Taming Uncertainty in Sparse-view Generalizable NeRF via Indirect Diffusion Guidance

Yaokun Li, Chao Gou, Guang Tan
Shenzhen Campus of Sun Yat-sen University
liyk58@mail2.sysu.edu.cn, {gouchao,tanguan}@mail.sysu.edu.cn

Abstract

Neural Radiance Fields (NeRF) have demonstrated effectiveness in synthesizing novel views. However, their reliance on dense inputs and scene-specific optimization has limited their broader applicability. Generalizable NeRFs (Gen-NeRF), while intended to address this, often produce blurring artifacts in unobserved regions with sparse inputs, which are full of uncertainty. In this paper, we aim to diminish the uncertainty in Gen-NeRF for plausible renderings. ***We assume that NeRF's inability to effectively mitigate this uncertainty stems from its inherent lack of generative capacity.*** *Therefore, we innovatively propose an Indirect Diffusion-guided NeRF framework, termed ID-NeRF, to address this uncertainty from a generative perspective by leveraging a distilled diffusion prior as guidance. Specifically, to avoid model confusion caused by directly regularizing with inconsistent samplings as in previous methods, our approach introduces a strategy to indirectly inject the inherently missing imagination into the learned implicit function through a diffusion-guided latent space. Empirical evaluation across various benchmarks demonstrates the superior performance of our approach in handling uncertainty with sparse inputs.*

1. Introduction

Implicit neural representations [17, 18, 22], represented by NeRF, have recently exhibited remarkable potential across diverse 3D vision tasks, such as novel view synthesis (NVS), virtual try-on, avatar reconstruction, etc. Although they offer substantial advantages in spatial resolution and representational capacity compared to conventional explicit representations, they are also limited by necessity the for per-scene optimization with a substantial number of posed images, which is often impractical for real-world applications.

A line of research [2, 3, 8, 10, 15, 37, 43, 50, 52] attempts to address the issue by implementing *generalizable NeRF* from sparse inputs. These methods achieve generalization by adding scene-specific conditional features to NeRF. Typically, these methods take a geometric transformation to extract existing features as conditions. This approach is illustrated in Fig. 1, where the 3D points of target rays are reprojected into the sparse input (or reference) views to extract visual cues. However, this approach becomes problematic when the sparse input views differ significantly from the

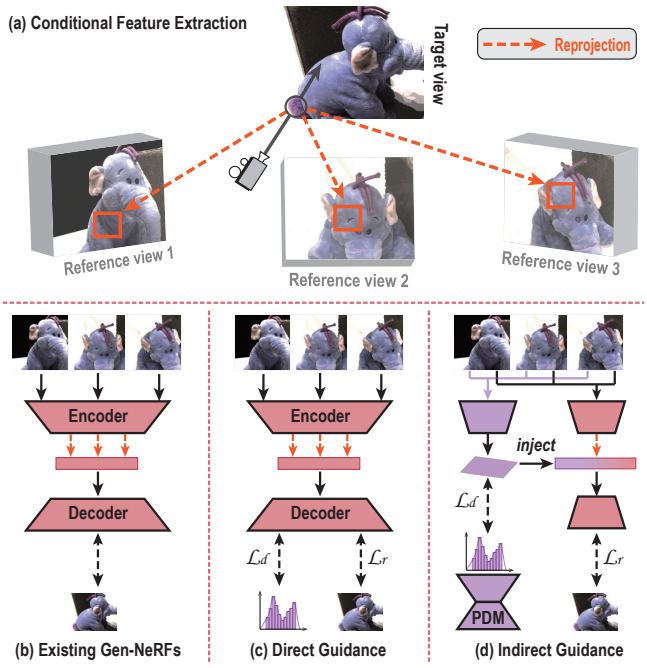

Figure 1. (a) A scenario illustrating the reprojection principle of Generalizable NeRF. In the target view, 3D points from a target ray are reprojected to the reference views to extract features, which serve as conditions for NeRF; (b) The inference process of existing Gen-NeRFs; (c) Gen-NeRF under direct guidance using the modeled distribution; (d) Our model that uses indirect guidance. In this model, the reprojected features are refined using a diffusion-guided latent space (purple patch). $\mathcal{L}_d$ and $\mathcal{L}_r$ are distillation and reconstruction losses, respectively.

target view [14, 53]. As shown in Fig. 1 (a), the purple tail is unobserved in the inputs. *As a result, the corresponding area is inherently uncertain, potentially appearing in yellow, red, or some other color.* These visual cues are absent in the input views, causing the conditional feature obtained through reprojection to be incorrect and resulting in undesirable artifacts.

To tackle this issue, we can draw inspiration from prior research efforts [12, 45, 49] that use distributions generated by generative models to directly guide (supervise) NeRF renderings, as depicted in Fig. 1 (c). This method can reduce the uncertainty, as the distribution modeled by generative models contains potential visual cues for unobserved regions. However, it may still encounter difficulties in producing photorealistic renderings. This limitation arises from the inconsistency of views sampled from the distribution. Direct supervision using the sampled views can perplex the model, causing it to favor smoothing [20, 21, 53] predictions.

