PVSeRF: Joint Pixel-, Voxel- and Surface-Aligned Radiance Field for Single-Image Novel View Synthesis

Xianggang Yu
The Chinese University of Hong Kong, Shenzhen
xianggangyu@link.cuhk.edu.cn

Jiapeng Tang
Technische Universität München
jiapeng.tang@tum.de

Yipeng Qin
Cardiff University
qiny16@cardiff.ac.uk

Chenghong Li
The Chinese University of Hong Kong, Shenzhen
chenghongli@link.cuhk.edu.cn

Xiaoguang Han*
The Chinese University of Hong Kong, Shenzhen
hanxiaoguang@cuhk.edu.cn

Linchao Bao
Tencent AI Lab
linchaobao@gmail.com

Shuguang Cui
The Chinese University of Hong Kong, Shenzhen
shuguangcui@cuhk.edu.cn

ABSTRACT

We present PVSeRF, a learning framework that reconstructs neural radiance fields from single-view RGB images, for novel view synthesis. Previous solutions, such as pixelNeRF [71], rely only on pixel-aligned features and suffer from feature ambiguity issues. As a result, they struggle with the disentanglement of geometry and appearance, leading to implausible geometries and blurry results. To address this challenge, we propose to incorporate explicit geometry reasoning and combine it with pixel-aligned features for radiance field prediction. Specifically, in addition to pixel-aligned features, we further constrain the radiance field learning to be conditioned on i) voxel-aligned features learned from a coarse volumetric grid and ii) fine surface-aligned features extracted from a regressed point cloud. We show that the introduction of such geometry-aware features helps to achieve a better disentanglement between appearance and geometry, i.e. recovering more accurate geometries and synthesizing higher quality images of novel views. Extensive experiments against state-of-the-art methods on ShapeNet benchmarks demonstrate the superiority of our approach for single-image novel view synthesis.

CCS CONCEPTS

• **Computing methodologies** → **Rendering**; **Reconstruction**.

KEYWORDS

Novel view synthesis, Single image, Neural radiance field

*Corresponding author is Xiaoguang Han.

MM '22, October 10–14, 2022, Lisboa, Portugal

ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3547893

ACM Reference Format:
Xianggang Yu, Jiapeng Tang, Yipeng Qin, Chenghong Li, Xiaoguang Han, Linchao Bao, and Shuguang Cui. 2022. PVSeRF: Joint Pixel-, Voxel- and Surface-Aligned Radiance Field for Single-Image Novel View Synthesis. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisboa, Portugal.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3503161.3547893

1 INTRODUCTION

Novel view synthesis is a long-standing problem in computer vision and graphics, which plays a crucial role in various practical applications, including gaming, movie production, and virtual/augment reality. Recently, it has made great strides thanks to the advances in differentiable neural rendering [38, 69], especially the neural radiance fields (NeRF) [32] that simplifies novel view synthesis to an optimization problem over a dense set of ground truth views. Although achieving impressive results, the vanilla NeRF suffers from several limitations: i) the dense views it strictly requires are not always available; ii) it is slow in inference due to the long optimization process; iii) each NeRF is dedicated to a specific scene and cannot be generalized to new ones.

To address these issues, follow-up works such as pixelNeRF [71], IBRNet [60], and GRF [57], proposed to predict neural radiance fields in a feed-forward manner. Taking pixelNeRF as an example, it tackles the shortcomings of NeRF by extending its network to be conditioned on scene priors learnt by a convolutional image encoder. These scene priors are represented by spatial feature maps that allow the mapping from a pair of query spatial point and viewing direction to their corresponding pixel-aligned features. In pixelNeRF, such a mapping is implemented by standard camera projection and bilinear interpolation. During inference, the scene priors are obtained via a forward pass through the image encoder and thus allow fast novel view synthesis from a single input view of diverse scenes. Although effective, pixelNeRF suffers from feature ambiguity issues that originates from the *many-to-one* mapping between queries and their corresponding pixel-aligned features. In other words, pixelNeRF naively assigns the same pixel-aligned features to different points in some novel view as long as these

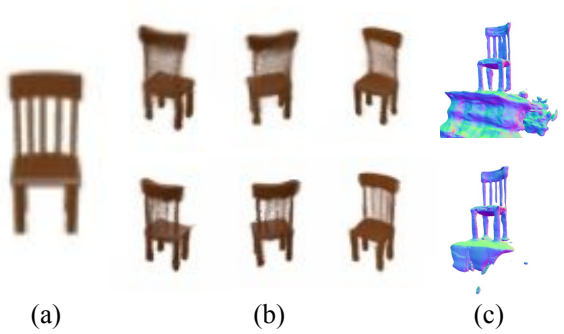

Figure 1: Novel view synthesis from a single image. (a) Input image. (b) Novel view synthesis results: pixelNeRF [71] (top) and Ours (bottom). (c) Surface meshes extracted from predicted radiance fields: pixelNeRF [71] (top) and Ours (bottom). By augmenting the 2D pixel-aligned features with complementary 3D geometric features for radiance field prediction, we can synthesize higher quality of novel views. A by-product of our approach is a cleaner implicit surface mesh, due to the introduction of explicit geometric features.

points overlap with each other in the input view, which can cause confusion (Fig. 2).

To clarify such ambiguity issues, we propose to incorporate explicit geometry reasoning and combine it with pixel-aligned features for radiance field prediction. Specifically, we leverage the recent success in single-view 3D reconstruction [5, 8, 11, 14, 31, 39, 50, 51] and inject rich geometry information into radiance field prediction by incorporating geometry-aware features of two shape representations: i) voxel-aligned features learned from a coarse volumetric grid and ii) fine surface-aligned features extracted from a regressed point cloud. Intuitively, such geometry-aware features augment pixel-aligned features with additional "dimensions", thereby allowing previously ambiguous points to be separable. Furthermore, by constraining the radiance field learning on these geometry-aware features, our method not just synthesize higher quality images of novel views, but also recover more accurate underlying geometries in radiance field, as witnessed in Fig. 1.

Our main contributions include:

- We propose a novel approach of learning neural radiance fields from single-view images jointly conditioned on pixel-, voxel-, surface-aligned features.
- We design an efficient way to alleviate the feature ambiguity issue of solely pixel-aligned features by incorporating explicit geometry reasoning via single-view 3D reconstruction.
- We propose a hybrid use of geometric features, including complementary coarse volumetric features and fine surface features.

2 RELATED WORK

2.1 Novel view synthesis and Neural radiance field

The task of novel view synthesis aims to generate new views of a scene from single or a set of sparse views. There are various kinds

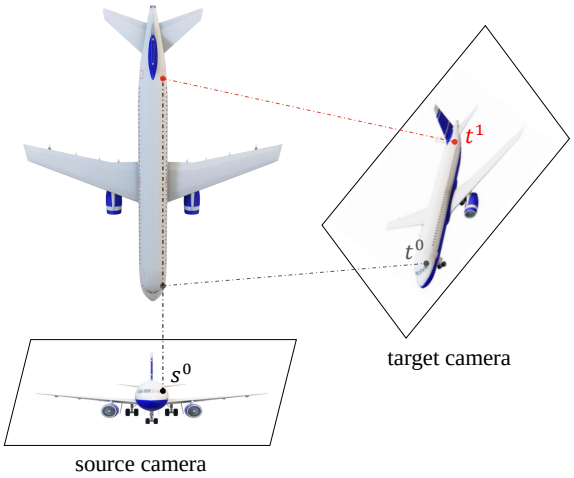

Figure 2: Illustration of the feature ambiguity issue. The feature ambiguity issues that originates from the *many-to-one* mapping between queries and their corresponding pixel-aligned features. Two rays shot from points t^0 and t^1 on target camera intersect the same ray shot from source camera. After the pixel-aligned process, the two different intersecting points will be projected to the same image coordinate s^0 on image-plane, obtaining the same pixelwise feature.

of approaches dedicated to this problem. Traditional methods [9, 13, 26] choose to estimate light fields and then render novel views. Recent years, with the advance of deep neural networks (DNN), a plethora of models are designed to learn novel view synthesis in an end-to-end manner. Pioneering methods [40, 54, 73] consider it as a image-to-image transformation problem and directly utilize 2D CNN to output novel views. These methods always cannot generate satisfactory results for viewpoints that are largely deviated from the given view.

