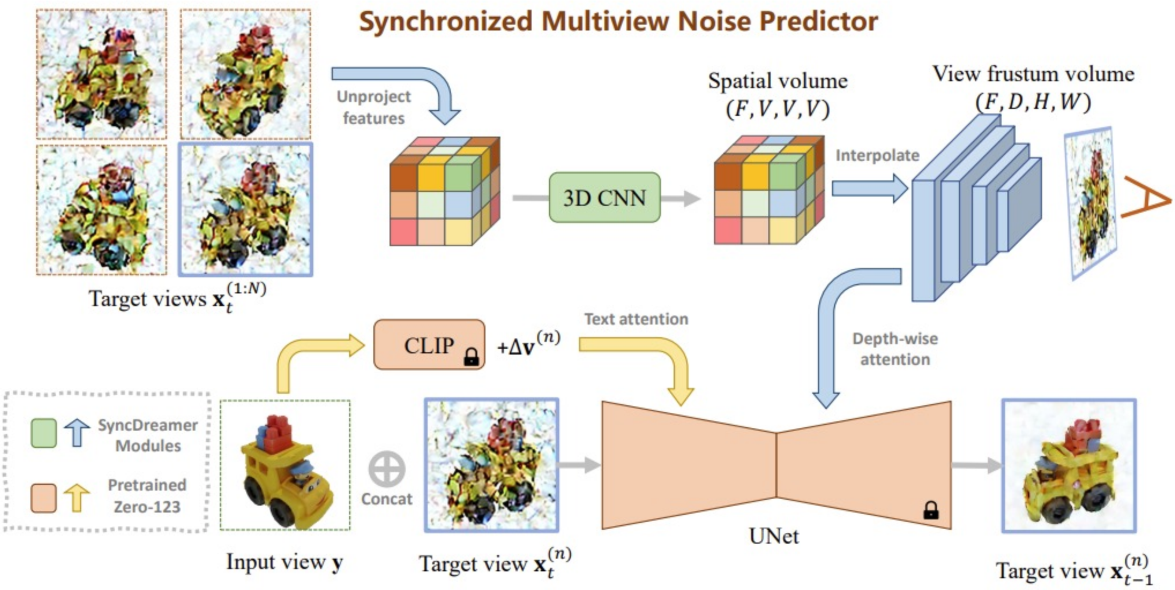
SyncDreamer: Generating Multiview-consistent Images from a Single-view Image (6 8 8 8 10)

Authors: Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, Wenping Wang

▶ **Abstract**

📄 Paper 🌐 Project 💻 Code #object_generation #diffusion #multi-view_diffusion #viewpoint_information #spatial_information

Pipeline



- Given the noisy 4 images from target views, we can get a spatial volume to represent these 4 images
- Pretrained zero123 model concatenates the input view y with the noisy target view ${x_t}^{(n)}$ as the input to UNet. The viewpoint information $\Delta v^{(n)}$ and CLIP feature as the condition
- Also, construct a view frustum volume of target view from the spitial volume to enforce consistency among multiple generated views.
  Ralated works:
- Zero-1-to-3: Zero-shot One Image to 3D Object(ICCV2023) Paper

Poster

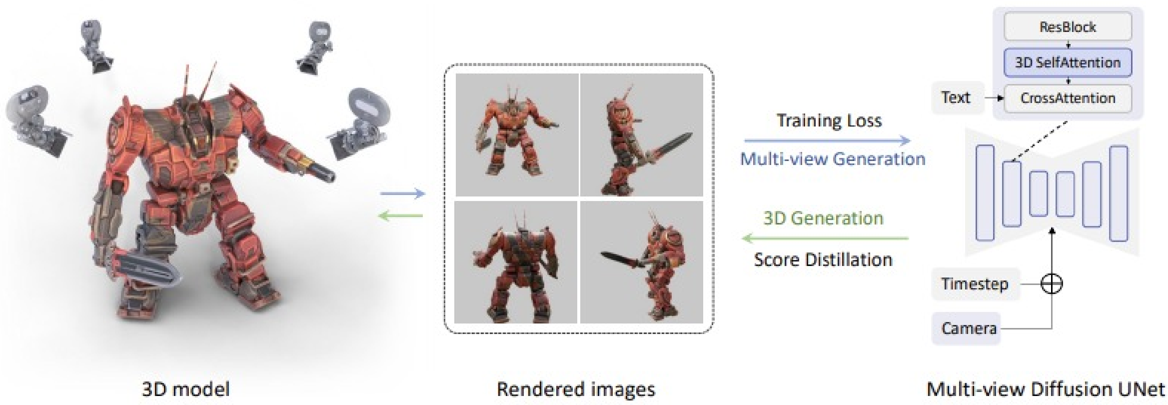
MVDream: Multi-view Diffusion for 3D Generation (6 6 6 8)

