中山大學
SUN YAT-SEN UNIVERSITY

基于SD/SVD的3D重建和NVS

李奕清

目录

3DGS/NeRF + SD/SVD

1. Taming Video Diffusion Prior with Scene-Grounding Guidance for 3D Gaussian Splatting from Sparse Inputs (Guidevd-3DGS)
2. Difix3D+: Improving 3D Reconstructions with Single-Step Diffusion Model

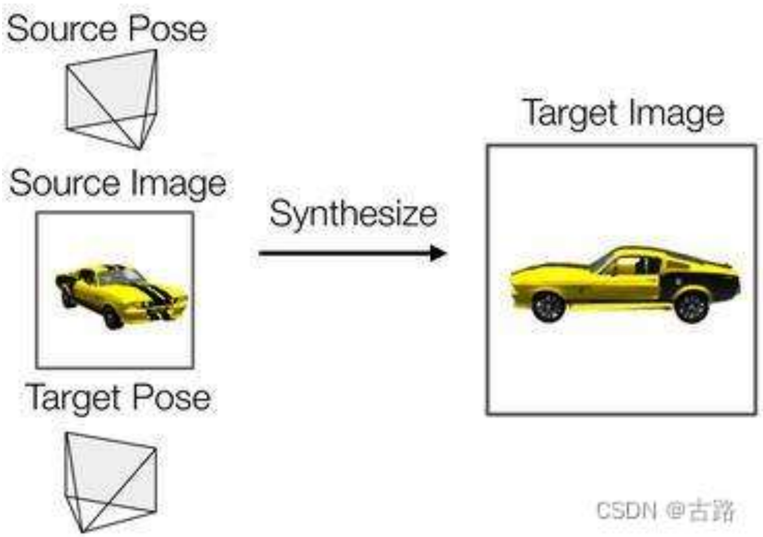
[1] Taming Video Diffusion Prior with Scene-Grounding Guidance for 3D Gaussian Splatting from Sparse Inputs . CVPR, 2025
[2] Difix3D+: Improving 3D Reconstructions with Single-Step Diffusion Model. CVPR, 2025

中山大學 SUN YAT-SEN UNIVERSITY

任务定义

- 3D重建算法：从多张二维图像恢复出场景的三维几何模型。传统方法比如摄影测量，新方法NeRF，3DGS
- 新视角合成(Novel View Synthesis, NVS)：给定 源图像 及 源相机位姿 ， 渲染生成目标相机位姿对应的图片。
- 常见的NVS流程：1、重建：从已有视角进行3D重建，2、渲染：根据重建场景渲染出新视角的图片。

3D重建



新视角合成

任务定义

源图像

新视角合成

背景

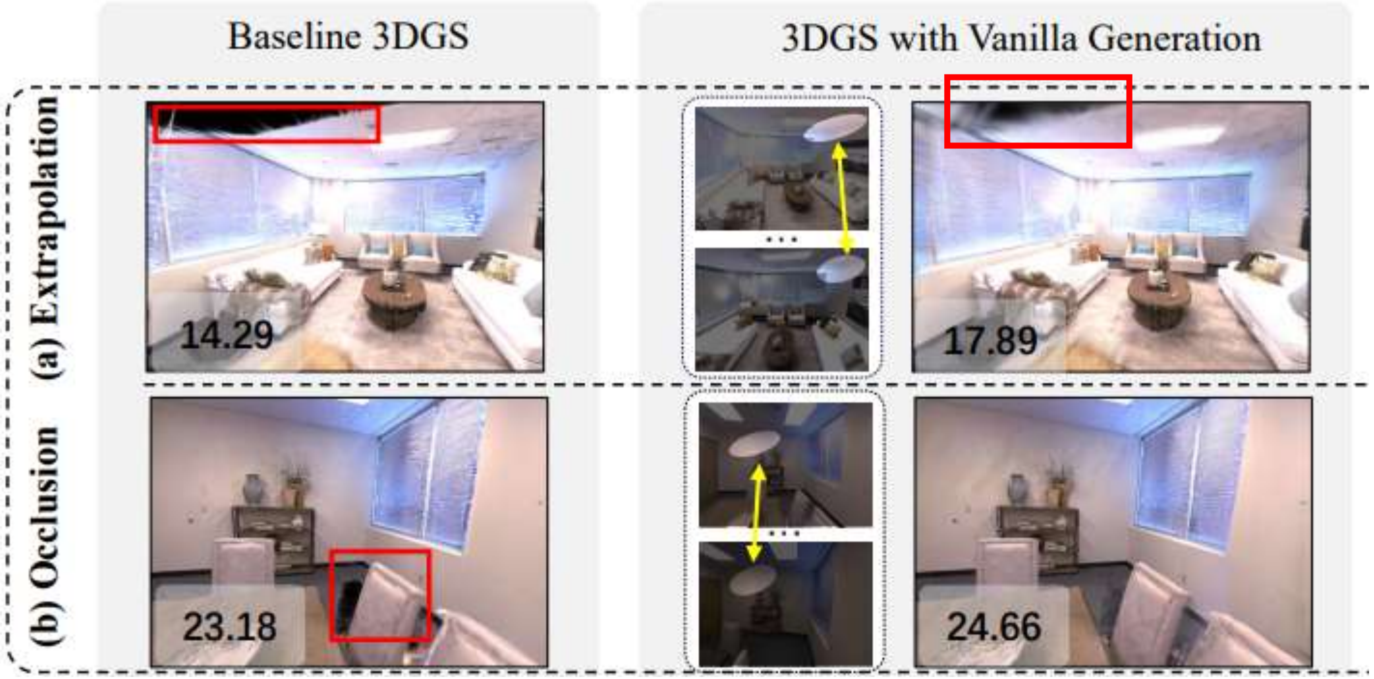
- 现有方法普遍采用face-forward的视角设置，过度简化了现实世界的**稀疏输入**建模，忽略了两个关键问题：