Later work explore 3D-aware image synthesis and solve the inverse rendering problem via neural networks [12, 24, 36, 37, 48, 63, 74]. The common characteristic of this line of literature is that they recover the explicit or implicit 3D geometry and appearance properties first, then render novel views at desired camera viewpoints by means of differentiable rendering techniques [36] or generative models. Among these work, various 3D representations are employed. DeepVoxels [48] represents 3D scene properties by low-resolution volumetric feature grid lifted from 2D feature maps. Wiles *et al.* [63] use 3D surface features that are learned from the point cloud unprojected from the estimated depth map of the input view. Other approaches [12, 24, 37, 74] learn implicit 3D embedding that can be used to generate novel views of the same scene using unsupervised learning techniques.

Recently, witnessing the great success of neural radiance field (NeRF) [32], there has been an explosion of NeRF-based approaches for novel view synthesis[4, 7, 27–29, 34, 42, 46, 56, 57, 60, 70, 71]. There are two divisions in the prevalence of NeRF: 1) the first track tries to train scene-specific model for generating novel views of the scene [27–29, 34, 42, 46, 56, 70]. Specifically, they capture many diverse viewpoints of a scene, and optimizing a neural radiance field for that scene. Despite synthesizing high-fidelity novel views, these

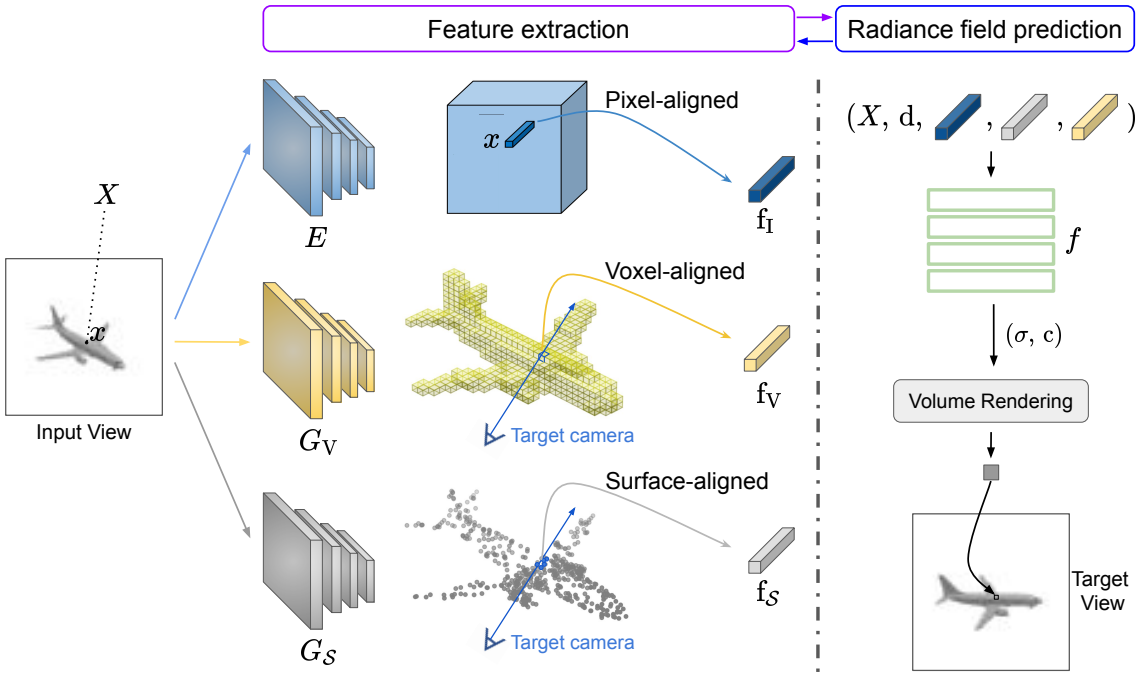

Figure 3: Overview of our PVSeRF framework. Given a single input image, we first 1) extract the spatial feature map using a fully convolutional image encoder E, 2) learn a volumetric grid through volume generator G_{V}, and 3) regress a surface point set of the object through a point set generator $G_{\mathcal{S}}$. From volumetric grid and surface point set, we can learn voxel features and point-wise features. Then, for a 3D location X and a target view direction d, we query pixel-, voxel-, and surface-aligned f_{I}, f_{V}, $\mathrm{f}_{\mathcal{S}}$ from spatial feature map, voxel features and point-wise features respectively. Next, the 3D location, view direction and all corresponding features are directed into a MLP to predict density σ and radiance r. Lastly, the volume rendering is used to accumulate the radiance prediction of points on the same ray to compute the final color values.

methods require longstanding optimization process and cannot generalize to new scenes. 2) the second track attempts to learn generalize neural radiance field across multiple scenes [4, 7, 57, 60, 71]. Among this, pixelNeRF [71] is the most relevant method to ours, which learns the scene priors conditioned on the pixel-aligned features, and can switch to new scenes flexibly. Although other methods [4, 7, 57, 60] can also be applied to novel scene through a single forward pass, they are equipped to multiple input views, while we focus on the more challenging single-view input setting.

2.2 Single-view 3D Object Reconstruction

Given a single image containing a object, 3D object reconstruction aims to recover the 3D geometry of the object. Traditional 3D reconstruction methods [1, 16, 19, 55] need to find dense correspondence across multi-view at the first, followed by the depth fusion stage. Recently, due to the establishment of large-scale 3D model datasets, such as ShapeNet [2] and ModelNet [66], it is popular to reconstruct complete 3D shape from a single image by utilizing shape priors modeled by deep neural networks. It also achieved various degrees of success by designing 3D shape decoders tailored for different shape representations including voxel [8, 15], point cloud [11, 35], mesh [14, 39, 51], and implicit field [5, 31, 41, 44, 51, 52]. The voxel decoders [8, 65] take advantages of conventional 3D convolution operations to generate volumetric grids. The point decoders [11, 68] directly regress the coordinates of 3D points. The mesh decoders mainly approximate a target shape by performing template mesh deformation [14, 21, 39, 50, 51]. The neural implicit functions [5, 31, 41, 44, 51] represent 3D surfaces by continuous functions defined in 3D space. In this paper, we want to incorporate explicit geometry reasoning into the process of single-view novel view synthesis by marrying single-view 3D shape generators with a generic radiance field learning model.

3 PVSERF

3.1 Preliminary: NeRF

Mildenhall *et al.* [32] proposed neural radiance field (NeRF) to represent a scene as a continuous 5D vector-valued function F of color and density. In particular, given a 3D location $\mathbf{X} \in \mathbb{R}^3$ and viewing direction vector $\mathbf{d} \in \mathbb{R}^2$, the continuous function F maps them into the emitted color $\mathbf{c} = (r, g, b)$ and volume density σ. NeRF use a multi-layer perceptron (MLP) network parameterized by weights Θ to approximate the 5D continuous function $F(\Theta)$: $(\mathbf{X}, \mathbf{d}) \rightarrow (\mathbf{c}, \sigma)$. To render the neural radiance field into a pixel, NeRF follows the classic volume rendering technique [20, 30]:

$$\mathbf{C}(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t), \mathbf{d})dt \tag{1}$$

where $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ is a camera ray casted from the camera center $\mathbf{o}$ along the direction $\mathbf{d}$ passing through the pixel on the image plane, $\mathbf{C}(\mathbf{r})$ is the expected color for that pixel, and $T(t)$ is the accumulated transmittance along the ray from t_n to t:

$$T(t) = \exp\left(-\int_{t_n}^{t} \sigma(\mathbf{r}(s))ds\right). \tag{2}$$

In practice, these integrals are approximated using the numerical quadrature rule [30]:

$$\begin{aligned} \hat{\mathbf{C}}(\mathbf{r}) &= \sum_{k=1}^{K} T_k \left(1 - \exp\left(-\sigma_k \left(t_{k+1} - t_k\right)\right)\right) \mathbf{c}_k, \\ \text{with} \quad T_k &= \exp\left(-\sum_{k'<k} \sigma_{k'} \left(t_{k'+1} - t_{k'}\right)\right). \end{aligned} \tag{3}$$

During training, the weights of a NeRF network are randomly initialized and optimized for an individual scene using a collection of RGB images, by minimizing a sole photometric loss $L_{photo} = \sum_{\mathbf{r}\in\mathcal{R}} \|\hat{\mathbf{C}}(\mathbf{r}) - \mathbf{C}(\mathbf{r})\|_2^2$, in which $\mathbf{r} \in \mathcal{R}$ is a set of randomly sampled rays from some images and $\mathbf{C}(\mathbf{r})$ is the ground truth color value of the pixel corresponding to ray $\mathbf{r}$.