In this work, we introduce generative models for Gen-NeRFs to address the challenge of uncertainty. Instead of relying on direct guidance, we present an innovative framework that adopts *indirect guidance*. The main idea is to use a diffusion-guided latent space to refine the problematic reprojected features yielded by the traditional Gen-NeRFs. As illustrated in Fig. 1 (d), we infer this latent space by employing an additional scene encoder and then applying score-based distillation [24, 42] to it, using the distribution generated by a conditioned pre-trained diffusion model (PDM) [29]. In doing so, our method avoids blurry rendering caused by inconsistent supervised signals and is capable of producing photorealistic predictions.

We have conducted extensive experiments to validate our design. The results demonstrate that our method achieves superior results compared with state-of-the-art (SOTA) methods. In summary, our primary contributions are as follows:

- We make an early attempt to address the uncertainty in existing Gen-NeRFs from an indirect generative perspective.
- We propose a novel ID-NeRF framework that provides indirect guidance to unobserved regions by distilling an imaginative latent space, avoiding model confusion while achieving high-quality renderings.
- Extensive experiments are conducted to show that our method achieves remarkable performance.

2. Related Work

2.1. Generalizable NeRF

NeRF [18] in its basic form is an MLP-based mapping function for predicting rendering attributes like color and density with scene-agnostic inputs of coordinates and orientations. Therefore, it needs to be retrained when rendering a new scene, since the mapping function cannot produce different content for the same inputs. To address this issue, subsequent works [2, 3, 8, 10, 15, 37, 43, 50, 52] have attempted to make NeRF generalizable by adding scene-dependent discriminative conditions as additional inputs.

Pixel-NeRF [50] is among the pioneering efforts in this direction. It constructs a feature volume and extracts image features from it to serve as conditions for NeRF. IBRNet [43] proposes a weighted MLP network to process the local features extracted from nearby views, followed by a ray transformer and an MLP to predict density and color. MVSNeRF [2] utilizes a 3D CNN network to process a constructed 3D cost volume, and then uses a conditional MLP to predict color and density. A recent approach, DBARF [3], trains a cost feature map in a self-supervised manner and generalizes by projecting and interpolating local features, similar to IBRNet.

These methods extract local features as discriminative conditions for each scene through reprojection. This implies that they rely on visual cues present in the reference views. As a result, when the input views are sparse or differ significantly from the target view, the reprojected visual cues for unseen regions will likely be erroneous. Some works try to address the uncertainty by incorporating additional geometric constraints. For instance, ContraNeRF [48] proposes to use geometry-aware contrastive learning to refine the extracted features, while MatchNeRF [4] performs cosine similarity computation on sampled local features. Nevertheless, this filter-based approach does not fundamentally improve the visual cues, since it cannot generate new and reasonable content for unobserved regions beyond what is already present in the reference views.

2.2. NeRF with Generative Models

Despite the challenge posed by sparse inputs, recent research [11, 19, 32, 39, 41] has increasingly focused on this scenario. Some of these works attempt to alleviate the issue of information sparsity by introducing additional supervision to regularize geometry. However, these approaches typically require cumbersome and time-consuming extra inputs like depth maps [5, 28, 41].

In contrast, an alternative line of research [12, 45, 49] seeks to produce such additional supervised information in a generative manner. For instance, DiffusioNeRF [45] generates a scene prior using a denoising diffusion model (DDM) to supervise NeRF's color and density predictions, while FreatureNeRF [49] supervises a feature map rendered by NeRF using pre-trained vision foundation models [1, 29]. However, these methods sample views from the generated distribution as the ground truth, which may correspond to inconsistent 3D objects, such as two elephants with purple

and red tails. These inconsistent supervision signals can confuse the model and result in ambiguous predictions [20].

Coincidentally, a similar challenge also arises in the domain of text-to-3D. In this context, several works [13, 23, 24, 26, 42, 47] have attempted to directly supervise NeRF renderings with distributions generated by text-conditioned PDMs [29, 30], resulting in suboptimal outcomes. To address this issue, Sparsefusion [53] and Nerfdiff [6] propose minimizing the negative log-likelihood of NeRF's parameters. These approaches essentially select the most probable virtual view from the modeled distribution to supervise the NeRF renderings. While this strategy can circumvent model confusion, it may entail a loss of information.

Due to the explosive growth of relevant work, our discussion in this section is necessarily limited to a subset of preceding studies. However, our focal point remains on the imaginative generative capability currently absent in Gen-NeRFs, which is crucial for addressing the uncertainty therein. Thus, drawing inspiration from the efficacy of generative models in the prior research, we incorporate the PDM into Gen-NeRFs to address the challenge of uncertainty. Nonetheless, to prevent model confusion, we present a novel indirect guidance framework for its implementation.

3. METHODOLOGY

We propose ID-NeRF, a Gen-NeRF framework under the indirect guidance of the PDM. In this section, we first present the preliminaries and then unfold our approach step-by-step. An overview of our approach is illustrated in Fig. 2.