1. **外推**：即使稀疏输入尽可能多地覆盖了场景，仍可能存在视野之外的区域。
2. **遮挡**：当新视角与训练输入视角略有偏差时，遮挡问题频繁出现。

动机

- 视频扩散模型（SVD）可以为**不可见区域**提供信息。但直接使用可能导致性能下降。其主要原因是生成序列存在**多视角不一致性**，具体表现为：

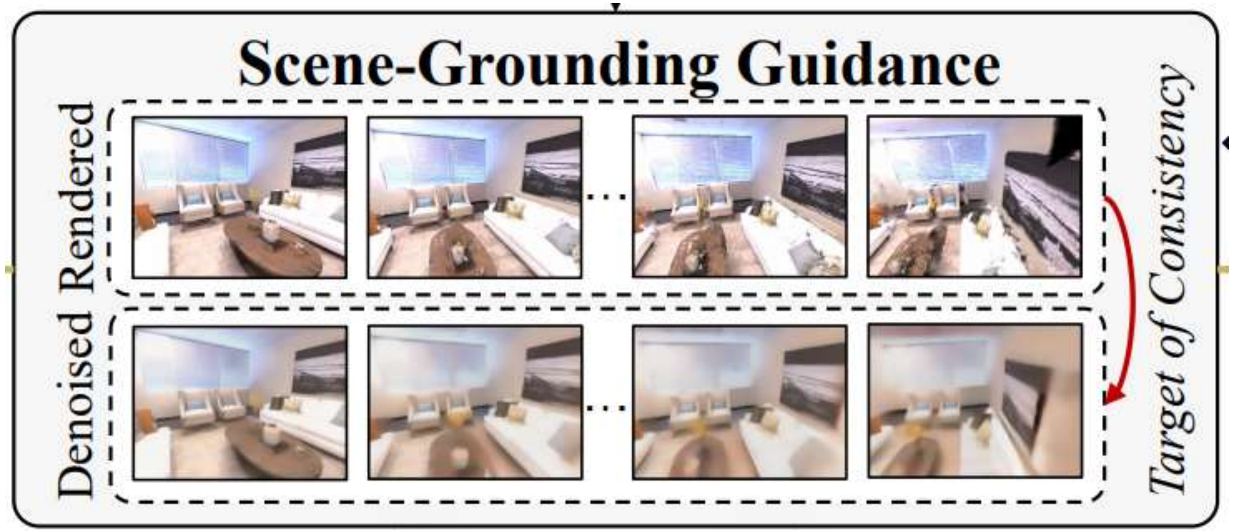
1. **帧间外观不一致：**同一序列中的不同帧可能存在外观差异。
2. **虚假元素**：生成的序列可能包含场景中并不存在的元素。



方法

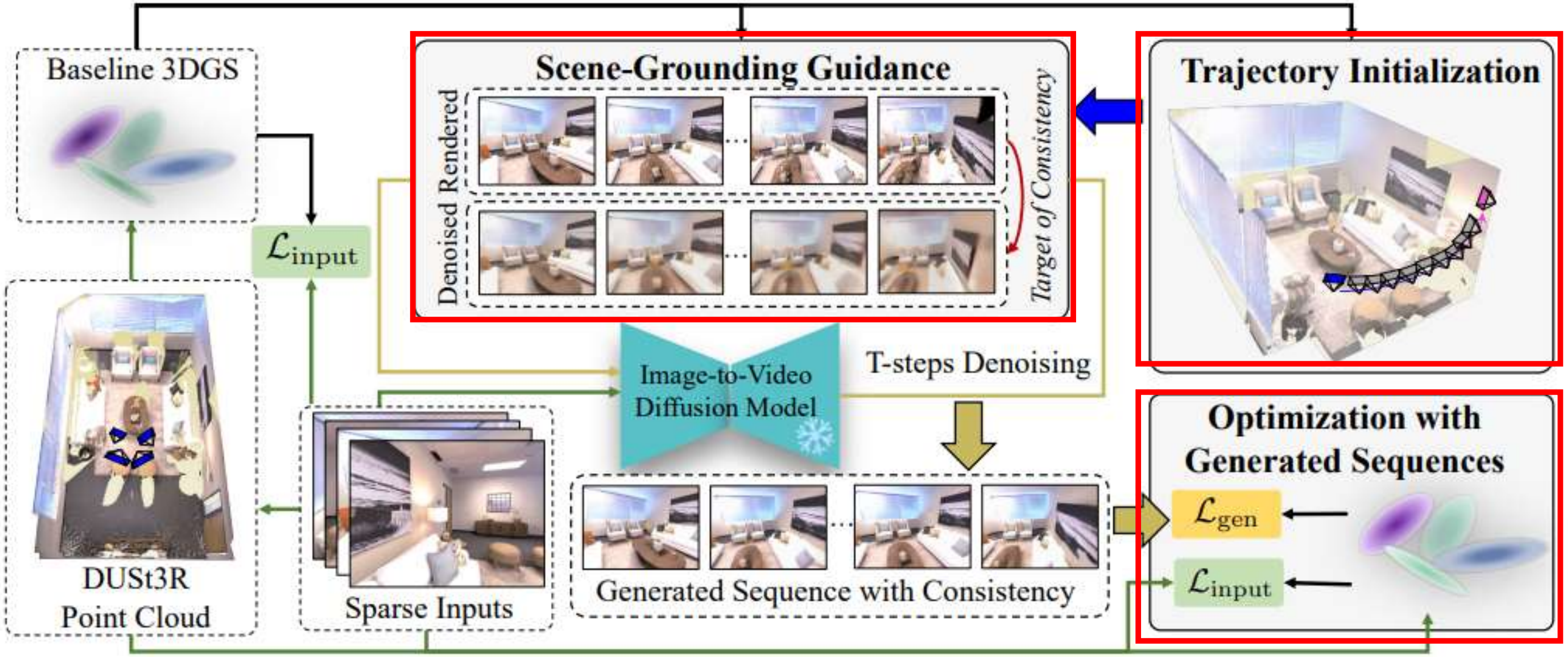
- 受无需训练的方法启发，提出了**场景锚定引导**策略，以确保生成序列的一致性。具体而言，在每一步去噪过程中，生成的噪声序列会从**渲染序列**中接收**梯度引导**。为什么能采用**渲染序列**进行一致性约束：

1. **相邻帧内容一致**：由于相机运动范围有限，渲染序列中的相邻帧具有高度一致的外观。
2. **渲染序列提供场景锚定**：可引导扩散模型避免生成场景中不存在的元素。



方法

1. **场景锚定引导**(Scene-Grounding Guidance)，无需训练和微调
2. 轨迹初始化策略，有效覆盖un-seen区域和遮挡区域
3. 基于生成序列的3DGS优化策略



场景锚定引导

分数函数：由Unet估计

扩散模型去噪公式：$\mathbf{x}_{t-1} = (1+\beta_t/2)\mathbf{x}_t + \beta_t \boxed{\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t)} + \sqrt{\beta_t}\mathbf{z} \quad (2)$

受方法[1,2]启发，添加一致性目标 Q 来引导去噪

$$p(\mathbf{x}_t|\mathbf{c}) = \frac{p(\mathbf{c}|\mathbf{x}_t)p(\mathbf{x}_t)}{p(\mathbf{c})},$$

贝叶斯展开

一致性约束项

$$\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathcal{Q}) = \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t) + \boxed{\nabla_{\mathbf{x}_t} \log p(\mathcal{Q}|\mathbf{x}_t)}, \quad (3)$$

根据[2]，能量函数 $p(\mathbf{c}|\mathbf{x}_t) = \frac{\exp\{-\lambda\mathcal{E}(\mathbf{c}, \mathbf{x}_t)\}}{Z}$，进一步推导一致性约束

$$\nabla_{\mathbf{x}_t} \log p(\mathcal{Q}|\mathbf{x}_t) \propto -\nabla_{\mathbf{x}_t} \mathcal{L}(\mathcal{Q}, \mathbf{x}_t), \quad (4)$$

如何定义Q？不像[1,2]使用额外的模型，而是使用3DGS渲染序列S作为Q，好处是：不用引入额外模型，不要微调就能提供**场景锚定**

$$\begin{aligned}\mathcal{L}(\mathbf{S}, \mathbf{M}, \mathbf{X}_{0|t}) &= \|\mathbf{M} \odot (\mathbf{S} - \mathbf{X}_{0|t})\|_1 \\ &+ \lambda_{\mathrm{perc}} \mathcal{L}_{\mathrm{perc}}(\mathbf{M} \odot \mathbf{S}, \mathbf{M} \odot \mathbf{X}_{0|t}),\end{aligned} \quad (6)$$

[1] Universal guidance for diffusion models. CPVR, 2023.
[2] Training-free energy-guided conditional diffusion model. ICCV, 2023.