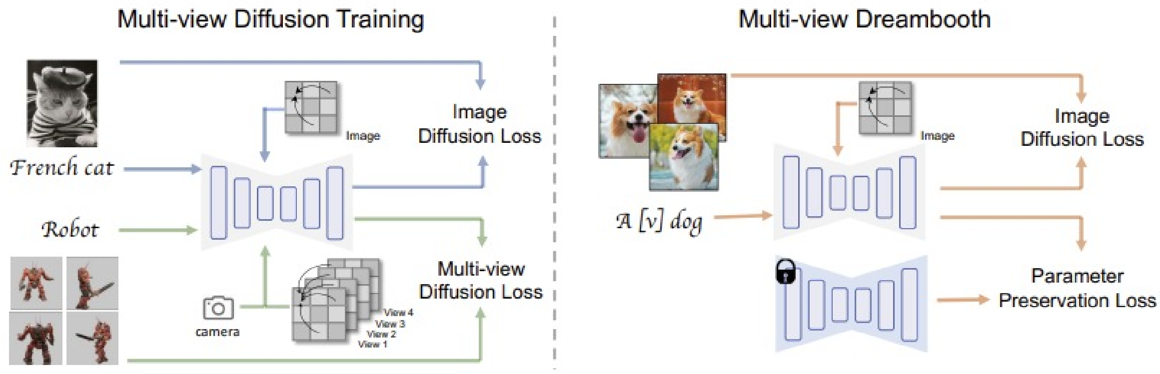
Authors: Yichun Shi, Peng Wang, Jianglong Ye, Long Mai, Kejie Li, Xiao Yang

▶ **Abstract**

📄 Paper 🌐 Project 💻 Code #object_generation #diffusion #multi-view_diffusion #viewpoint_information

Pipeline





- Connecting all different views together and doing 3D self-attention to generate consistent multi-view image at once
- Add camera embeddings to time embeddings as residuals
- Support multi-view Dreambooth

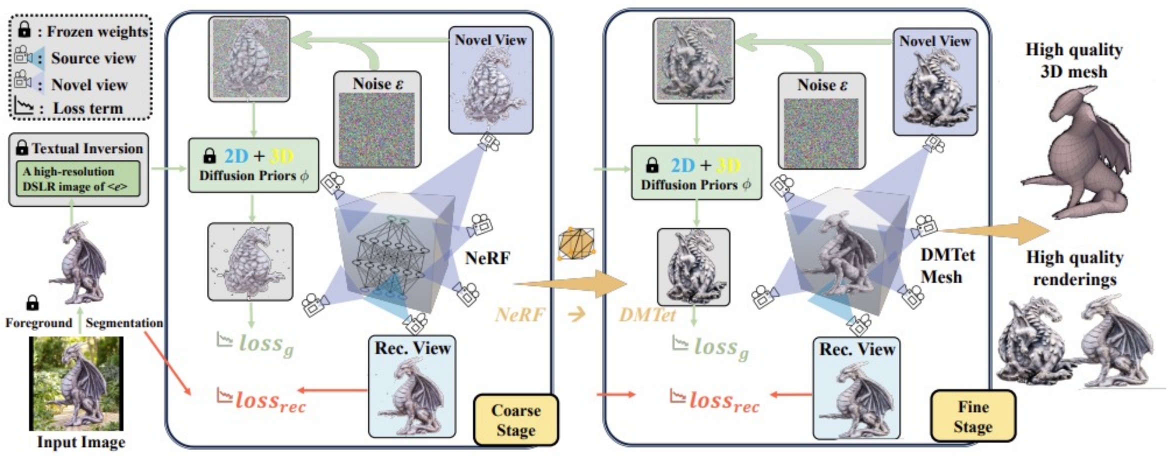
Magic123: One Image to High-Quality 3D Object Generation Using Both 2D and 3D Diffusion Priors (5 5 8 8)

Authors: Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skorokhodov, Peter Wonka, Sergey Tulyakov, Bernard Ghanem