3.1. Preliminaries

Generalizable NeRF. Vanilla NeRF [18] maps the coordinate x and ray direction d of a 3D point to its density σ and color c using a two-stage MLP network $\mathcal{M}_\theta : (x, d) \rightarrow (\sigma, c)$. The below volume rendering is then used to compute the color of each emitted ray by integrating over the N sampling points within its near and far bounds $[t_n, t_f]$:

$$C(\mathrm{r}) = \sum_{i=1}^{N} T_i (1 - \exp(-\sigma_i \delta_i)) c_i \tag{1}$$

where $T_i = \exp(-\sum_{j=1}^{i-1} \sigma_j \delta_j)$ is the volume transmittance and $\delta_i = t_{i+1} - t_i$ is the interval distance. The reconstruction loss is performed on $C(\mathrm{r})$ for optimization.

For different scenes, the inputs x, d are the same for the above manner and hence the vanilla NeRF cannot be generalized. Therefore the current Gen-NeRFs inputs scene-specific geometric information into $\mathcal{M}_\theta$ as discriminative condition. Given a scene with N input views $\mathcal{I} = \{I_i\}_{i=1}^N$ and corresponding camera intrinsic and extrinsic parameters $\mathcal{P} = \{P_i\}_{i=1}^N$, for predicting the color and density of a 3D point of a novel view I, Gen-NeRFs will first project it to N input views to extract scene geometric features:

$$z = \mathcal{G}(\{p(x, P_i), b(I_i)\}_{i=1}^N) \tag{2}$$

where $\mathcal{G}(\cdot)$ is the aggregation function, different for different methods. $b(\cdot)$ is the backbone. $p(\cdot)$ is the projection operation, which utilizes the camera parameter P_i to project the 3D point x into 2D coordinates on each reference view plane and then extracts the corresponding feature values, as described in more detail in [48, 50]. The geometric condition z is then entered into $\mathcal{M}_\theta$ along with the coordinate x and direction d to predict the rendering factor.

$$\sigma, c = \mathcal{M}_\theta(x, d, z) \tag{3}$$

Score-based Distillation. Denoising diffusion probabilistic models (DDPM) [7, 34], or score-based generative models [35, 36] have recently shown remarkable success in content generation [27, 29, 30]. In the field of text-to-3D, in order to utilize the knowledge of pre-trained DDPMs to bootstrap 3D generation, DreamFusion [24] pioneered the Score Distillation Sampling (SDS) loss, on which most of the subsequent works [13, 26, 31, 47] have been implemented. For a NeRF-rendered image I, the SDS optimization first performs a diffusion process on it, i.e., noising it to a standard Gaussian distribution in t time steps.

$$I_t = \sqrt{\bar{\alpha}_t} I + \sqrt{1 - \bar{\alpha}_t} \epsilon \tag{4}$$

In the reverse (generative) process, the pre-trained DDPM generates t noises under the text condition γ to continuously denoise the standard Gaussian distribution. Then, the score-based distillation gradient of the rendered image can be obtained as follows, where ϵ_ϕ is the pre-trained denoiser.

$$\mathcal{L}_{SDS} := \mathbb{E}_{I,t,\gamma,\epsilon\sim\mathcal{N}(0,1)}[\|\epsilon_\phi(I_t, t, \gamma) - \epsilon\|_2^2] \tag{5}$$

3.2. Scene-specific Geometric Information Extraction

Given N input views $\mathcal{I} = \{I_i\}_{i=1}^N$, ID-NeRF first extracts scene-specific geometric information like traditional Gen-NeRFs. Among these methods, we choose MatchNeRF [4] as the main reference. Firstly, we use an identical simple CNN network as MatchNeRF to extract two downsampled features $(1/4, 1/8)$, and then we process them with GMFlow 's Transformer [46] to obtain two enhanced features $\{F_i, \hat{F}_i\}_{i=1}^N$. For each 3D point of the target ray, we reprojected it into these features to obtain the local features $\{f_i, \hat{f}_i\}_{i=1}^N$, and then we extracted the local color c_i as additional geometric information like most methods [2, 4, 43]. Finally, the above information is concatenated together to obtain the reprojected features $\mathcal{F} : \{F_i\}_{i=1}^N$.