3.2 Overview

Different from NeRF's approach which must be optimized per-scene individually, our PVSeRF framework leverage the prior knowledge across multiple scenes and can reconstruct a neural radiance field from as little as a single image which is similar to pixelNeRF [71]. Specifically, given a single calibrated image $\mathbf{I}$ with its corresponding intrinsic $\mathbf{K}$ and extrinsic parameters (rotation $\mathbf{R}$ and translation $\mathbf{t}$), our PVSeRF aims to learn a neural network for radiance field reconstruction:

$$\sigma, \mathbf{c} = \text{PVSeRF}(\mathbf{X}, \mathbf{d}; \mathbf{I}, [\mathbf{Rt}], \mathbf{K}) \tag{4}$$

where $\mathbf{X} \in \mathbb{R}^3$ represents the 3D location, $\mathbf{d} \in \mathbb{R}^2$ is the viewing direction, σ is the volume density at $\mathbf{X}$, and $\mathbf{c}$ is the predicted color at $\mathbf{X}$ depending on the viewing direction $\mathbf{d}$. By accumulating the σ and $\mathbf{c}$ of multiple points sampled on the ray defined by $\mathbf{X}$ and $\mathbf{d}$, we can obtain the color values of all pixels in a target view image $\mathbf{I_t}$ via differentiable rendering, thereby enabling novel view synthesis.

The distinct advantage of our PVSeRF is that it addresses the feature ambiguity issue of pixelNeRF [71] by a novel geometric regularization using both voxel- and surface-aligned features. As aforementioned, pixelNeRF's feature ambiguity issue stems from the fact that its network is solely conditioned on the 2D pixel-aligned features where multiple query 3D points are mapped to a single location. To clarify this ambiguity, we propose to augment the 2D pixel-aligned features with complementary 3D geometric features for radiance field construction. As Fig. 3 shows, in addition to the pixel-aligned features, our method incorporates i) voxel-aligned and ii) surface-aligned features into radiance field prediction. Specifically,

- We follow [71] and extract the pixel-aligned features $\mathbf{f_I}$ of a query point $\mathbf{X}$ by projecting it with $[\mathbf{Rt}]$ and $\mathbf{K}$ to the 2D image coordinates x, indexing the multi-scale feature maps of an input image $\mathbf{I}$ extracted by a fully-convolutional image encoder E.
- We extract the voxel-aligned features $\mathbf{f_V}$ of a query point $\mathbf{X}$ by trilinearly interpolating $\mathbf{X}$ in a low-resolution volumetric feature $\mathbf{F_V}$ learnt from the input image $\mathbf{I}$ using a volume generator $G_{\mathbf{V}}$. Note that $\mathbf{f_V}$ only captures coarse geometry contexts of the scene due to the low-resolution nature of $\mathbf{F_V}$.
- To capture the geometric information on surface, we extract the fine-grained surface-aligned features $\mathbf{f}_{\mathcal{S}}$ of a query point $\mathbf{X}$ as the weighted sum of the associated features $\mathbf{F}_{\mathcal{S}}$ of its K nearest neighbors in a point cloud $\mathcal{S}$, which is reconstructed from the input image $\mathbf{I}$ using a point set generator $G_{\mathcal{S}}$.

Thus, our PVSeRF is conditioned on $\mathbf{f_I}$, $\mathbf{f_V}$, and $\mathbf{f}_{\mathcal{S}}$ and can be reformulated as:

$$\sigma, \mathbf{c} = \text{PVSeRF}(\mathbf{X}, \mathbf{d}; \mathbf{f_I} \oplus \mathbf{f_V} \oplus \mathbf{f}_{\mathcal{S}}) \tag{5}$$

where $\oplus$ denotes a concatenation operation. Thanks to the incorporation of $\mathbf{f_V}$ and $\mathbf{f}_{\mathcal{S}}$, the previously ambiguous points that share the same $\mathbf{f_I}$ are now separable by the concatenation $\mathbf{f_I} \oplus \mathbf{f_V} \oplus \mathbf{f}_{\mathcal{S}}$. We present more details about each component of our method as follows.

3.3 Feature Extraction

Pixel-aligned Features Following pixelNeRF [71], we also use pixel-aligned features that contain fine-grained details about the scene's geometry and appearance properties to learn neural radiance fields. Given an input image $\mathbf{I} \in \mathbb{R}^{H\times W\times 3}$, we employ a fully-convolutional image encoder E implemented by ResNet-34 [17] to extract its multi-scale feature maps $\{\mathbf{F_I^0}, \mathbf{F_I^1}, \mathbf{F_I^2}, \mathbf{F_I^3}\}$, which are the intermediate features at 'conv1', 'layer1', 'layer2', and 'layer3' of ResNet-34 but upsampled to the size of the input image $\mathbf{I}$. Then, we acquire the pixel-aligned feature vector $\mathbf{f_I}$ of a query 3D point $\mathbf{X}$ by projecting $\mathbf{X}$ to the 2D image coordinates x, and bilinearly interpolating the feature maps concatenated by $\{\mathbf{F_I^0}, \mathbf{F_I^1}, \mathbf{F_I^2}, \mathbf{F_I^3}\}$ through $\mathcal{B}$:

$$\mathbf{f_I} = \mathcal{B}(\mathbf{F_I^0} \oplus \mathbf{F_I^1} \oplus \mathbf{F_I^2} \oplus \mathbf{F_I^3}, \mathbf{K[Rt]X}) \tag{6}$$

where $\oplus$ represents feature concatenation. However, $\mathbf{K[Rt]}$ may project multiple 3D points $\mathbf{X}$ along the viewing direction of input image to a single position on the 2D image coordinates, leading to ambiguous $\mathbf{f_I}$ and blurry synthesized novel views. To clarify such ambiguity, we propose to augment $\mathbf{f_I}$ with complementary geometric features, including both coarse voxel-aligned features learned from a volumetric grid, and fine surface-aligned features extracted from a regressed point cloud.

Voxel-aligned Features We compute the voxel-aligned feature $\mathbf{f_V}$ with respect to $\mathbf{X}$ as follows. First, we reconstruct a volumetric feature grid $\mathbf{F_V} \in \mathbb{R}^{32\times 32\times 32\times C}$ from the input image $\mathbf{I}$ using a volume generator consisting of a VGG-16 [47] image encoder and a 3D CNN decoder. Then, we have:

$$\mathbf{f_V} = \mathcal{T}(\mathbf{F_V}, \Omega(\mathbf{X})) \tag{7}$$

where $\mathcal{T}$ is a multi-scale trilinear interpolation inspired by GeoPiFu [18] and IFNet [6], $\Omega(\mathbf{X})$ is a point set around $\mathbf{X}$:

$$\Omega(\mathbf{X}) = \{\mathbf{X} + s \cdot \mathbf{n} | \mathbf{n} = (1,0,0), (0,1,0), (0,0,1), \ldots\} \tag{8}$$

where $s \in \mathbb{R}$ is the step length, $\mathbf{n} \in \mathbb{R}^3$ represents the unit vectors defined along the three axes in a Cartesian coordinate system. Intuitively, $\mathbf{f_V}$ is a concatenation of all queried feature vectors at points in $\Omega(\mathbf{X})$ that are trilinearly interpolated from $\mathbf{F_V}$.