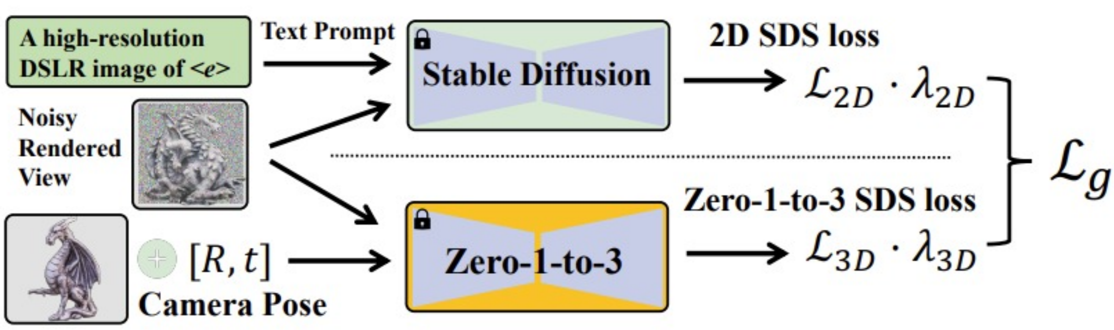
▶ **Abstract**

📄 Paper 🌐 Project 💻 Code #object_generation #diffusion #SDS #viewpoint_information #texture_refinement

Pipeline



- In coarse stage, do SDS on both 2D diffusion model(SD) and 3D diffusion model(zero123)



- In fine stage, do refinement on DMTet Mesh
  Ralated works:

- Zero-1-to-3: Zero-shot One Image to 3D Object(ICCV2023) Paper

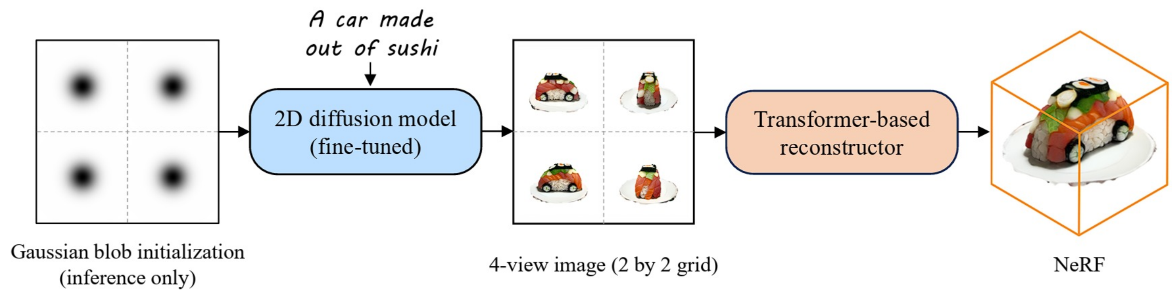
Instant3D: Fast Text-to-3D with Sparse-view Generation and Large Reconstruction Model(6 8 8)

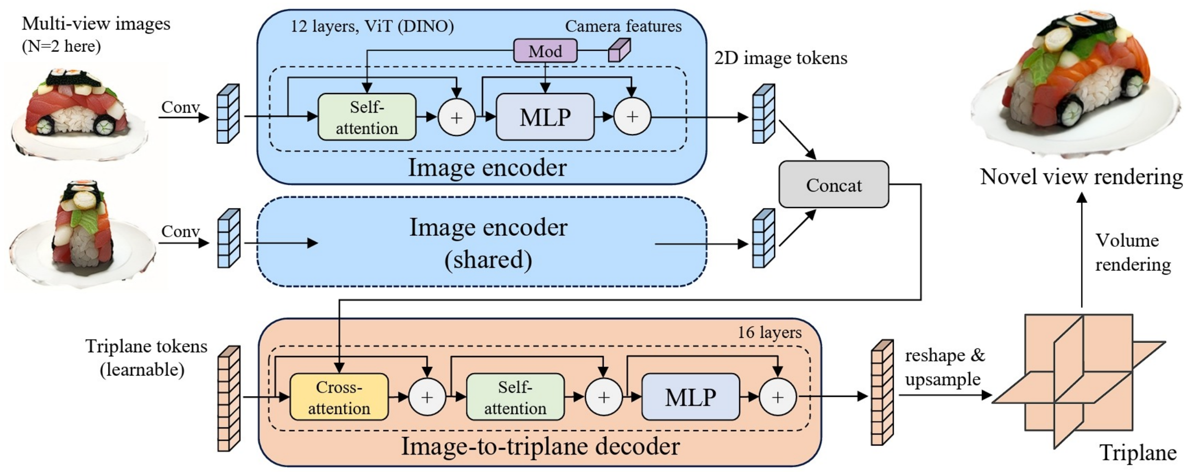
Authors: Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, FujunLuan, YinghaoXu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, Sai Bi