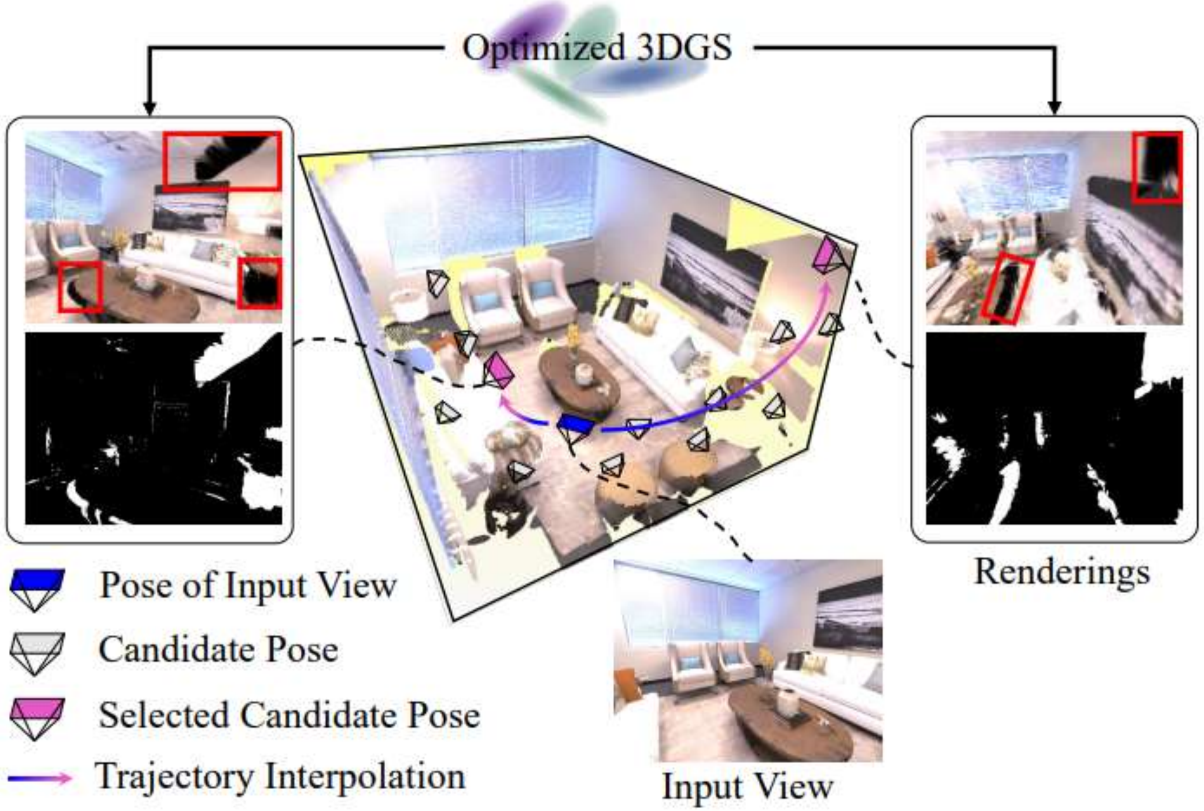
场景锚定引导

Algorithm 1 Generation with Scene-Grounding Guidance

Function GENERATOR($\mathcal{R}, I, \{\phi_j\}_{j=1}^{L}$)
Input: Optimized 3DGS model $\mathcal{R}$, input image I, camera trajectory of a sequence $\{\phi_j\}_{j=1}^{L}$.
Given: Latent image-to-video diffusion model ϵ_θ, VAE decoder $\mathcal{D}$, pre-defined $\beta_t, \bar{\alpha}_t$ and guidance scale γ_t.
Abbreviate $\epsilon_\theta(\mathbf{x}_t, t, I, \{\phi_j\}_{j=1}^{L})$ **as** $\epsilon_\theta(\mathbf{x}_t, t)$
$\mathbf{S}, \mathbf{M} = \text{rasterize}(\{\phi_j\}_{j=1}^{L}, \mathcal{R})$ ▷ Eq. (1)& (5)
$\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$
for $t = T, \ldots, 1$ **do**
  $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
  $\hat{\mathbf{x}}_{t-1} = (1 + \frac{1}{2}\beta_t)\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t) + \sqrt{\beta_t}\mathbf{z}$
  $\mathbf{x}_{0|t} = \frac{1}{\sqrt{\bar{\alpha}_t}}(\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\mathbf{x}_t, t))$
  $\mathbf{X}_{0|t} = \mathcal{D}(\mathbf{x}_{0|t})$
  $\mathbf{g}_t = \nabla_{\mathbf{x}_t}\mathcal{L}(\mathbf{S}, \mathbf{M}, \mathbf{X}_{0|t})$ ▷ Eq. (6)
  $\mathbf{x}_{t-1} = \hat{\mathbf{x}}_{t-1} - \gamma_t\mathbf{g}_t$ ▷ Eq. (2)& (4)
end for
return $\mathcal{D}(\mathbf{x}_0)$

轨迹初始化策略

对每个稀疏输入视角，在其周围采样多个候选相机姿态，并使用3DGS渲染。选择在渲染图片中存在显著黑洞（未覆盖区域）的候选姿态，并插值生成完整的相机轨迹：$\Phi = \{\{\phi_j^{(i,c)}\}_{j=1}^{L} | i, c\}$,



基于生成序列的3DGS优化方案

1. 训练一个初始的3DGS。
2. 进行轨迹初始化，构建轨迹池。
3. 迭代过程，每隔固定步数生成新的序列，并将其用于优化。
4. 结合输入视图和生成视图的损失函数，更新3DGS。