		plane	bench	cbnt.	car	chair	disp.	lamp	spkr.	rifle	sofa	table	phone	boat	mean
↑ PSNR	DVR [38]	25.29	22.64	24.47	23.95	19.91	20.86	23.27	20.78	23.44	23.35	21.53	24.18	25.09	22.70
	SRN [49]	26.62	22.20	23.42	24.40	21.85	19.07	22.17	21.04	24.95	23.65	22.45	20.87	25.86	23.28
	pixelNeRF [71]	29.76	26.35	27.72	27.58	23.84	24.22	28.58	24.44	30.60	26.94	25.59	**27.13**	29.18	26.80
	Ours	**31.32**	**27.43**	**28.40**	**28.12**	**24.37**	**24.61**	**28.73**	**24.44**	**30.82**	**27.42**	**26.60**	26.99	**29.92**	**27.48**
↑ SSIM	DVR [38]	0.905	0.866	0.877	0.909	0.787	0.814	0.849	0.798	0.916	0.868	0.840	0.892	0.902	0.860
	SRN [49]	0.901	0.837	0.831	0.897	0.814	0.744	0.801	0.779	0.913	0.851	0.828	0.811	0.898	0.849
	pixelNeRF [71]	0.947	0.911	0.910	**0.942**	0.858	0.867	0.913	**0.855**	**0.968**	0.908	0.898	**0.922**	0.939	0.910
	Ours	**0.956**	**0.923**	**0.912**	0.940	**0.869**	**0.867**	**0.915**	0.853	0.965	**0.912**	**0.911**	0.915	**0.940**	**0.915**
↓ LPIPS	DVR [38]	0.095	0.129	0.125	0.098	0.173	0.150	0.172	0.170	0.094	0.119	0.139	0.110	0.116	0.130
	SRN [49]	0.111	0.150	0.147	0.115	0.152	0.197	0.210	0.178	0.111	0.129	0.135	0.165	0.134	0.139
	pixelNeRF [71]	0.084	0.116	0.105	0.095	0.146	**0.129**	0.114	0.141	0.066	0.116	0.098	**0.097**	0.111	0.108
	Ours	**0.065**	**0.098**	**0.097**	**0.087**	**0.128**	0.133	**0.104**	**0.140**	**0.066**	**0.104**	**0.082**	0.107	**0.101**	**0.096**

Table 1: Quantitative comparison on category-agnostic view synthesis. The best quantitative values are marked as boldface. Our method outperforms all baselines by a wide margin in terms of all mean metrics.

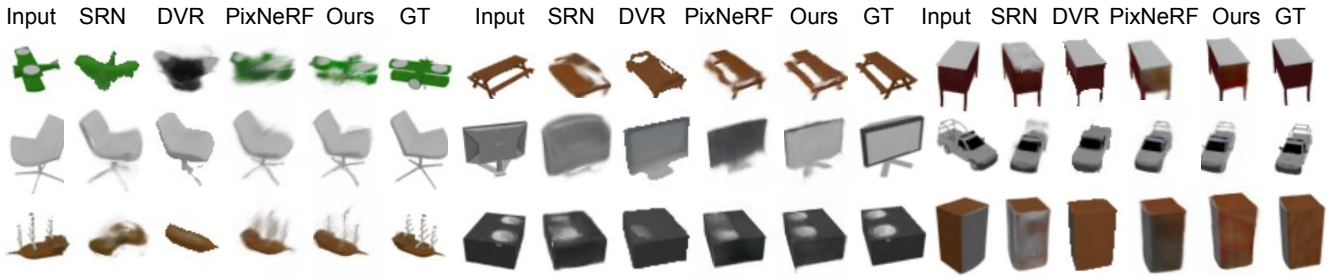

Figure 4: Qualitative comparison on category-agnostic view synthesis. A single model is trained among 13 ShapeNet categories, and tested on a single image for novel view synthesis. We observe that our method produces detailed novel views, and is consistent in both geometry and appearance. Conversely, pixelNeRF [71] fails to infer correct geometry and produce inconsistent and blurry textures.

Surface-aligned Features Although they capture a global context about the shape of a 3D object, voxel-aligned features are queried from a low-resolution volumetric grid and thus lack geometric information on surface. As a complement, we introduce surface-aligned features that capture fine details of surface to facilitate radiance field learning. Given an input image $\mathbf{I}$, we first regress a sparse point cloud $\mathcal{S}$ of size 1024 from $\mathbf{I}$ using a point set generator $G_{\mathcal{S}}$ based on GraphX-convolutions [35]. Then, we feed the generated point cloud to a PointNet++ [45] network to extract point-wise features $\mathbf{F}_{\mathcal{S}}$. For each query point $\mathbf{X}$, we define its surface-aligned feature $\mathbf{f}_{\mathcal{S}}$ as the weighted sum of the corresponding feature vectors of $\mathbf{X}$'s K-nearest neighbors in $\mathcal{S}$:

$$\mathbf{f}_{\mathcal{S}} = \sum_{k=0}^{K} w_k * \mathbf{F}_{\mathcal{S}_{m(k)}} \tag{9}$$

where $m(k), k = 0, 1, 2..., K$ is the indices of the K points, w_k is inversely proportional to its distance to $\mathbf{X}$:

$$w_k = 1/(1 + \exp(||\mathbf{X} - \mathcal{S}_{m(k)}||) \tag{10}$$

In this way, the features from the nearest neighbor contributes most to the $\mathbf{f}_{\mathcal{S}}$, and vice versa.

3.4 Radiance Field Prediction and Rendering

We parameterize our PVSeRF framework using a MLP f which regresses the volume density σ and view-dependent radiance $\mathbf{r}$ from the 3D coordinates of a query point $\mathbf{X}$, a viewing direction $\mathbf{d}$, and the corresponding pixel-, voxel-, and surface-aligned features (i.e. $\mathbf{f}_{\mathbf{I}}$, $\mathbf{f}_{\mathbf{V}}$, and $\mathbf{f}_{\mathbf{S}}$) extracted from the input single-view image $\mathbf{I}$:

$$\sigma, \mathbf{c} = f(\gamma_m(\mathbf{X}), \gamma_n(\mathbf{d}); \mathbf{f}_{\mathbf{I}} \oplus \mathbf{f}_{\mathbf{V}} \oplus \mathbf{f}_{\mathbf{S}}) \tag{11}$$

where γ_m and γ_n are position encoding functions [32, 59] applied to $\mathbf{X}$, $\mathbf{d}$ respectively, which alleviates the positional bias inherent in Cartesian coordinates without sacrificing their discrepancy in-between. Specifically, γ maps Cartesian coordinates from $\mathbb{R}$ into a high dimensional space $\mathbb{R}^{2L}$:

$$\begin{aligned}\gamma_L(\mathbf{p}) = (&\sin(2^0\pi\mathbf{p}), \cos(2^0\pi\mathbf{p}), \\ &..., \sin(2^{L-1}\pi\mathbf{p}), \cos(2^{L-1}\pi\mathbf{p}))\end{aligned} \tag{12}$$

where $\gamma(\cdot)$ is applied separately to each component of vector $\mathbf{p}$. With the constructed radiance field represented by σ and $\mathbf{c}$, we render novel view images via the numerical quadrature approximation of differentiable volume rendering techniques which illustrated in Section 3.1.

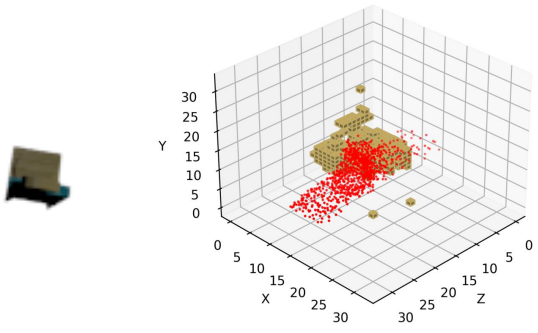

Figure 5: Failure case of explicit geometry reasoning. Under the challenging viewpoint, the scene geometry is ambiguously captured in a single image, causing the network being unable to predict plausible geometries.

3.5 Loss Functions

Corresponding to our pixel-, voxel- and surface-aligned features, we train our model using three different loss functions as follows.

RGB Rendering Loss Similar to existing works in the NeRF series, we use *L2* rendering loss as the main loss function. It constrains that the rendered color value of each ray should be consistent with the corresponding ground-truth pixel value. Thus, we have:

$$L_r = \|\hat{\mathbf{C}}(\mathbf{r}) - \mathbf{C}(\mathbf{r})\|_2^2 \tag{13}$$

where $\hat{\mathbf{C}}(\mathbf{r})$ and $\mathbf{C}(\mathbf{r})$ are the predicted and ground-truth color values of sampled pixels from novel view $\mathbf{I}_{\mathbf{novel}}$ with viewpoint $[\mathbf{Rt}]_{\mathbf{novel}}$ respectively.