▶ **Abstract**

📄 Paper 🌐 Project #object_generation #diffusion #multi-view_diffusion #triplane #NeRF

Pipeline





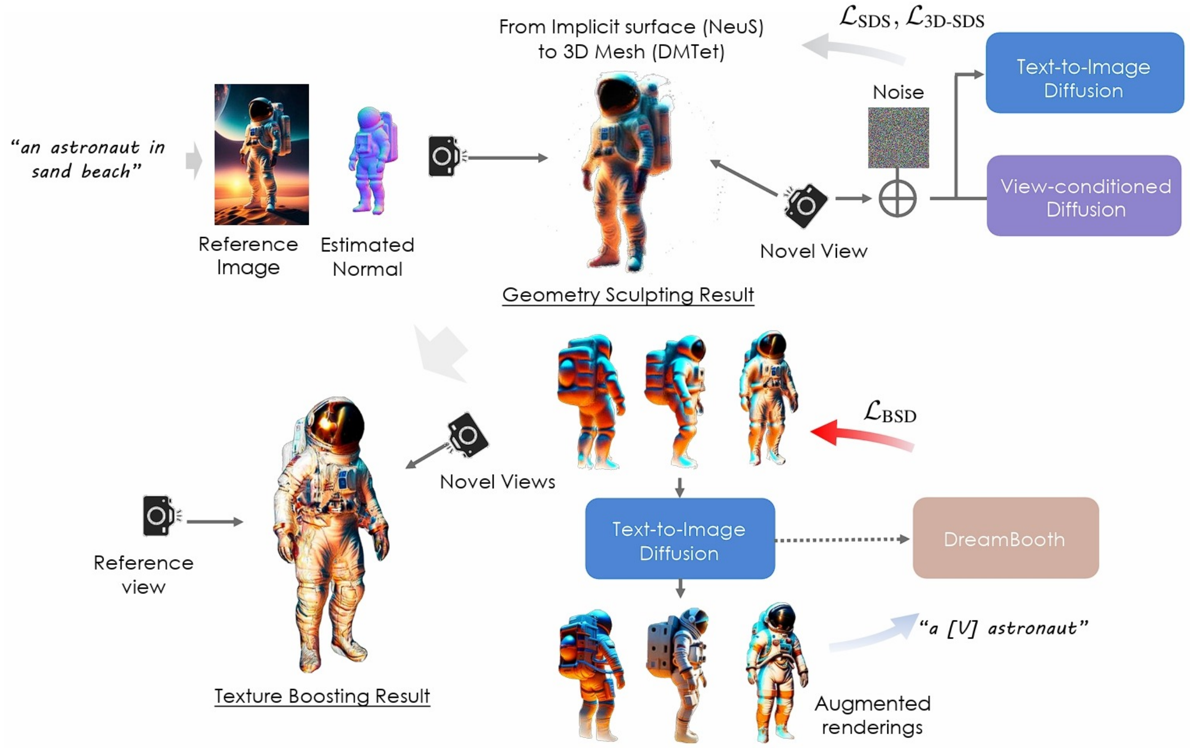
- By dividing a picture into four Gaussian blobs, the 2D diffusion model can generate pictures from 4 viewpoints at once.
- The architecture ot the Transformer-based reconstructor is just the same as LRM

DreamCraft3D: Hierarchical 3D Generation with Bootstrapped Diffusion Prior (5 6 6 8)

Authors: Jingxiang Sun, Bo Zhang, Ruizhi Shao, Lizhen Wang, Wen Liu, Zhenda Xie, Yebin Liu

▶ **Abstract**

📄 Paper 🌐 Project 💻 Code #object_generation #diffusion #viewpoint_information #SDS #texture_refinement