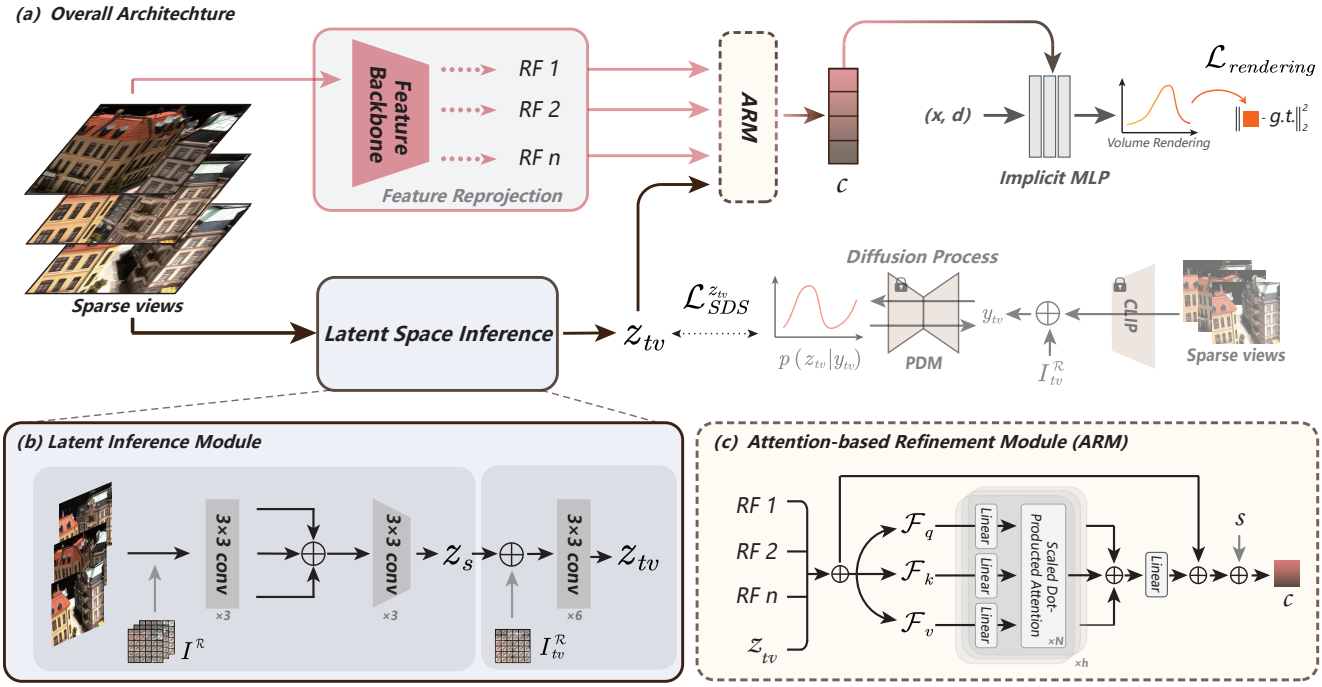

Figure 2. Overview of our ID-NeRF. Given sparse views, there are two information flows to process them. The first (red) utilizes geometric reprojection to obtain reprojected features (RF). The other one (black) uses an inference module to predict the latent space z_{tv}, which is performed score-based distillation with the PDM-predicted distribution $p(z_{tv}|y)$. Then, these features are fed together into the ARM to obtain the refined conditional feature c. $I^{\mathcal{R}}$ and $I^{\mathcal{R}}_{tv}$ are ray images used to enhance the pose information, see Sec. 3.3 for their details.

3.3. Latent Space Inference Assisted by Distilling PDM

Another stream of data processing represents the core of our approach, which infers the latent space with the aid of distilling PDM. On the one hand, we use a latent inference module $\mathcal{E}$ to compress $\mathcal{I}$ into a latent space z that contains all possible visual cues. On the other hand, we sample a $\hat{z}$ from the conditional distribution $p(z|y)$ predicted by a latent PDM [29] to supervise it.

Specifically, the inference module contains two stages. In the first stage, for better inference, we add pose information to the reference views before feeding them to the CNN network. To achieve this, we concatenate a ray image $I^r_i = \mathcal{R}_d - \mathcal{R}_o$ for each I_i to obtain the $I^{\mathcal{R}}_i$, where $\mathcal{R}_d, \mathcal{R}_o \in \mathbb{R}^{3\times H\times W}$ are the direction and origin of the ray at each pixel of I_i, respectively. These enhanced images are passed through a 3-layer weight-sharing CNN network to obtain geometric features $\{f_i\}_{i=1}^N$, which is then concatenated and fed into another CNN to predict the scene's latent space z_s.

The first stage is similar to an auto-encoder and aims at inferring the latent containing the entire scene information. However, the inferred latent space z_s may be some noise for a specific view. Therefore in order to provide fine-grained guidance at the target view level, we further introduce the target pose into z_s in the next stage to infer the target view's latent space z_{tv}. Specifically, we take the same manner as described above to create a target viewpoint ray image $I^{\mathcal{R}}_{tv}$, which undergoes a 6-layer CNN to yield z_{tv} after being concatenated with z_s. After this, $I^{\mathcal{R}}_{tv}$ is added to $\mathcal{E}_c(\mathcal{I})$ to obtain condition y_{tv}, where $\mathcal{E}_c$ is the CLIP's encoder [25].

Finally, y_{tv} is fed into the PDM to predict $p(z_{tv}|y_{tv})$, and then z_{tv} is optimized by the following SDS loss, where z_{tv_t} is the noise latent code of z_{tv} at the t-th time step. Note that the parameters of the CLIP and PDM modules used here are frozen, and therefore, they do not participate in the gradient updating process during training. Additionally, they are not involved in the inference process.

$$\mathcal{L}^{z_{tv}}_{SDS} = \mathbb{E}_{\mathcal{E}_2(z_s, I^{\mathcal{R}}_{tv}), t, y_{tv}, \epsilon\sim\mathcal{N}(0,1)}[\|\epsilon_\phi(z_{tv_t}, t, y_{tv}) - \epsilon\|^2_2] \tag{6}$$

3.4. Attention-based Refinement for Reprojected Visual Cues

When inputs are sparse, errors will occur in the reprojected features $\mathcal{F}$. In this subsection, we use the above distilled latent space to instruct them based on the attention mechanism [40]. Specifically, we first concatenate z_{tv} and each F_i into a vector sequence. Then, the se-