损失函数

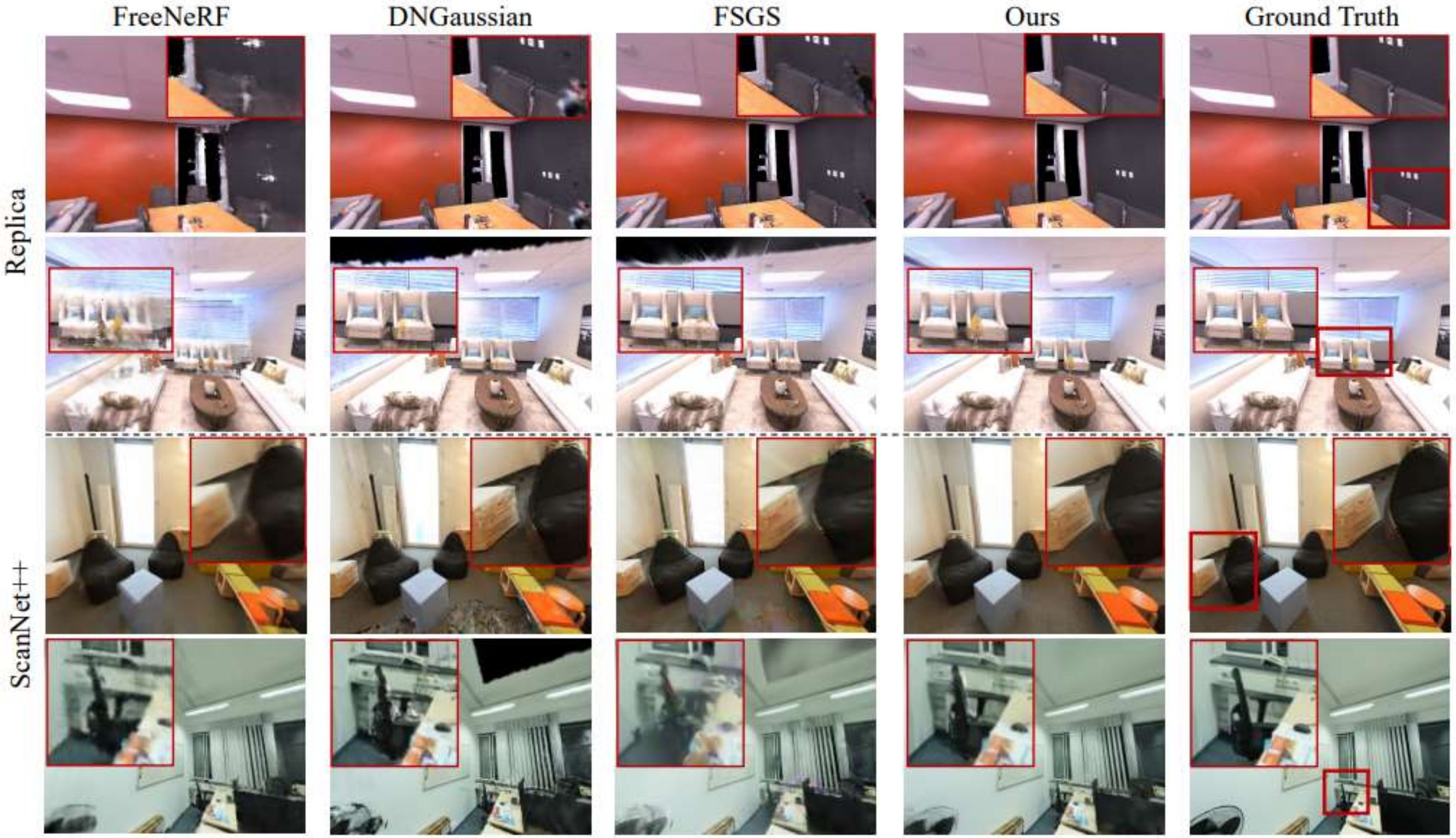
$$\mathcal{L}^{\text{input}} = (1-\lambda)\mathcal{L}_1(C_i, C_i^{\text{gt}}) + \lambda\mathcal{L}_{\text{D-SSIM}}(C_i, C_i^{\text{gt}}), \quad (8)$$

$$\mathcal{L}^{\text{gen}} = \lambda_{\text{gen1}}\mathcal{L}_1(C_j, S_j) + \lambda_{\text{gen2}}\mathcal{L}_{\text{perc}}(C_j, S_j), \quad (9)$$

Algorithm 2 3DGS Optimization with Generation

Input: Sparse inputs of N images $\{C_i^{\text{gt}}, \varphi_i\}_{i=1}^N$.
Given: Number of iterations N_{iter}, generation interval N_{gen}, ratio of samples from other sequences η.
Variable: Global list of generated views $\mathbf{G} = []$.
Baseline 3DGS model optimization $\Rightarrow \mathcal{R}$
Trajectory initialization $\Rightarrow \Phi$ ▷ Eq. (7)
for $t = 0, \ldots, N_{\text{iter}} - 1$ **do**
  If t % N_{gen} = 0 **then**
    Sample an input view I
    Sample a trajectory around I from $\Phi \Rightarrow \{\phi_j\}_{j=1}^L$
    $\mathbf{S}$ = GENERATOR$(\mathcal{R}, I, \{\phi_j\}_{j=1}^L)$
    Append $\mathbf{S}$ to $\mathbf{G}$
  End If
  Sample an input view to get $\mathcal{L}^{\text{input}}$ ▷ Eq. (8)
  If rand() $\geq \eta$ **then**
    Sample a generated view from $\mathbf{S}$
  Else Sample a generated view from $\mathbf{G}$
  End If
  Use the generated view to get $\mathcal{L}^{\text{gen}}$ ▷ Eq. (9)
  $(\mathcal{L}^{\text{input}} + \mathcal{L}^{\text{gen}})$.backward( )
  # Densification and opacity reset
end for

实验结果



ScanNet++数据集和Replica数据集，6个视角输入

实验结果

Method	Replica [44]			ScanNet++ [58]		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Mip-NeRF [2]	18.12	0.707	0.391	19.58	0.755	0.389
InfoNeRF [20]	13.07	0.598	0.552	14.54	0.646	0.495
DietNeRF [16]	18.99	0.676	0.444	19.76	0.719	0.431
FreeNeRF [56]	20.99	0.765	0.324	20.17	0.756	0.368
S^{3}NeRF [68]	22.54	0.800	0.287	22.21	0.787	0.364
3DGS‡ [19]	22.80	0.818	0.179	21.41	0.817	0.211
DNGaussian [21]	17.63	0.718	0.435	19.01	0.754	0.367
DNGaussian‡ [21]	22.71	0.821	0.189	20.68	0.788	0.281
FSGS [69]	20.22	0.760	0.304	17.95	0.730	0.373
FSGS‡ [69]	22.99	0.833	0.205	21.23	0.813	0.257
Ours	26.35	0.872	0.122	23.89	0.850	0.182

定量结果

(a)	*Gen.*	*Guide.*	*Traj.*	Full Image			Observable Regions		
				PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Baseline 3DGS				22.80	0.818	0.179	25.45	0.860	0.129
w/ Vanilla Generation	✓			23.69	0.840	0.160	25.00	0.870	0.119
w/ Guided Generation	✓	✓		25.03	0.852	0.139	26.52	0.881	0.101
w/ Guided Generation&Traj.	✓	✓	✓	**25.58**	**0.859**	**0.138**	**26.53**	**0.883**	**0.100**