- In coarse stage, do SDS on both 2D diffusion model and 3D diffusion model(zero123)
- In refinement stage, finetune the diffusion model with the multi-view texture-augmented images, using DreamBooth. And use this finetuned model to gradually optimize the textures (Hope the score function of the optimized 3D scene match the score function of the DreamBooth model)
  $\nabla_\theta \mathcal{L}_{\text{BSD}}(\phi, g(\theta)) = \mathbb{E}_{t,\epsilon,c}[\omega(t)(\boldsymbol{\epsilon}_{\text{DreamBooth}}(\boldsymbol{x}_t; y, t, r_{t'}(\boldsymbol{x}), c) - \boldsymbol{\epsilon}_{\text{lora}}(\boldsymbol{x}_t; y, t, \boldsymbol{x}, c))\frac{\partial \boldsymbol{x}}{\partial \theta}]$
  Compare with the ProlificDreamer(Hope the score function of the optimized 3D scene match the score function of the pretrained model))
  $\nabla_\theta \mathcal{L}_{\text{VSD}}(\phi, g(\theta)) = \mathbb{E}_{t,\epsilon}[\omega(t)(\boldsymbol{\epsilon}_{\text{Pretrained}}(\boldsymbol{x}_t; y, t) - \boldsymbol{\epsilon}_{\text{lora}}(\boldsymbol{x}_t; y, t, x, c))\frac{\partial \boldsymbol{x}}{\partial \theta}]$

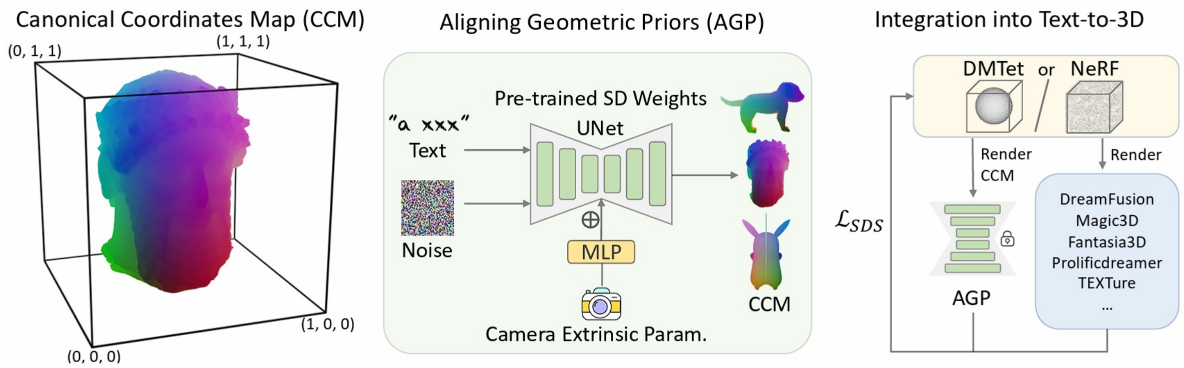
SWEETDREAMER: ALIGNING GEOMETRIC PRIORS IN 2D DIFFUSION FOR CONSISTENT TEXT-TO-3D (5 5 6 8)

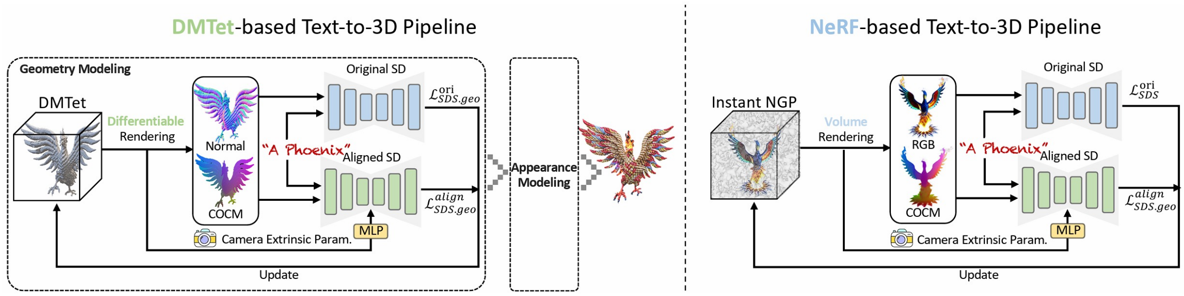
Authors: Weiyu Li, Rui Chen, Xuelin Chen, Ping Tan

▶ **Abstract**

📄 Paper 🌐 Project 💻 Code (not yet) #object_generation #diffusion #spatial_information #viewpoint_information #SDS

Pipeline





- In first stage, fine-tune a 2D diffusion model to generate viewpoint conditioned canonical coordinates maps(CCM)
- In the SDS stage, render both CCM and rgb image from the 3D representation(Nerf or DMTet), and do use both orginal and fine-tuned 2D diffusion models to optimize the 3D reprensentation.