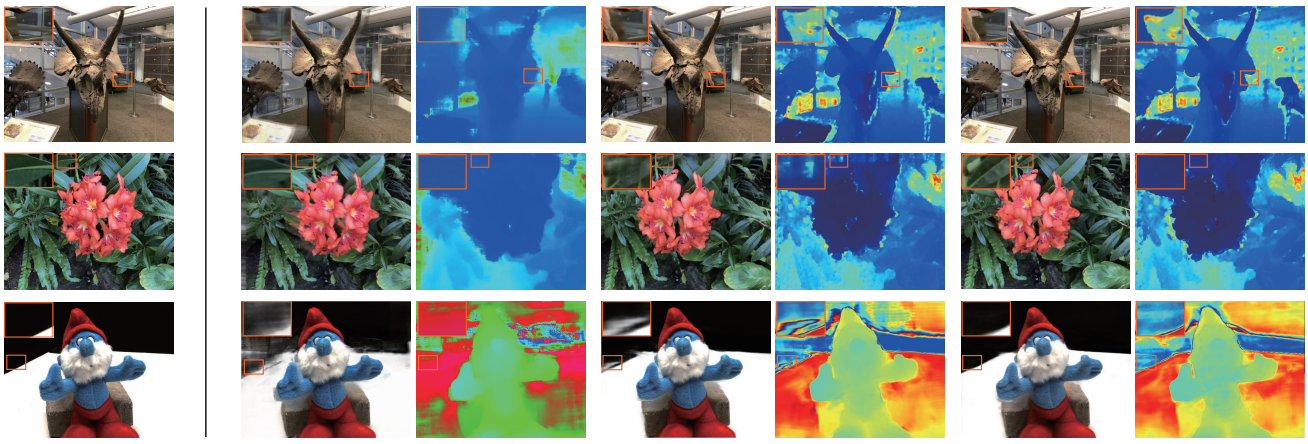

Figure 3. Qualitative comparison of rendering results. We present the rendered RGB images and depth maps of our ID-NeRF as well as representative MVSNeRF [2] and MatchNeRF [4], with each result zoomed in on details.

quence $x \in \mathbb{R}^{(N+1)\times d}$ is input into a N-layer multi-head self-attention (MSA) module [40] to adaptively generate weights for fusion. The MSA can be formulated as follows:

$$SA(x) = softmax(\frac{xW_q(xW_k)^\top}{\sqrt{d_k}})xW_q \tag{7}$$

$$MSA(x) = cat(SA_1(x), \ldots, SA_h(x))xW_m \tag{8}$$

where $W_q, W_k, W_v \in \mathbb{R}^{d\times d_p}$ and $W_m \in \mathbb{R}^{hd_p\times d}$ are the learnable weight matrices and h is the number of heads. As shown in Fig. 2, after fusion we perform a skip connection operation. And, for a fair comparison with MatchNeRF, we add similarity information s to the refined feature, which is computed as detailed in [4].

3.5. Optimization

Volume Rendering. The purpose of ID-NeRF is to improve the reprojected features in the traditional Gen-NeRFs. In order to demonstrate the effectiveness of our method, we take the same MLP-based rendering network in MatchNeRF to introduce the conditional feature c. The predicted colors and densities are rendered by Eq. 1 to get the predicted colors $C(r)$ for each ray.

Training objective. After rendering, the following loss is implemented on $C(r)$ by the ground truth $\tilde{C}(r)$:

$$\mathcal{L}_{rendering} = \sum_{r\in\mathcal{R}} \left\| C(r) - \tilde{C}(r) \right\|_2^2 \tag{9}$$

Thus, the total loss is as follows.

$$\mathcal{L}_{total} = \mathcal{L}_{SDS}^{z_{tv}} + \mathcal{L}_{rendering} \tag{10}$$

4. Experimental Results

4.1. Experimental Settings

Datasets and Evaluation. We adopt the same experimental protocol [2, 4, 50] for comparison with SOTA methods, i.e., training on 88 scenes and then testing on another 16 scenes on the DTU [9] dataset. For all compared methods, we choose 3 input views for training, followed by testing in 4 views of each test scene. In addition, we perform additional tests in a more challenging 2-input setting on 8 scenes from Real Forward-Facing (RFF) [18] and Blender [18], respectively. For evaluation, we choose PSNR, SSIM [44] and LPIPS [51] as metrics.

Implementation Details. We implement our ID-NeRF on a single NVIDIA A100 GPU with 600k steps, sampling 1024 rays for training and 4096 rays for testing. In the MSA module, d, dp, and h are 8, 4, and 4, respectively. AdamW [16] algorithm and one cycle policy [33] are adopted for optimization. We keep the learning rates of GMFlow and rendering network consistent with the MatchNeRF and set the initial learning rate of the remaining modules to 1e-3. We implement SDS loss based on the codebase [38] utilizing stable-diffusion (SD) with version v2-1.

4.2. Main Results

For fair comparisons, we mainly compare with SOTA Gen-NeRFs that use only images, including PixelNeRF [50], IBRNet [43], MVSNeRF [2], and MatchNeRF [4]. Additionally, we take the measurement in MVSNeRF [2]. We first perform quantitative comparisons with these methods in the 3-input view setting, which are reported in Table 1. Furthermore, to showcase the performance of our method, we also perform the comparison in the more challenging 2-input setting, which is reported in Table 2. The

Table 1. Quantitative results of 3-input setup on the DTU dataset.