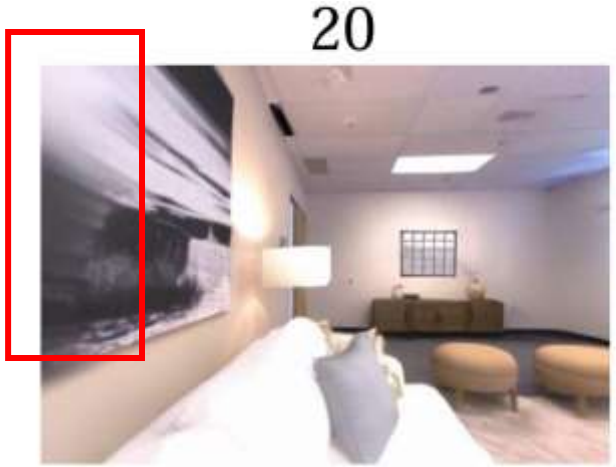
(b)	PSNR↑	SSIM↑	LPIPS↓
Baseline 3DGS	22.80	0.818	0.179
w/ Guided Generation&Traj.	25.58	0.859	0.138
w/ perceptual loss	**26.35**	**0.872**	**0.122**
w/o local sampling	26.28	0.871	0.127
w/o global list	26.01	0.867	0.122

消融结果

Guidevd-3DGS

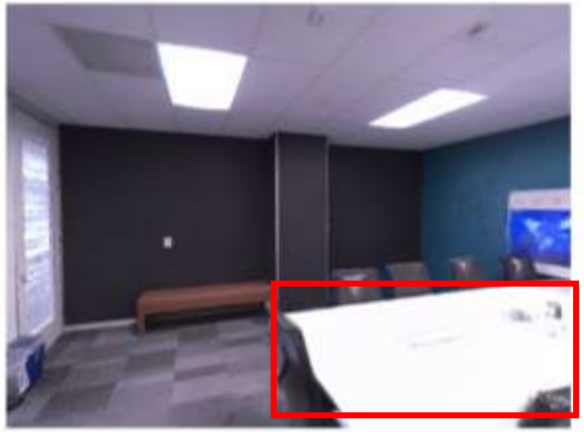
不足

1. 图片内容过平滑，分辨率不高，320×512
2. **多视角一致性**和**图片保真度**实质上还是缺少保证，依赖于SVD的能力

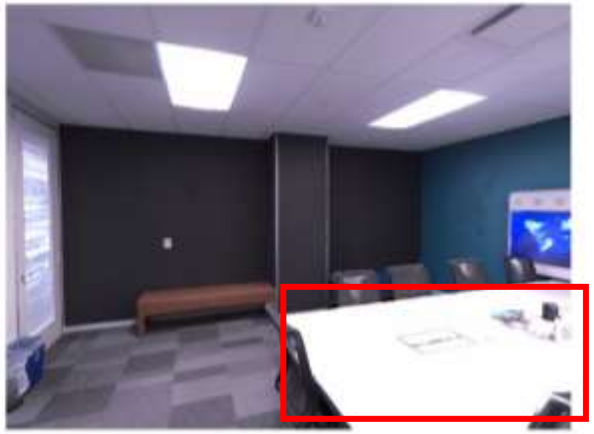


Guidedvd-3dgs

GT



Guidedvd-3dgs

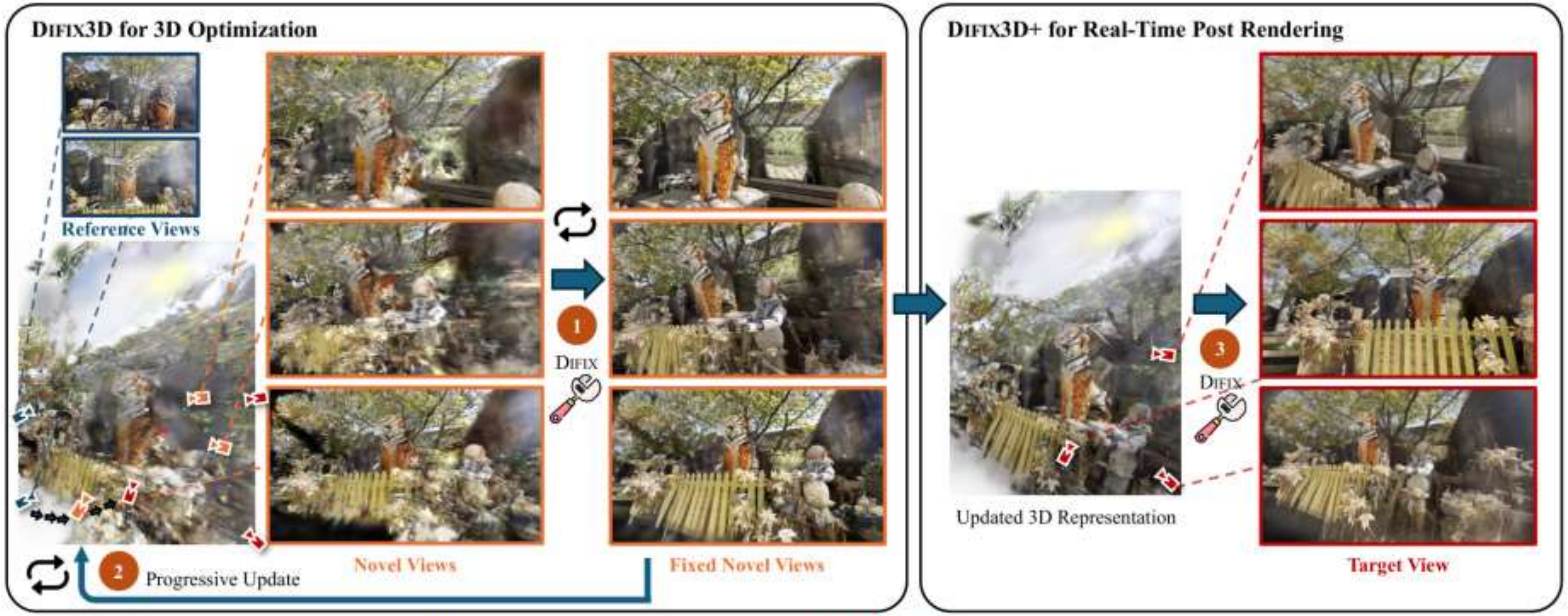


GT


Enhancing NeRF and 3DGS
Nerfacto
Nerfacto
Nerfacto
Reference Views
Target Views
DIFIX
Novel Views
3DGS
3DGS
3DGS
Target Views
Enhanced Target Views
Step 3: DIFIX acts as a real-time neural enhancer, further improving the quality of the rendered novel views.