TEXT-TO-3D WITH CLASSIFIER SCORE DISTILLATION (5 6 8 8)

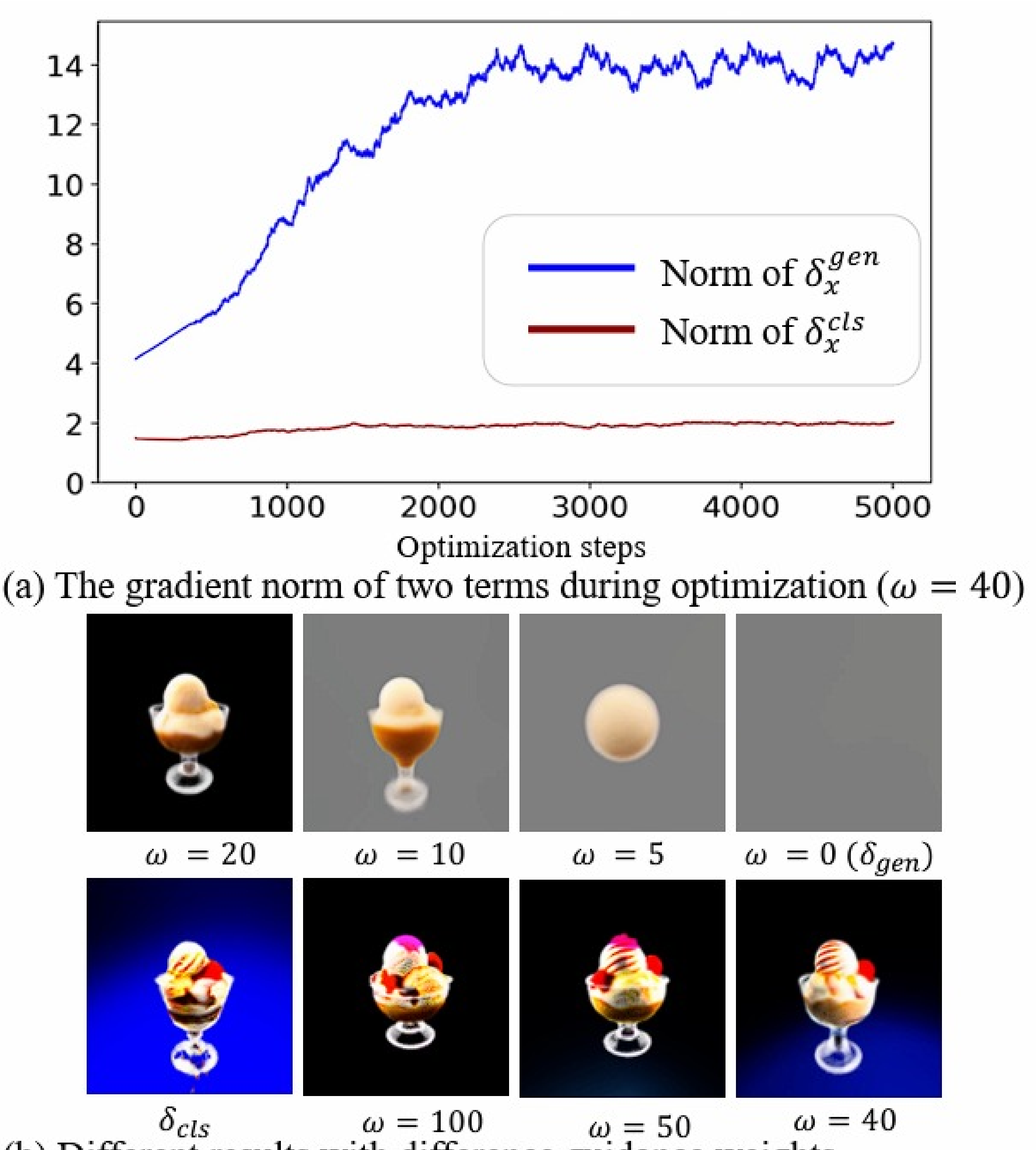
Authors: Xin Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Song-Hai Zhang, Xiaojuan Qi

