Supplementary Material for X-NeRF: Explicit Neural Radiance Field for Multi-Scene 360° Insufficient RGB-D Views

A. Minkowski Sparse Tensor

Minkowski Engine, an auto-differentiation library for sparse tensors, offers an effective way to represent and process sparse high-dimensional data structures such as point clouds. An Minkowski sparse tensor represents a D-dimensional sparse tensor $\mathscr{T} \in \mathbb{R}^{N_1 \times N_2 \times \ldots \times N_D}$ as a set of coordinates with non-zero values $\mathcal{C} = \{(x_i, y_i, z_i, t_i)\}_i$, and the associated features $\mathcal{F} = \{\mathbf{f}_i\}_i$ so that

$$\mathscr{T}\left[x_i^1, x_i^2, \cdots, x_i^D\right] = \begin{cases} \mathbf{f}_i & \text{if } \left(x_i^1, x_i^2, \cdots, x_i^D\right) \in \mathcal{C} \\ 0 & \text{otherwise} \end{cases}, \tag{1}$$

where x_d^i denotes d-th axis coordinate of the i-th non-zero element and $\mathbf{f}_i$ is the feature associated to the i-th non-zero element. It is easy to find that this representation is equivalent to the original sparse tensor $\mathcal{T} \Leftrightarrow (\mathcal{C}, \mathcal{F})$, since the non-zero elements contain all the information about $\mathcal{T}$. These sets can also be converted to matrices $\mathbf{C}, \mathbf{F}$ through the COO representation.

What we focus on is to represent a point cloud as a sparse tensor, which can be accomplished by simply discretizing the coordinates of points. The process requires a pre-defined discretization step size, or voxel size, which affects the resolution of the input sparse tensor and probably the model performance.

To represent a point cloud as a sparse tensor, which is interested in this paper, we can simply discretize the coordinates of points. This process requires defining the discretization step size, or voxel size, which is a hyper-parameter that affects the resolution of the input sparse tensor and probably the performance of a neural network.

We consider sparse tensor as a perfect way to represent multi-view RGB-D data, since we can easily build a colorful point cloud from them. With Minkowski sparse tensor, directly operating convolutions with high efficiency on point cloud is possible, which can sufficiently exploit the spatial and local information.

B. Dataset Details

We collect a dataset for our proposed multi-scene 360° insufficient RGB-D views setting. The dataset contains 10 scenes in total. In each scene, a robot arm is doing different tasks in different environments. There are 7 RGB-D cameras around the scenes. In this paper, scene 1-6 are treated as seen scenes while scene 7-10 are unseen scenes. Among the 7 views of each scene, 6 views are seen (training) views while the other one is unseen (testing) views. Fig. 1 and Fig. 2 show all the RGB-D images in our dataset.

C. Hyper-Parameter Setups

We set the voxel size as 4×10^{-3} when quantizing point clouds. In each batch, we sample 2 random image patch of size 40×40 for all 6 training views, which is equivalent to a total ray batch size of $6 \times 2 \times 40 \times 40 = 19200$. We train our models for 240 epochs with an initial learning rate of 10^{-3} and an weight decay of 10^{-5}. The learning rate is divided by 10 at 120th and 200th epoch. When sampling points on a ray, we use a step size of 0.5 voxels. When applying view augmentation, we do random rotation on point clouds in a probability of 0.15. The weighting factor of depth loss λ_{D} is set to 0.1. We set the weight of perceptual loss $\lambda_{\mathrm{percep}}$ to 1.0 so that

$$\mathcal{L}_{\mathrm{overall}} = \mathcal{L}_{\mathrm{render}} + \mathcal{L}_{\mathrm{percep}}\ . \tag{2}$$

The multi-stage weighting factor is set to 4^{-s}, i.e.

$$\mathcal{L}_{\mathrm{total}} = \sum_s 4^{-s} \mathcal{L}_{\mathrm{overall}}^s\ , \tag{3}$$

where s denotes the stage number.

D. All Quantitative Results

In the main paper, we only show part of the qualitative results. Here we offer all the qualitative results on single scene and multi-/cross-scene experiments, which are shown in Fig. 3 and Fig. 4.

Figure 1. **RGB-D images of all seen scenes in dataset.** Invalid depth values are shown in black areas.

Figure 2. **RGB-D images of all novel scenes in dataset.** Invalid depth values are shown in black areas.