Method	PSNR↑	SSIM↑	LPIPS↓
PixelNeRF [50]	19.31	0.789	0.382
IBRNet [43]	26.04	0.917	0.190
MVSNeRF [2]	26.63	0.931	0.168
MatchNeRF [4]	26.91	**0.934**	0.159
ID-NeRF	**27.06**	0.921	**0.158**

Table 2. Quantitative results of 2-input setup on different datasets

Datasets	Method	PSNR↑	SSIM↑	LPIPS↓
DTU	MVSNeRF [2]	24.03	0.914	0.192
	MatchNeRF [4]	25.03	0.919	0.181
	ID-NeRF	**25.20**	**0.920**	**0.178**
Blender	MVSNeRF [2]	20.56	0.856	0.243
	MatchNeRF [4]	20.57	0.864	0.200
	ID-NeRF	**20.60**	**0.868**	**0.197**
RFF	MVSNeRF [2]	20.22	0.763	0.287
	MatchNeRF [4]	20.59	0.775	0.276
	ID-NeRF	**20.83**	**0.811**	**0.206**

visualization results are given in Fig. 3.

For the experiment trained with 3-input views, we report the results of testing on the DTU dataset in Table 1. As shown in the table, our method achieves the best results on PSNR and LPIPS metrics. Due to the reliance on a large number of input images, PixelNeRF does not work well under the challenging 3-input setting. The follow-up method (MVSNeRF), despite some improvement on this, predicts the volume density poorly due to inferior acquisition of visual cues, further leading to mediocre rendered depth maps, as shown in Fig. 3. MatchNeRF enhances the visual cues by geometric matching but still produces some artifacts on the contours of the target or background. On this basis, we enhance the visual cues by injecting the powerful knowledge of PDM to be more sensitive to the boundary information, making it possible to generate more accurate depth maps while ensuring clear contours, as illustrated in Fig. 3.

To further validate our model, we train it in a more challenging 2-input setting and test it on both DTU, Blender, and RFF datasets. MVSNeRF and MatchNeRF, which perform second and third above, are chosen for comparison. As shown in Table 2, under this more challenging setting, our model instead performs better, achieving the optimum

Table 3. Quantitative results for different sparsity settings.

Input views	Method	PSNR↑	SSIM↑	LPIPS↓
Setting 1	MatchNeRF [4]	26.23	0.896	0.143
	Ours	**26.48**	**0.897**	**0.141**
Setting 3	MatchNeRF [4]	17.18	0.725	0.278
	Ours	**17.98**	**0.734**	**0.270**
Setting 2	MatchNeRF [4]	14.42	0.674	0.322
	Ours	**15.79**	**0.688**	**0.318**

Table 4. Ablation experiments of latent space on the DTU dataset.

Latent Space	PSNR↑	SSIM↑	LPIPS↓
$No\text{-}latent$	26.62	0.912	0.166
z_{tv}	**27.06**	**0.921**	**0.158**

on all metrics for all three datasets. Analytically, the case of the 2-input view contains fewer visual cues for the target view, making the unobserved region larger and the uncertainty consequently greater. Our model performs better in such a setup, demonstrating that our model can better mitigate the uncertainty problem relative to MVSNeRF and MatchNeRF.

Moreover, it is important to note that the measurement standard for the above results takes the nearby views as inputs. This means that the input views are few but contain many regions of the target view. In other words, there are few unobserved regions in the above settings so the uncertainty problem has actually little impact. Therefore, to adequately demonstrate the advantages of our model in dealing with unobserved regions, we choose a fixed view (view 44 of all DTU's validation scenes) as the target view and then select three sets of input views to render it. These three sets of views are (43, 33, 31), (26, 22, 10), and (22, 10, 3), which are progressively sparser and contain more and more unobserved regions relative to the target view. The results are reported in Table 3, from which we can see that as the inputs get sparser (settings 1 to 3), our method results in a larger performance advantage over MatchNeRF. This suggests that our method is better at handling uncertain unobserved regions compared to MatchNeRF.

4.3. Ablation Studies

Effect of latent space. We evaluated the role of the latent space in our model and the results are reported in Table 4. The $No\text{-}latent$ method utilizes more reprojected fea-

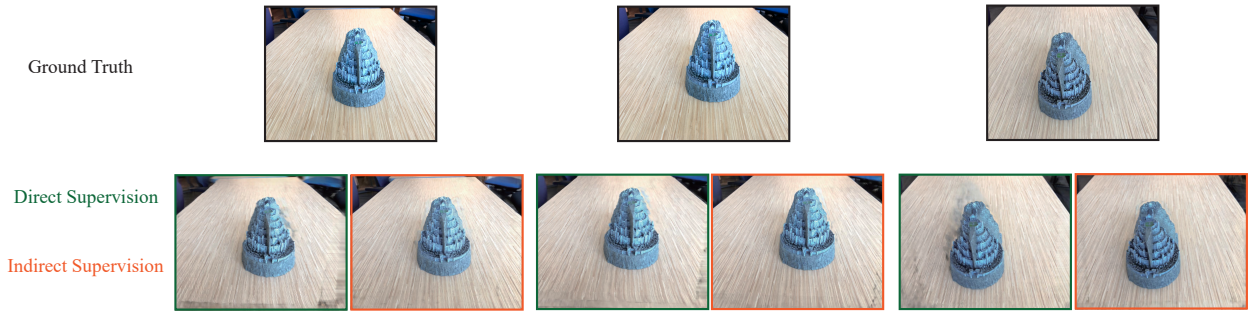

Figure 4. Qualitative comparison of two supervision approaches. Both methods are trained on the DTU dataset with 3 input views.