Volume Reconstruction Loss To learn volumetric features $\mathbf{F_V}$, we add a 3D convolutional layer after $\mathbf{F_V}$ to estimate a low-resolution occupancy volume $\mathbf{V} \in \mathbb{R}^{32\times32\times32}$, whose ground-truth label is $\mathbf{V}^*$. Then, we apply a standard binary cross-entropy loss and have:

$$L_v = \sum_{i\in[1:32]^3} \mathbf{V}^*(i)\log\mathbf{V}(i) + (1-\mathbf{V}^*(i))\log(1-\mathbf{V}(i)) \tag{14}$$

Point Regression Loss We employ the Chamfer distance to constraint our point set generation and have:

$$L_p = \sum_{\mathbf{q}\in\mathcal{S}} \min_{\mathbf{q}^*\in\mathcal{S}^*} \|\mathbf{q}-\mathbf{q}^*\|^2 + \sum_{\mathbf{q}^*\in\mathcal{S}^*} \min_{\mathbf{q}\in\mathcal{S}} \|\mathbf{q}-\mathbf{q}^*\|^2 \tag{15}$$

where $\mathcal{S}$ is the predicted point set and $\mathcal{S}^*$ is its corresponding ground truth.

Overall Loss Function Our overall loss function is:

$$L = \lambda_1 * L_r + \lambda_2 * L_v + \lambda_3 * L_p \tag{16}$$

where λ_1, λ_2, and λ_3 are weighting parameters.

4 EXPERIMENTS

To demonstrate the superiority of our PVSeRF, we first compare it against state-of-the-art methods on two single-image novel view synthesis tasks, i.e. category-agnostic view synthesis and category-specific view synthesis. Then, we evaluate our approach on real images, demonstrating the generalization ability of our method. Finally, we conduct ablation studies to validate the effectiveness of each component of our PVSeRF.

Datasets We benchmark our method extensively on the synthetic images from the ShapeNet [2] dataset. Specifically, for the category-agnostic view synthesis task, we use the renderings and splits from Kato *et al.* [21] which renders objects from 13 categories of the ShapeNetCore-V1 dataset. Each object was rendered at 64×64 resolution from 24 equidistant azimuth angles, with a fixed elevation angle. For the category-specific view synthesis task, we use the dataset and splits provided by Sitzmann *et al.* [49], which renders 6,591 chairs and 3,514 cars from the ShapeNetCore V2 dataset. For the evaluation on real images, we use the collected real-world cars images from [25]. To provide supervision for volume reconstruction and point set regression, we convert each ground-truth mesh to a point set of size 2048 and a volumetric grid of resolution 32^3.

Implementation Details We implement our model with PyTorch [43]. Details of the network architecture are presented in the supplementary material. The training process of our approach consists of two stages: i) we pre-train the volume generator $G_{\mathbf{V}}$ and the point set generator $G_{\mathbf{S}}$ respectively using loss functions defined in Eq. 14 and Eq. 15. Specifically, $G_{\mathbf{V}}$ is trained with an initial learning rate of 10^{-3} and a batch size of 64 for 250 epochs. The learning rate drops by a factor of 5 after 150 epochs. $G_{\mathcal{S}}$ is trained with an initial learning rate of 10^{-5} and a batch size of 4 for 10 epochs. The learning rate drops by a factor of 3 after 5 and 8 epochs. ii) we fine-tune the whole network for 400 epochs. We set the learning rate as 10^{-4} and the batch size as 4. We use an Adam [22] optimizer for all the training mentioned above. We empirically set the multi-scale trilinear interpolation step length $s = 0.0722$, the number of nearest neighbors $K = 5$, the number of frequencies of positional encoding for $\mathbf{X}$, $\mathbf{d}$ as $m = 6$, $n = 0$, and the weights for loss function as $\lambda_1 = \lambda_2 = \lambda_3 = 1$.

Evaluation Protocol Following the community standards [32, 38, 49], we use peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) [62] to measure the quality of the synthesized novel views. We also use LPIPS [72] that has been shown to be closer to human perception.

4.1 Category-agnostic View Synthesis

Category-agnostic novel view synthesis aims to learn object priors that can generalize across multiple categories.

Baselines We compare our method against three closely-related state-of-the-art methods: SRN [49], DVR [38] and pixelNeRF [71], which are applicable to synthesize novel views for all categories. For DVR and pixelNeRF, we use pretrained models from their official Github repositories[1]. For SRN [49], we use the model trained by [71] to make it comparable with [49, 71]. All methods are trained using the same dataset and settings introduced in Sec. 4. To facilitate a fair comparison, we follow the random view indices provided by pixelNeRF and select the input view for each test object accordingly.

Results As Fig. 4 shows, our method outperforms all previous methods by synthesizing more detailed novel views. In addition, it

[1]Niemeyer *et al.* [38]: https://github.com/autonomousvision/differentiable_volumetric_rendering, Yu *et al.* [71]: https://github.com/sxyu/pixel-nerf.

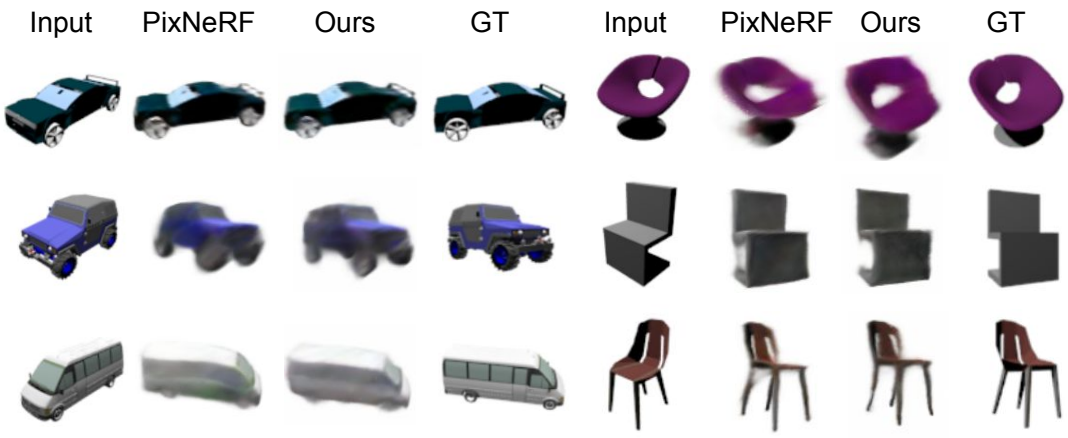

Figure 6: Qualitative comparison on category-specific view synthesis. The performance of our method is comparable to that of the state-of-the-art pixelNeRF [71].

		PSNR↑	SSIM↑
Chairs	TCO [53]	21.27	0.88
	dGQN [10]	21.59	0.87
	SRN [49]	22.89	0.89
	pixelNeRF [71]	**23.72**	**0.91**
	Ours	23.33	**0.91**
Cars	SRN [49]	22.25	0.89
	pixelNeRF [71]	**23.17**	**0.90**
	Ours	22.98	**0.90**

Table 2: Quantitative comparison on category-specific view synthesis. Since the renderings from [49] contain many challenging camera viewpoints, our performance is degenerated ascribe to the invalidity of explicit geometry reasoning. Nevertheless, our method is comparable to the state-of-the-art method [71].

can be observed that i) the two baseline methods, DVR [38] and SRN [49], tend to generate blurry images and distorted geometries; ii) pixelNeRF [71] shows blurry and inconsistent appearance. The quantitative results in Table 1 further justify the superiority of our method against all baselines in terms of the mean values of PSNR, SSIM and LPIPS metrics. Notably, the PSNR of our approach attains a significant improvement over the second best method by 0.68.

4.2 Category-specific View Synthesis

For category-specific view synthesis, all methods are trained on the chair or car categories of ShapeNet [2].

Baselines We choose SRN [48] and pixelNeRF [71] as the baseline methods. We also report the quantitative results from TCO [53] and dGQN [10] provided by [49], to keep in line with prior arts.
Results We show the quantitative and qualitative results in Table 2 and Fig. 6 respectively. It can be observed that the performance of our method is comparable to the state-of-the-art method [71] both qualitatively and quantitatively. Such comparable results indicate that the advantages of our method are not significant in some neural rendering cases. We carefully investigate the results and ascribe this to the invalidity of explicit geometry reasoning in some cases (Fig. 5). Since the renderings provided by [49] contain many challenging camera viewpoints, the explicit geometry reasoning from single-view becomes a more challenging problem. We postpone the discussion of this phenomenon to Sec. 5.