方法

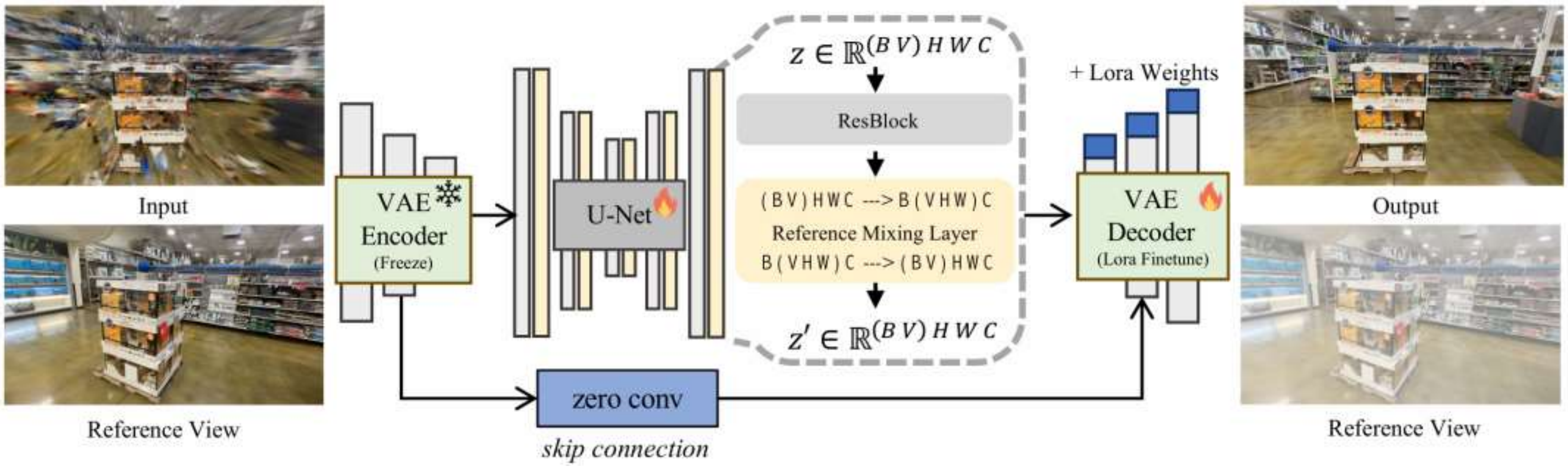
- 通过单步扩散模型SD-Turbo，增强、去除un-seen视角欠拟合造成的伪影
- 可以在两个阶段起作用：重建阶段用来去除伪影，增强图片。后处理阶段作为一个实时增强器



Blue Cameras: Training Views; Red Cameras: Target Views;
Orange Cameras: Intermediate Novel views along the progressive 3D updating trajectory (Sec. 4.2).

方法

- 输入：低质图像和参考图像，输出干净图像。
- 修改了SD-Turbo中的自注意力层，将低质图像和参考图像拼接，考虑到相互的图像内容。
- VAE的Decoder也进行LoRA的微调



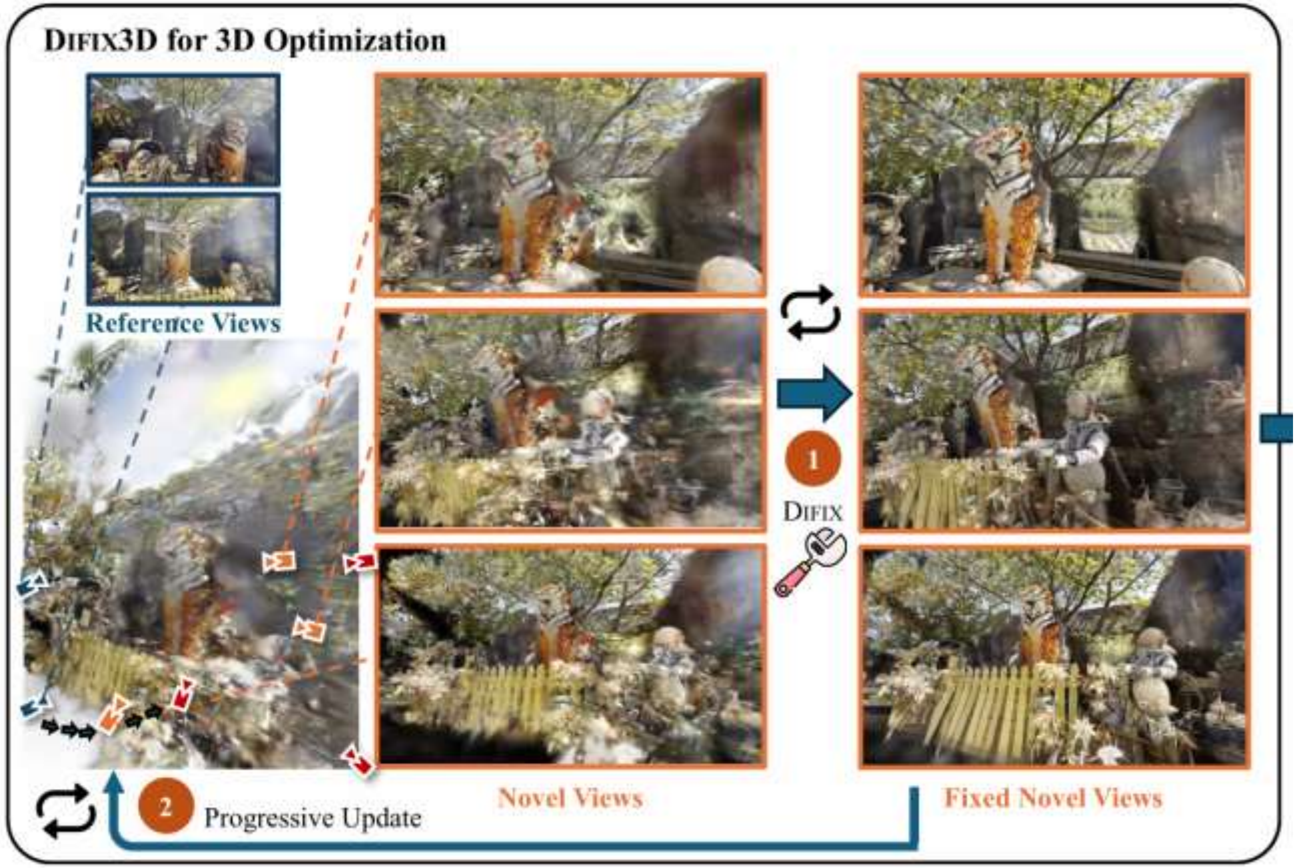
方法

- 微调时，输入不是随机高斯噪声，而是退化的渲染图像。
- 发现**退化图像分布**与原始扩散模型在特定噪声水平 τ = 200下训练的噪声图像的分布相似。

τ	1000	800	600	400	200	10
PSNR	12.18	13.63	15.64	17.05	**17.73**	17.72
SSIM	0.4521	0.5263	0.6129	0.6618	**0.6814**	0.6752