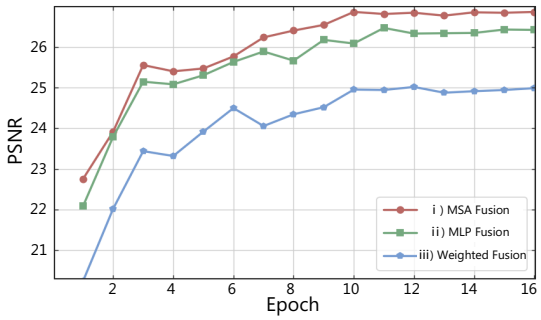

Figure 5. Comparison of different guidance approaches on the DTU dataset.

tures than MatchNeRF (reported in Table 1) yet achieves worse results. This anomaly proves that there exists some noise in the reprojected features, which interferes with the model's predictions. Further, with the introduction of the latent space z_{tv} as a guide, the results are substantially improved, as shown in Table 4. This confirms the effectiveness of our latent space in improving these reprojected features.

Effect of attention guidance. To enable the guidance of the latent space to the reprojected features, we propose the AGM module that utilizes the MSA mechanism for this purpose. The ablation experiments are executed here to explore its effectiveness. Specifically, three different guidance strategies are devised for the latent and reprojected features: i) using a layer of MSA to adaptively generate weights to fuse them; ii) inputting them directly into a layer of MLP to perform the fusion; and iii) inputting them into a layer of MLP to generate respective weights, and then performing weighted fusion. The results are reported in Fig. 5, where the MSA approach makes the best prediction.

Effect of indirect supervision. Our method indirectly supervises the NeRF rendering results using the distilled latent space, in order to demonstrate the superiority of this indirect supervision manner, we compare it here with the direct supervision approach. Specifically, for direct supervision, the latent space z_{tv} is no longer used, and the distribution generated by SD conditioned on the reference images will be computed loss directly with NeRF renderings. After the same implementation details described above, its visualization results are shown in Fig. 4 for comparison with our indirect supervision manner. As can be seen in the results, our indirect supervision approach can make sharp renderings, while the direct supervision approach may struggle in regions such as background and object contours.

4.4. Limitations

Although our approach takes a promising step towards addressing the uncertainty in Gen-NeRFs, there are still some limitations that need to be addressed. For instance, there is room for improvement in image fidelity within our model, as evidenced by the experiments where SSIM often performs worse than PSNR. This may be because the visual cues generated by stable-diffusion are sometimes less compatible with the human visual system, which leads to the limitation of ID-NeRF on both SSIM and LPIPS.

5. CONCLUSIONS

In this study, we aim to address the uncertainty issue in Gen-NeRFs with sparse inputs from a generative perspective. We introduce an innovative Gen-NeRF framework, ID-NeRF, which injects generative capabilities into NeRF indirectly using a pre-trained diffusion model. This indirect guidance strategy facilitates the transfer of knowledge from the PDM into a latent space through score-based distillation, subsequently refining the reprojected features of traditional Gen-NeRFs. Consequently, ID-NeRF not only alleviates the inherent uncertainty problem associated with sparse inputs but also effectively avoids bothersome model confusion. Our empirical results demonstrate the outstanding performance of our method across various experimental settings. In the future, we will continue to explore solutions to the challenge of uncertainty in Gen-NeRFs.

References

[1] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 2

[2] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14124–14133, 2021. 1, 2, 3, 5, 6

[3] Yu Chen and Gim Hee Lee. Dbarf: Deep bundle-adjusting generalizable neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24–34, 2023. 1, 2

[4] Yuedong Chen, Haofei Xu, Qianyi Wu, Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. Explicit correspondence matching for generalizable neural radiance fields. *arXiv preprint arXiv:2304.12294*, 2023. 2, 3, 5, 6

[5] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12882–12891, 2022. 2

[6] Jiatao Gu, Alex Trevithick, Kai-En Lin, Joshua M Susskind, Christian Theobalt, Lingjie Liu, and Ravi Ramamoorthi. Nerfdiff: Single-image view synthesis with nerf-guided distillation from 3d-aware diffusion. In *International Conference on Machine Learning*, pages 11808–11826. PMLR, 2023. 3

[7] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 3

[8] Xin Huang, Qi Zhang, Ying Feng, Xiaoyu Li, Xuan Wang, and Qing Wang. Local implicit ray function for generalizable radiance field representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 97–107, 2023. 1, 2

[9] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 406–413, 2014. 5

[10] Mohammad Mahdi Johari, Yann Lepoittevin, and François Fleuret. Geonerf: Generalizing nerf with geometry priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18365–18375, 2022. 1, 2

[11] Mijeong Kim, Seonguk Seo, and Bohyung Han. Infonerf: Ray entropy minimization for few-shot neural volume rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12912–12921, 2022. 2