4.3 Novel View Synthesis on Real Images

To highlight the generalization ability of our method, we evaluate our pretrained models directly on real images without any finetuning. Specifically, we first take the images from the Stanford cars dataset [25] and apply the PointRend model [23] to mask their clutter backgrounds. Then, we feed the preprocessed images into a category-specific model of ShapeNet "cars" to predict novel views. As Fig. 7 shows, our method can not only synthesize visually compelling novel views, but also infer accurate geometries. This effectively demonstrates the excellent generalization performance of our method on real image as it is only trained on synthetic images.

4.4 Ablation Study

To validate the effectiveness of each proposed component, we conduct an ablation study on our method, yielding three variants: i) *w/o surface-aligned feature*, in which only pixel- and voxel-aligned features are incorporated; ii) *w/o voxel-aligned feature*, where the radiance field is conditioned only on pixel- and surface-aligned features; iii) *w/o joint training*, in which we fix all feature extractors[2] and solely train the radiance field predictor f. As Table 3 shows, it can be observed that the *w/o joint training* variant constantly performs the worst among all variants. This demonstrates that the joint learning of pixel-, voxel- and surface-aligned features is crucial in our explicit geometric reasoning. In addition, the performance of the *w/o surface-aligned* variant is always worse than the *w/o voxel-aligned feature* variant, as the voxel-aligned features queried from a low-resolution volume are better at capturing global geometry

[2]We use a PointNet++ [45] model trained on PartNet [33] segmentation task as a point-feature extractor.

Figure 7: Novel view synthesis results on real car images. Although extensively trained on synthetic data, our method can easily generalize to real single-view images, and produce plausible view synthesis results and underlying geometries.

		plane	bench	cbnt.	car	chair	disp.	lamp	spkr.	rifle	sofa	table	phone	boat	mean
	w/o joint	29.03	25.18	26.12	25.82	21.97	22.25	26.33	22.19	28.55	25.18	24.27	24.67	27.54	25.16
	w/o surface-aligned	30.82	27.00	28.31	27.67	24.05	24.33	28.73	24.33	30.63	26.97	26.27	26.85	29.58	27.15
↑ PSNR	w/o voxel-aligned	30.83	27.14	**28.40**	27.93	24.35	**24.66**	**29.10**	**24.75**	**31.05**	27.29	26.48	**27.01**	29.61	27.38
	Ours	**31.32**	**27.43**	**28.40**	**28.12**	**24.37**	24.61	28.73	24.44	30.82	**27.42**	**26.60**	26.99	**29.92**	**27.48**
	w/o joint	0.926	0.863	0.864	0.924	0.844	0.803	0.826	0.812	0.947	0.878	0.853	0.836	0.923	0.876
	w/o surface-aligned	0.954	0.917	0.912	0.936	0.860	0.860	**0.915**	0.849	**0.967**	0.904	0.906	0.912	**0.940**	0.911
↑ SSIM	w/o voxel-aligned	0.955	0.921	**0.913**	**0.941**	0.867	**0.870**	**0.915**	**0.856**	0.966	0.911	0.910	**0.917**	0.939	**0.915**
	Ours	**0.956**	**0.923**	0.912	0.940	**0.869**	0.867	**0.915**	0.853	0.965	**0.912**	**0.911**	0.915	**0.940**	**0.915**
	w/o joint	0.097	0.134	0.134	0.099	0.137	0.163	0.175	0.164	0.098	0.123	0.120	0.149	0.120	0.123
	w/o surface-aligned	0.064	0.100	**0.096**	0.094	0.136	0.132	0.102	0.144	0.064	0.108	0.085	**0.100**	**0.100**	0.099
↓ LPIPS	w/o voxel-aligned	**0.063**	**0.096**	**0.096**	**0.085**	0.130	**0.126**	**0.100**	**0.137**	**0.060**	**0.104**	**0.081**	0.104	**0.100**	**0.095**
	Ours	0.065	0.098	0.097	0.087	**0.128**	0.133	0.104	0.140	0.066	**0.104**	0.082	0.107	0.101	0.096

Table 3: Quantitative comparison of ablation studies. Our joint method that employs complementary coarse volumetric features and fine surface features achieves the best performance. Whereas, removing any part of the proposed method will cause more or less deterioration.

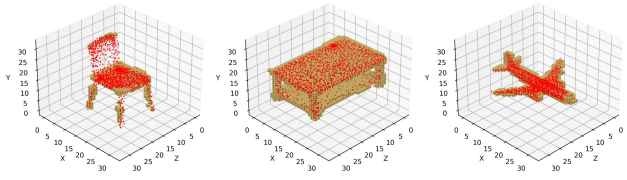

Figure 8: Illustration of the complementary properties of point set and volumes. We randomly show several predicted geometries. It can be seen that these two representations exhibit reciprocal behaviors: the missing parts of volumetric grid are spanned by point set, while the regions where the point set is too sparse are occupied by volumes.

contexts. Our full method achieves the best performance among all variants, which validates the effectiveness of employing a hybrid of geometric features that complement each other [61]. This is also demonstrated in Fig. 8.

5 CONCLUSION

For the task of novel view synthesis from single-view RGB images, we present PVSeRF, a novel learning framework that reconstructs neural radiance fields conditioned on joint pixel-, voxel-, and surface-aligned features. By augmenting hybrid geometric features with image features, we effectively address the feature confusion issue of pixel-aligned features. Compare to previous arts, our framework gains superior or comparable results in terms of both visual perception and quantitative measures. Moreover, a suite of ablation studies also verify the efficacy of our key contributions.

Limitation and Future Works Despite the effectiveness of our method, there are still some limitations to be addressed in future work. First, the performance of our method is dependent on the amount of geometric information within the input single-view image. As discussed in Sec. 4.2, when the scene geometry is little captured in the input image due to challenging viewpoints, the novel views synthesized by our method may become less clear. In future work, we plan to include multi-view consistency as an additional supervision to train our geometry reasoning network, thereby increasing its robustness to challenging viewpoints. Secondly, we focus on the geometry reasoning from complete geometries (i.e. surface and voxel) of 3D shapes that reconstructs neural radiance fields from single-view RGB image and have not investigated that from more challenging partial geometries (e.g. depth maps or multiplane images [58, 64]). In future work, we plan to extend our method to such partial geometries, thereby making our method more flexible.

REFERENCES

[1] Neill DF Campbell, George Vogiatzis, Carlos Hernández, and Roberto Cipolla. 2008. Using multiple hypotheses to improve depth-maps for multi-view stereo. In *European Conference on Computer Vision*. Springer, 766–779.

[2] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. 2015. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012* (2015).

[3] R. Qi Charles, Hao Su, Mo Kaichun, and Leonidas J. Guibas. 2017. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *CVPR*.

[4] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. 2021. MVSNeRF: Fast Generalizable Radiance Field Reconstruction from Multi-View Stereo. *ArXiv* abs/2103.15595 (2021).

[5] Zhiqin Chen and Hao Zhang. 2019. Learning implicit fields for generative shape modeling. In *CVPR*.

[6] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. 2020. Implicit functions in feature space for 3d shape reconstruction and completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6970–6981.

[7] Julian Chibane, Aayush Bansal, Verica Lazova, and Gerard Pons-Moll. 2021. Stereo Radiance Fields (SRF): Learning View Synthesis for Sparse Views of Novel Scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7911–7920.

[8] Christopher Bongsoo Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 2016. 3D-R2N2: A Unified Approach for Single and Multi-view 3D Object Reconstruction. In *ECCV*.

[9] Abe Davis, Marc Levoy, and Fredo Durand. 2012. Unstructured light fields. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, 305–314.

[10] SM Ali Eslami, Danilo Jimenez Rezende, Frederic Besse, Fabio Viola, Ari S Morcos, Marta Garnelo, Avraham Ruderman, Andrei A Rusu, Ivo Danihelka, Karol Gregor, et al. 2018. Neural scene representation and rendering. *Science* 360, 6394 (2018), 1204–1210.

[11] Haoqiang Fan, Hao Su, and Leonidas J. Guibas. 2017. A Point Set Generation Network for 3D Object Reconstruction from a Single Image. In *CVPR*.