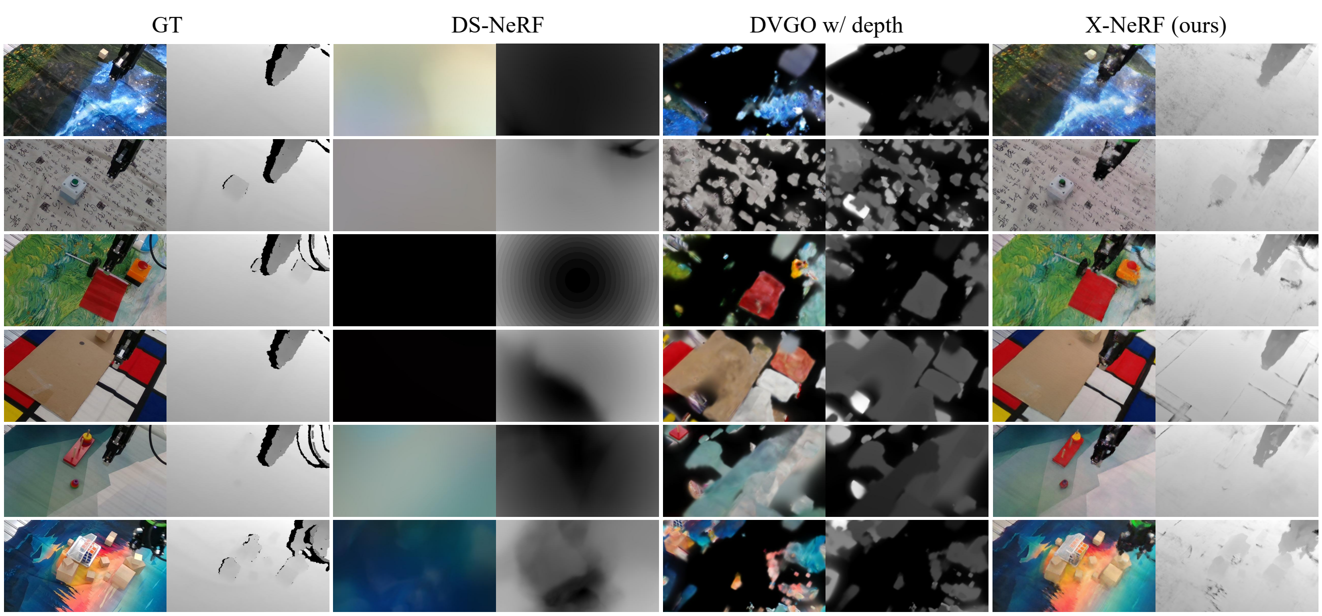

Figure 3. **All qualitative result on single scene.**

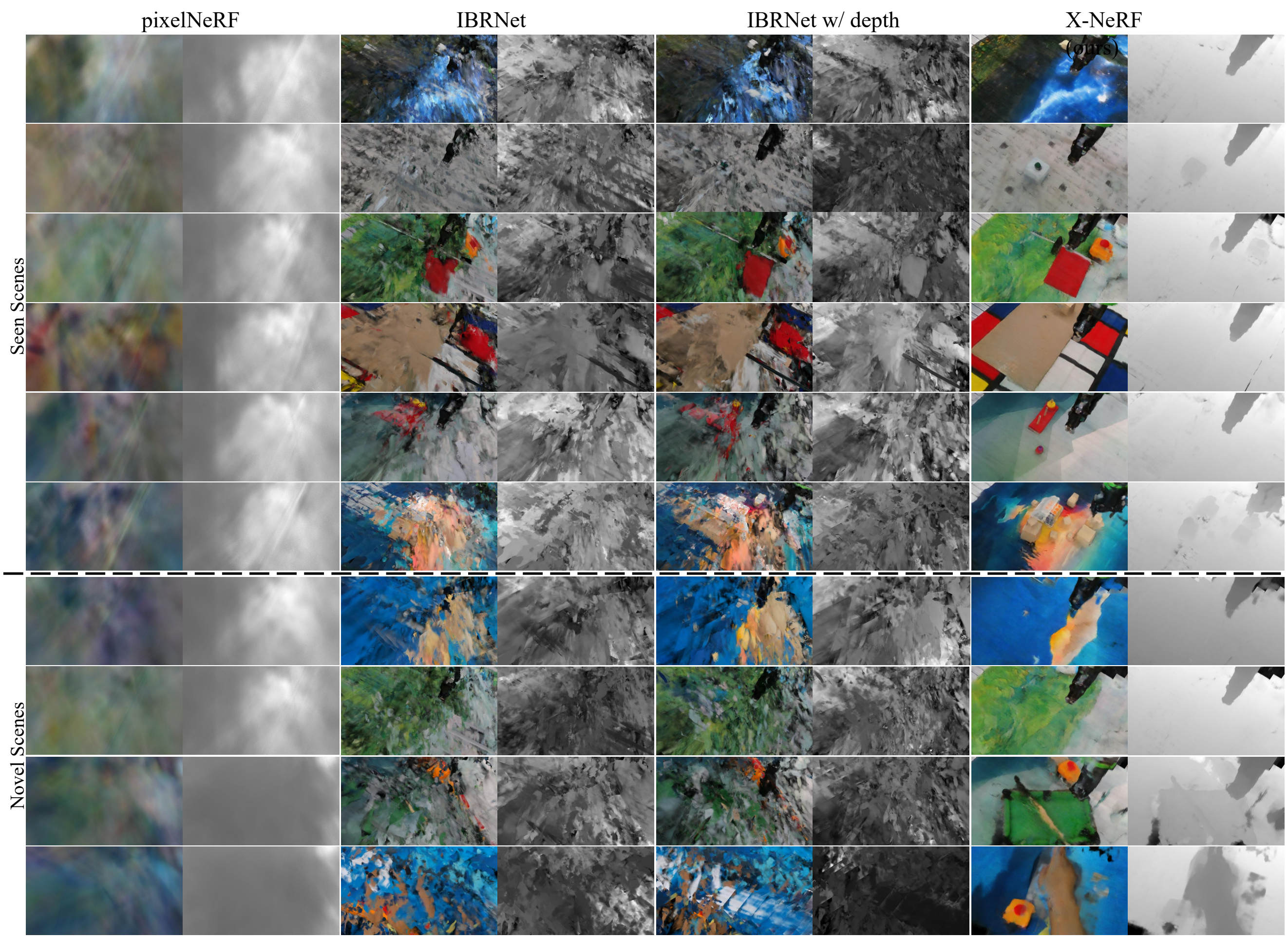

Figure 4. **All qualitative result on multi-scene and cross-scene.**