[12] Adam R Kosiorek, Heiko Strathmann, Daniel Zoran, Pol Moreno, Rosalia Schneider, Sona Mokrá, and Danilo Jimenez Rezende. Nerf-vae: A geometry aware 3d scene generative model. In *International Conference on Machine Learning*, pages 5742–5752. PMLR, 2021. 2

[13] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309, 2023. 3

[14] Kai-En Lin, Yen-Chen Lin, Wei-Sheng Lai, Tsung-Yi Lin, Yi-Chang Shih, and Ravi Ramamoorthi. Vision transformer for nerf-based view synthesis from a single input image. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 806–815, January 2023. 2

[15] Yuan Liu, Sida Peng, Lingjie Liu, Qianqian Wang, Peng Wang, Christian Theobalt, Xiaowei Zhou, and Wenping Wang. Neural rays for occlusion-aware image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7824–7833, 2022. 1, 2

[16] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

[17] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019. 1

[18] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2, 3, 5

[19] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5480–5490, 2022. 2

[20] Curtis Northcutt, Lu Jiang, and Isaac Chuang. Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research*, 70:1373–1411, 2021. 2, 3

[21] Curtis G Northcutt, Tailin Wu, and Isaac L Chuang. Learning with confident examples: Rank pruning for robust classification with noisy labels. *arXiv preprint arXiv:1705.01936*, 2017. 2

[22] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019. 1

[23] Ryan Po and Gordon Wetzstein. Compositional 3d scene generation using locally conditioned diffusion. *arXiv preprint arXiv:2303.12218*, 2023. 3

[24] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 2, 3

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry,

Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 4

[26] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan Barron, et al. Dreambooth3d: Subject-driven text-to-3d generation. *arXiv preprint arXiv:2303.13508*, 2023. 3

[27] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022. 3

[28] Barbara Roessle, Jonathan T Barron, Ben Mildenhall, Pratul P Srinivasan, and Matthias Nießner. Dense depth priors for neural radiance fields from sparse input views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12892–12901, 2022. 2

[29] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2, 3, 4

[30] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022. 3

[31] Junyoung Seo, Wooseok Jang, Min-Seop Kwak, Jaehoon Ko, Hyeonsu Kim, Junho Kim, Jin-Hwa Kim, Jiyoung Lee, and Seungryong Kim. Let 2d diffusion model know 3d-consistency for robust text-to-3d generation. *arXiv preprint arXiv:2303.07937*, 2023. 3

[32] Seunghyeon Seo, Donghoon Han, Yeonjin Chang, and Nojun Kwak. Mixnerf: Modeling a ray with mixture density for novel view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20659–20668, 2023. 2

[33] Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, volume 11006, pages 369–386. SPIE, 2019. 5

[34] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 3

[35] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 3

[36] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. 2021. 3

[37] Mohammed Suhail, Carlos Esteves, Leonid Sigal, and Ameesh Makadia. Generalizable patch-based neural rendering. In *European Conference on Computer Vision*. Springer, 2022. 1, 2

[38] Jiaxiang Tang. Stable-dreamfusion: Text-to-3d with stable-diffusion, 2022. https://github.com/ashawkey/stable-dreamfusion. 5

[39] Prune Truong, Marie-Julie Rakotosaona, Fabian Manhardt, and Federico Tombari. Sparf: Neural radiance fields from sparse and noisy poses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4190–4200, 2023. 2

[40] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4, 5

[41] Guangcong Wang, Zhaoxi Chen, Chen Change Loy, and Ziwei Liu. Sparsenerf: Distilling depth ranking for few-shot novel view synthesis. *arXiv preprint arXiv:2303.16196*, 2023. 2

[42] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12619–12629, 2023. 2, 3

[43] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2021. 1, 2, 3, 5, 6

[44] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 5

[45] Jamie Wynn and Daniyar Turmukhambetov. Diffusionerf: Regularizing neural radiance fields with denoising diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4180–4189, 2023. 2

[46] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, and Dacheng Tao. Gmflow: Learning optical flow via global matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8121–8130, 2022. 3

[47] Jiale Xu, Xintao Wang, Weihao Cheng, Yan-Pei Cao, Ying Shan, Xiaohu Qie, and Shenghua Gao. Dream3d: Zero-shot text-to-3d synthesis using 3d shape prior and text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20908–20918, 2023. 3

[48] Hao Yang, Lanqing Hong, Aoxue Li, Tianyang Hu, Zhenguo Li, Gim Hee Lee, and Liwei Wang. Contranerf: Generalizable neural radiance fields for synthetic-to-real novel view synthesis via contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16508–16517, 2023. 2, 3

[49] Jianglong Ye, Naiyan Wang, and Xiaolong Wang. Featurenerf: Learning generalizable nerfs by distilling pre-trained vision foundation models. *arXiv preprint arXiv:2303.12786*, 2023. 2

[50] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4578–4587, 2021. 1, 2, 3, 5, 6

[51] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 5

[52] Xiaoshuai Zhang, Sai Bi, Kalyan Sunkavalli, Hao Su, and Zexiang Xu. Nerfusion: Fusing radiance fields for large-scale scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5449–5458, 2022. 1, 2

[53] Zhizhuo Zhou and Shubham Tulsiani. Sparsefusion: Distilling view-conditioned diffusion for 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12588–12597, 2023. 2, 3