[12] Jiahao Geng, Tianjia Shao, Youyi Zheng, Yanlin Weng, and Kun Zhou. 2018. Warp-guided GANs for single-photo facial animation. *ACM Transactions on Graphics (TOG)* 37 (2018), 1 – 12.

[13] Steven J Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F Cohen. 1996. The lumigraph. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. 43–54.

[14] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan C. Russell, and Mathieu Aubry. 2018. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In *CVPR*.

[15] Xiaoguang Han, Zhen Li, Haibin Huang, Evangelos Kalogerakis, and Yizhou Yu. 2017. High-resolution shape completion using deep neural networks for global structure and local geometry inference. In *ICCV*.

[16] Richard Hartley and Andrew Zisserman. 2000. *Multiple View Geometry in Computer Vision*.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[18] Tong He, John Collomosse, Hailin Jin, and Stefano Soatto. 2020. Geo-pifu: Geometry and pixel aligned implicit functions for single-view human reconstruction. *arXiv preprint arXiv:2006.08072* (2020).

[19] Heiko Hirschmuller. 2007. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence* 30, 2 (2007), 328–341.

[20] James T. Kajiya and Brian Von Herzen. 1984. Ray tracing volume densities. *Proceedings of the 11th annual conference on Computer graphics and interactive techniques* (1984).

[21] Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. 2018. Neural 3d mesh renderer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3907–3916.

[22] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[23] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross B. Girshick. 2020. PointRend: Image Segmentation As Rendering. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020), 9796–9805.

[24] Jean Kossaifi, Linh Hoang Tran, Yannis Panagakis, and Maja Pantic. 2018. GAGAN: Geometry-Aware Generative Adversarial Networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018), 878–887.

[25] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 2013. 3D Object Representations for Fine-Grained Categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*. Sydney, Australia.

[26] Marc Levoy and Pat Hanrahan. 1996. Light field rendering. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. 31–42.

[27] David B. Lindell, Julien N. P. Martel, and Gordon Wetzstein. 2021. AutoInt: Automatic Integration for Fast Neural Volume Rendering. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), 14551–14560.

[28] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. 2020. Neural Sparse Voxel Fields. *ArXiv* abs/2007.11571 (2020).

[29] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. 2021. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), 7206–7215.

[30] Nelson L. Max. 1995. Optical Models for Direct Volume Rendering. *IEEE Trans. Vis. Comput. Graph.* 1 (1995), 99–108.

[31] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. 2019. Occupancy networks: Learning 3d reconstruction in function space. In *CVPR*.

[32] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2020. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*. Springer, 405–421.

[33] Kaichun Mo, Shilin Zhu, Angel X. Chang, Li Yi, Subarna Tripathi, Leonidas J. Guibas, and Hao Su. 2019. PartNet: A Large-Scale Benchmark for Fine-Grained and Hierarchical Part-Level 3D Object Understanding. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[34] Thomas Neff, Pascal Stadlbauer, Mathias Parger, Andreas Kurz, Joerg H. Mueller, Chakravarty Reddy Alla Chaitanya, Anton Kaplanyan, and Markus Steinberger. 2021. DONeRF: Towards Real-Time Rendering of Compact Neural Radiance Fields using Depth Oracle Networks. *Computer Graphics Forum* 40 (2021).

[35] Anh-Duc Nguyen, Seonghwa Choi, Woojae Kim, and Sanghoon Lee. 2019. GraphX-convolution for point cloud deformation in 2D-to-3D conversion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8628–8637.

[36] Thu Nguyen-Phuoc, Chuan Li, Stephen Balaban, and Yongliang Yang. 2018. RenderNet: A deep convolutional network for differentiable rendering from 3D shapes. In *NeurIPS*.

[37] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yongliang Yang. 2019. HoloGAN: Unsupervised Learning of 3D Representations From Natural Images. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (2019), 7587–7596.

[38] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. 2020. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3504–3515.

[39] Junyi Pan, Xiaoguang Han, Weikai Chen, Jiapeng Tang, and Kui Jia. 2019. Deep Mesh Reconstruction from Single RGB Images via Topology Modification Networks. In *ICCV*.

[40] Eunbyung Park, Jimei Yang, Ersin Yumer, Duygu Ceylan, and Alexander C Berg. 2017. Transformation-grounded image generation network for novel 3d view synthesis. In *Proceedings of the ieee conference on computer vision and pattern recognition*. 3500–3509.

[41] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. 2019. DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. In *CVPR*.

[42] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. 2021. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5865–5874.

[43] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.). Curran Associates, Inc., 8024–8035. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[44] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. 2020. Convolutional occupancy networks. In *ECCV*.

[45] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*.

[46] Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. 2021. KiloNeRF: Speeding up Neural Radiance Fields with Thousands of Tiny MLPs. *arXiv preprint arXiv:2103.13744* (2021).

[47] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[48] Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhofer. 2019. Deepvoxels: Learning persistent 3d feature embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2437–2446.

[49] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. 2019. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *arXiv preprint arXiv:1906.01618* (2019).

[50] Jiapeng Tang, Xiaoguang Han, Junyi Pan, Kui Jia, and Xin Tong. 2019. A skeleton-bridged deep learning approach for generating meshes of complex topologies from single rgb images. In *CVPR*.
[51] Jiapeng Tang, Xiaoguang Han, Mingkui Tan, Xin Tong, and Kui Jia. 2020. SkeletonNet: A Topology-Preserving Solution for Learning Mesh Reconstruction of Object Surfaces from RGB Images. *arXiv preprint arXiv:2008.05742* (2020).
[52] Jiapeng Tang, Jiabao Lei, Dan Xu, Feiying Ma, Kui Jia, and Lei Zhang. 2021. SA-ConvONet: Sign-Agnostic Optimization of Convolutional Occupancy Networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 6504–6513.
[53] Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox. 2015. Single-view to Multi-view: Reconstructing Unseen Views with a Convolutional Network. *ArXiv* abs/1511.06702 (2015).
[54] Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox. 2016. Multi-view 3d models from single images with a convolutional network. In *European Conference on Computer Vision*. Springer, 322–337.
[55] Engin Tola, Christoph Strecha, and Pascal Fua. 2012. Efficient large-scale multi-view stereo for ultra high-resolution image sets. *Machine Vision and Applications* 23, 5 (2012), 903–920.
[56] Edgar Tretschk, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Christoph Lassner, and Christian Theobalt. 2020. Non-Rigid Neural Radiance Fields: Reconstruction and Novel View Synthesis of a Deforming Scene from Monocular Video. *https://arxiv.org/abs/2012.12247* (2020).
[57] Alex Trevithick and Bo Yang. 2020. Grf: Learning a general radiance field for 3d scene representation and rendering. *arXiv preprint arXiv:2010.04595* (2020).
[58] Richard Tucker and Noah Snavely. 2020. Single-view view synthesis with multi-plane images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 551–560.
[59] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.
[60] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. 2021. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4690–4699.
[61] Yang Wang. 2021. Survey on deep multi-modal data analytics: Collaboration, rivalry, and fusion. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 17, 1s (2021), 1–25.
[62] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* 13, 4 (2004), 600–612.
[63] Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. 2020. Synsin: End-to-end view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7467–7477.
[64] Suttisak Wizadwongsa, Pakkapon Phongthawee, Jiraphon Yenphraphai, and Supasorn Suwajanakorn. 2021. Nex: Real-time view synthesis with neural basis expansion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8534–8543.
[65] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. 2016. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *NeurIPS*.
[66] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 2015. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1912–1920.
[67] Haozhe Xie, Hongxun Yao, Xiaoshuai Sun, Shangchen Zhou, Shengping Zhang, and Xiaojun Tong. 2019. Pix2Vox: Context-Aware 3D Reconstruction From Single and Multi-View Images. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (2019), 2690–2698.
[68] Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. 2018. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 206–215.
[69] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Ronen Basri, and Yaron Lipman. 2020. Multiview neural surface reconstruction by disentangling geometry and appearance. *arXiv preprint arXiv:2003.09852* (2020).
[70] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. 2021. Plenoctrees for real-time rendering of neural radiance fields. *arXiv preprint arXiv:2103.14024* (2021).
[71] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. 2021. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4578–4587.
[72] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 586–595.
[73] Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A Efros. 2016. View synthesis by appearance flow. In *European conference on computer vision*. Springer, 286–301.
[74] Jun-Yan Zhu, Zhoutong Zhang, Chengkai Zhang, Jiajun Wu, Antonio Torralba, Joshua B. Tenenbaum, and Bill Freeman. 2018. Visual Object Networks: Image Generation with Disentangled 3D Representations. In *NeurIPS*.