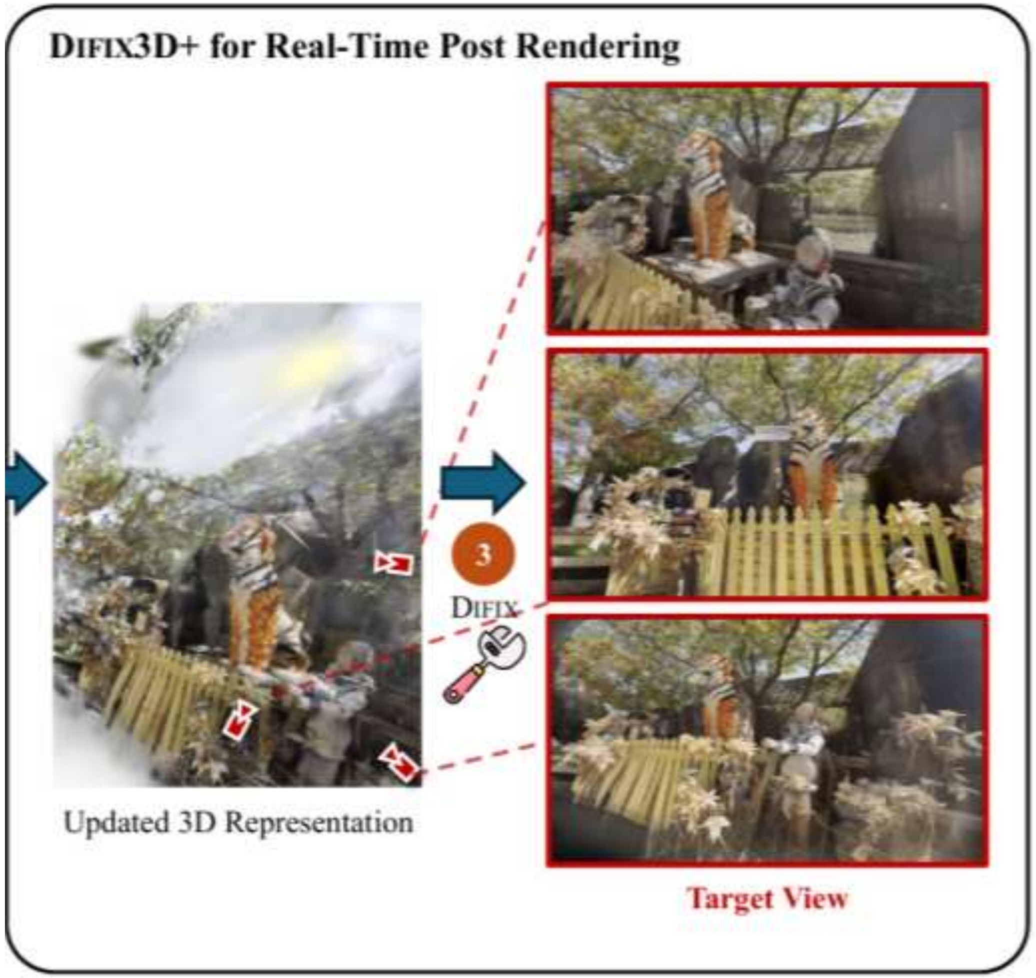
DIFIX3D：渐进式更新Progressive 3D Updates

1. 使用参考视图优化 3D 表示
2. 每隔n次迭代，将GT相机姿态向目标视图扰动 ∇，渲染新视角，使用DIFIX进行Refine
3. 优化后的图像添加到训练集，再进行n次迭代
4. 通过逐步扰动相机姿态、优化新视角和更新训练集，逐渐提高 3D 一致性



DIFIX3D+：实时后渲染处理

1. 进一步增强新视角，在推理时使用 DIFIX 作为后处理，有效地去除残留的伪影
2. 由于 DIFIX 是一个单步模型，额外的渲染时间在 A100上仅需 76 毫秒，比标准的多步去噪扩散模型快 10 倍以上



SUN YAT-SEN UNIVERSITY

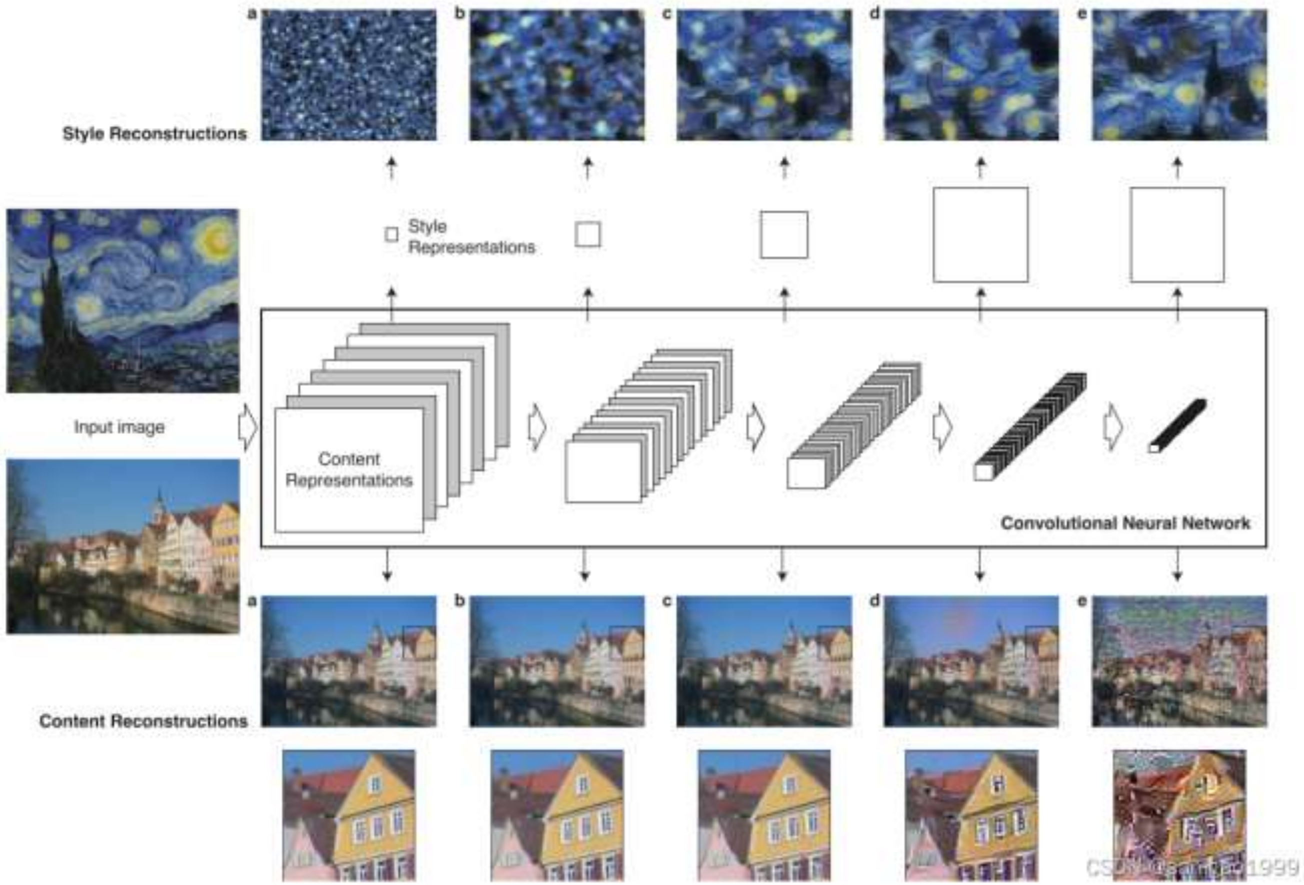
损失函数

- 矩阵风格损失：一般用在风格迁移任务。对齐CNN中的深层特征

$$\mathcal{L}_{\text{Gram}} = \frac{1}{L}\sum_{l=1}^{L}\beta_l \left\| G_l(\hat{I}) - G_l(I) \right\|_2,$$