▶ **Abstract**

📄 Paper 🌐 Project 💻 Code #object_generation #diffusion #SDS

Pipeline

In original SDS, the gradient is expressed as$\nabla_\theta \mathcal{L}_{\mathrm{SDS}} = \mathbb{E}_{t,\epsilon,\mathbf{c}}[w(t)(\epsilon_\phi(\mathbf{x}_t; y, t) - \epsilon)\frac{\partial \mathbf{x}}{\partial \theta}]$
And can be expressed as
$\epsilon_\phi(\mathbf{x}_t; y, t) - \epsilon = \delta_x(\mathbf{x}_t; y, t) = \underbrace{[\epsilon_\phi(\mathbf{x}_t; y, t) - \epsilon]}_{\delta_x^{\mathrm{gen}}} + \omega \cdot \underbrace{[\epsilon_\phi(\mathbf{x}_t; y, t) - \epsilon_\phi(\mathbf{x}_t; t)]}_{\delta_x^{\mathrm{cls}}}$



(a) The gradient norm of two terms during optimization ($\omega = 40$)
(b) Different results with difference guidance weights

The authors find that

- The gradient norm of the gen erative prior is several times larger than that of the classifier score in Fig(a)
- However, to generate high quality results, a large guidance weight must be set (e.g., ω = 40), as shown in Fig(b). When incorporating both components, the large guid ance weight actually causes the gradient from the classifier score to dominate the optimization di rection.
- Moreover, the optimization process fails whenrelying solely on the generative component, as indicated by setting ω = 0
  So they introduced to use classifier score sistillation(only considerδ_x^{cls}) to align the rendered noisy image and the text y.

DreamTime: An Improved Optimization Strategy for Diffusion-Guided 3D Generation (3 6 8 8)

Authors: Yukun Huang, Jianan Wang, Yukai Shi, Boshi Tang, Xianbiao Qi, Lei Zhang

▶ **Abstract**

📄 Paper #object_generation #diffusion #SDS

Pipeline

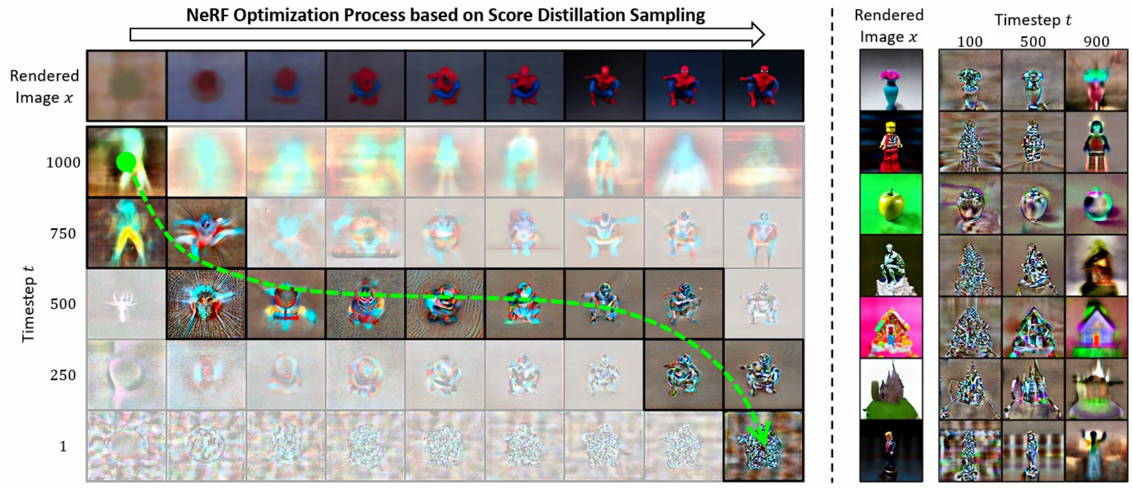


Figure 3: Visualization of SDS gradients under different timesteps t. (Left) Visualization of SDS gradients throughout the 3D model (in this case NeRF) optimization process, where the green curved arrow denotes the path for more informative gradient directions as NeRF optimization progresses. It can be observed that a non-increasing timestep t is more suitable for the SDS optimization process. (Right) We provide more examples to illustrate the effects of timestep t on SDS gradients. Small t provides guidance on local details, while large t is responsible for global structure.