A TRAINING COMPLEXITY

For our category-agnostic experiments, the training takes 11 days on 2 RTX 3090: 1 day for the pre-training of $G_{\mathbf{V}}$ and $G_{\mathcal{S}}$, and 10 days for the fine-tuning the entire network. For our category-specific experiments, the total training time is 7 days 12 hours on a single RTX 3090: 12 hours for the pre-training and 7 days for the fine-tuning.

B DETAILS OF NETWORK ARCHITECTURE

Image Encoder E. We implement the image encoder using the pretrained ResNet-34 [17] architecture with batch normalization. To capture multi-scale features, we employ the 'conv1', 'layer1', 'layer2', and 'layer3' of ResNet-34. The default max-pooling layer after 'conv1' is removed to keep a feasible feature size at deep layers. Therefore, for a $H \times W$ image, we have four scales of feature maps, i.e. feature maps with spatial size $H/2^L \times W/2^L, L = 1, 2, 3, 4$. Finally, all feature maps are upscaled to $H/2 \times W/2$ and composited as the final image features.

Volume Generator $G_{\mathbf{V}}$. The implementation of our volume generator is similar to the encoder-decoder architecture introduced in Pix2Vox [67]. Specifically, we use the Pix2Vox-F architecture as shown in Fig. 9. To construct the volumetric feature grid $\mathbf{F_V}$, we concatenate the generated volume and the features of the second last layer, so the final feature grid is of size $32 \times 32 \times 32 \times 9$. Note that for 64×64 images, we remove the last max-pooling operation between convolutional layers, to avoid the spatial feature size shrinking too much at the bottleneck.

Point Set Generator $G_{\mathcal{S}}$ and Feature Extractor. We regress a point set of size $N = 2048$ from the input image by leveraging the PCDNet introduced in [35]. We refer the reader to [35] for more details. For our point-wise feature extractor, we use a similar network originated from PointNet++ [45]. Specifically, we use the network structure for semantic and part segmentation, which comprises three set abstraction layers and three feature propagation layers. Let $\mathrm{SA}(K, r, [l_1, ..., l_d])$ denotes the set abstraction (SA) layer with K local regions of ball radius r, using the PointNet [3] architecture of d fully connected layers with width $l_i (i = 1, ..., d)$; and $\mathrm{FP}(l_1, ..., l_d)$ denotes the feature propagation (FP) layer; the network architecture of our point-wise feature extractor can be described as:

$$
\begin{gathered}
SA(512, 0.2, [64, 64, 128]) \\
\Downarrow \\
SA(128, 0.4, [128, 128, 256]) \\
\Downarrow \\
SA([256, 512, 1024]) \\
\Downarrow \\
FP(256, 256) \\
\Downarrow \\
FP(256, 128) \\
\Downarrow \\
FP(128, 128)
\end{gathered}
$$

Therefore, the final point-wise feature is of size $B \times N \times 128$, where B is object batch size and N is point set size.

C QUALITATIVE COMPARISON FOR ABLATION STUDY

In addition to the quantitative comparison for ablation study in Table 3 of the main manuscript, we present the qualitative results in Fig. 10. It can be seen that the *w/o joint* variant shows the worst visual quality, while the *w/o surface-aligned* and the *w/o voxel-aligned* variants gradually improve the results. More importantly, our full method combines the advantages of voxel- and surface-aligned techniques, and achieves the most compelling qualitative results. This also conforms to our discussions in Section 4.4 of the main paper.

ACKNOWLEDGMENTS

The work was supported in part by the National Key R&D Program of China with grant No. 2018YFB1800800, the Basic Research Project No. HZQB-KCZYZ-2021067 of Hetao Shenzhen-HK S&T Cooperation Zone, by Shenzhen Outstanding Talents Training Fund 202002, by Guangdong Research Projects No. 2017ZT07X152 and No. 2019CX01X104, and by the Guangdong Provincial Key Laboratory of Future Networks of Intelligence (Grant No. 2022B1212010001). It was also supported by the National Key R&D Program of China with grant No. 2019YFE0110100 and Shenzhen General Project (No. JCYJ20190814112007258).

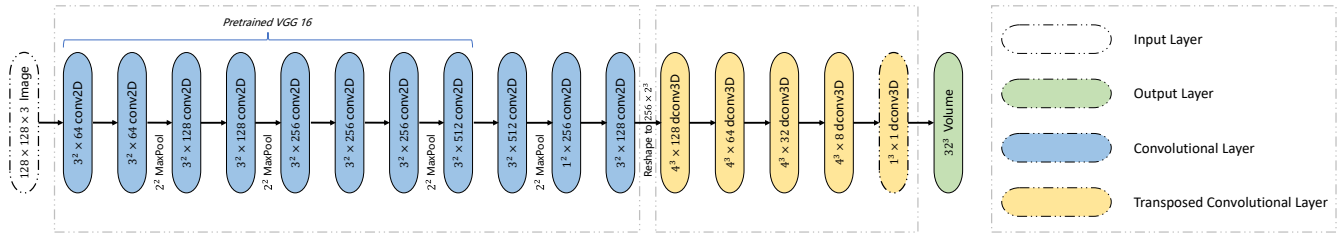

Figure 9: Network structure of volume generator G_V. We use an encoder-decoder architecture similar to the Pix2Vox-F network [67]. For 64×64 input image, the last max-pooling layer is removed.

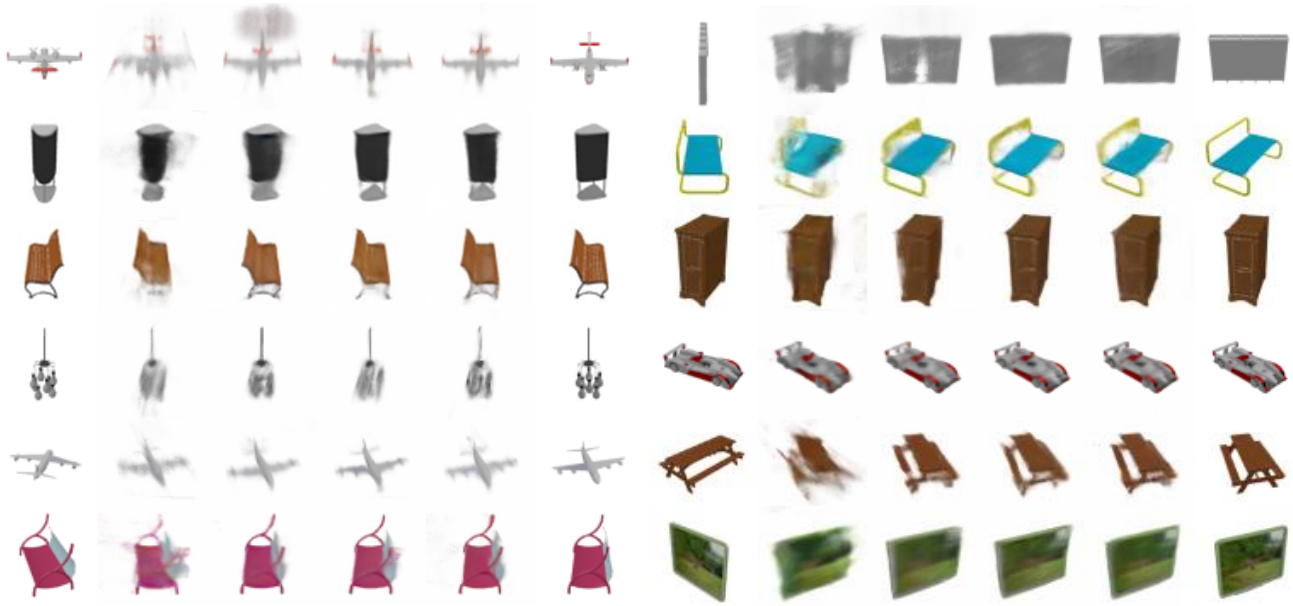

Figure 10: Qualitative comparison of ablation study. For each object, from left to right: *input image*, *w/o joint*, *w/o surface-aligned*, *w/o voxel-aligned*, *our full method*, and ground truth. Our full method exhibits the best visual quality, please see more discussions in Appendix C.