- L2重建损失 + 感知损失 + 矩阵风格损失

$$\mathcal{L} = \mathcal{L}_{\text{Recon}} + \mathcal{L}_{\text{LPIPS}} + 0.5\mathcal{L}_{\text{Gram}}.$$



实验结果

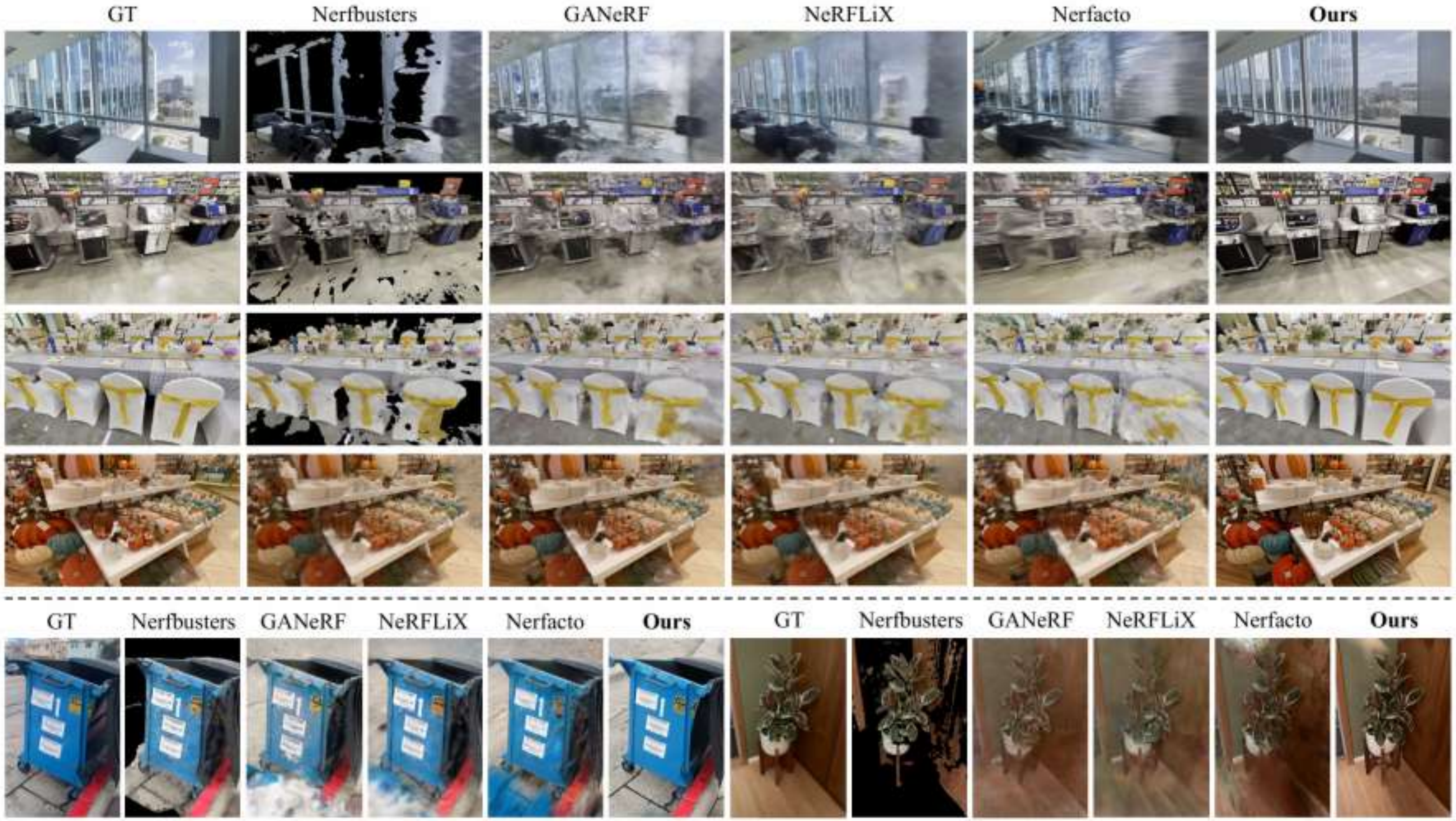


Figure 5. **In-the-wild artifact removal.** We show comparisons on held-out scenes from the DL3DV dataset [23] (*top*, above the dashed line) and the Nerfbusters [70] dataset (*bottom*). DIFIX3D+ corrects significantly more artifacts that other methods.

实验结果

Method	Nerfbusters Dataset PSNR↑	SSIM↑	LPIPS↓	FID↓	DL3DV Dataset PSNR↑	SSIM↑	LPIPS↓	FID↓
Nerfbusters [70]	17.72	0.6467	0.3521	116.83	17.45	0.6057	0.3702	96.61
GANeRF [46]	17.42	0.6113	0.3539	115.60	17.54	0.6099	0.3420	81.44
NeRFLiX [88]	17.91	<u>0.6560</u>	0.3458	113.59	17.56	<u>0.6104</u>	0.3588	80.65
Nerfacto [58]	17.29	0.6214	0.4021	134.65	17.16	0.5805	0.4303	112.30
DIFIX3D (Nerfacto)	<u>18.08</u>	0.6533	<u>0.3277</u>	<u>63.77</u>	<u>17.80</u>	0.5964	<u>0.3271</u>	<u>50.79</u>
DIFIX3D+ (Nerfacto)	**18.32**	**0.6623**	**0.2789**	**49.44**	**17.82**	**0.6127**	**0.2828**	**41.77**
3DGS [20]	17.66	0.6780	0.3265	113.84	17.18	0.5877	0.3835	107.23
DIFIX3D (3DGS)	<u>18.14</u>	<u>0.6821</u>	<u>0.2836</u>	<u>51.34</u>	<u>17.80</u>	<u>0.5983</u>	<u>0.3142</u>	<u>50.45</u>
DIFIX3D+ (3DGS)	**18.51**	**0.6858**	**0.2637**	**41.77**	**17.99**	**0.6015**	**0.2932**	**40.86**

Table 2. **Quantitative comparison on Nerfbusters and DL3DV datasets**. The best result is highlighted in **bold**, and the second-best is <u>underlined</u>.

Method	τ	SD Turbo Pretrain.	Gram	Ref	LPIPS↓	FID↓
pix2pix-Turbo	1000	✓			0.3810	108.86
DIFIX	200	✓			0.3190	61.80
DIFIX	200	✓	✓		0.3064	55.45
DIFIX	200	✓	✓	✓	**0.2996**	**47.87**

Table 5. **Ablation study of DIFIX components on Nerfbusters dataset**. Reducing the noise level, conditioning on reference views, and incorporating Gram loss improve our model.

不足

1. **多视角一致性**和**图片保真度**实质上还是缺少保证，依赖于SD注意力+ 3D表征的能力（例如，幻觉内容、几何结构偏移、直线扭曲）
2. 颜色色调会发生轻微改变（一般容易更深）

GT Difix3D+ GT Difix3D+

中山大學
SUN YAT-SEN UNIVERSITY

谢谢