SCORP: Scene-Consistent Object Refinement via Proxy Generation and Tuning

Supplementary Material

In this supplementary material, we provide more implementation details in Sec. A, followed by additional quantitative and qualitative results in Sec. B. Finally, we provide more dataset details in Sec. C. Our code is available at https://github.com/PolySummit/SCORP.

A. Implementation Details

We conducted our experiments primarily using Tesla V100-PCIE-32G GPUs. Inference time for each scene is proportional to the number of images present. Specifically, for a scene comprising approximately 100 images with a common resolution (*e.g.*, 1080p), the complete refinement process of SCORP requires about 10 minutes. A detailed analysis of time cost and VRAM usage is provided in Sec. B.7.

A.1. Object Segmentation

For the object segmentation of a GS-based 3D scene, we initially identify an RGB image encompassing all relevant objects and employ Qwen [1] to generate initial object prompts. These prompts are subsequently refined by users through testing with Grounding DINO [16] to obtain precise object masks in the selected image. Then, SAM2 [18] is utilized to track and segment all objects throughout the entire RGB stream bidirectionally from the first frame, using the grounded masks as a reference.

A.2. Proxy Objects Synthesis

Starting from an initially selected view, we iteratively expand the input set by considering both the mask completeness of the segmented target object and view diversity to better support the generative model. As a preprocessing step, we discard the bottom 30% of images where the target object occupies the smallest proportion of the image pixels. Subsequent selections are made from the remaining candidates. The shape completeness score Q_{shape} for a given masked image is defined as:

$$Q_{\text{shape}} = \sum M / \sum \text{ConvexHull}(M > 0) \tag{1}$$

Here, $M > 0$ denotes the foreground region in the binary mask M, and $\text{ConvexHull}(M > 0)$ represents the convex hull of that region. A value of Q_{shape} close to 1 indicates that the object shape is compact and complete, while lower values suggest fragmentation or missing regions in the mask. To assess viewpoint diversity, we define a viewpoint diversity score Q_{view}, which considers both positional and orientational differences among viewpoints:

$$Q_{\text{view}} = 0.5 \cdot \tilde{d}_{\text{pos}} + 0.5 \cdot \tilde{d}_{\text{rot}}, \tag{2}$$

$$\tilde{d}_{\text{pos}} = \frac{d_{\text{pos}} - d_{\text{min}}}{d_{\text{max}} - d_{\text{min}} + \varepsilon}, \quad \tilde{d}_{\text{rot}} = d_{\text{rot}} \tag{3}$$

where d_{pos} denotes the minimum Euclidean distance from the current view to the selected views, d_{min} and d_{max} represent the global minimum and maximum of all such distances respectively, and ε is a small constant to prevent division by zero. Similarly, the term d_{rot} indicates the maximum dot product between the camera's viewing direction (*i.e.*, the unit vector along the z-axis) of the current view and those of the selected views. Since the dot product is in $[0, 1]$, d_{rot} is already normalized and requires no further scaling. The final image quality score is computed as a linear combination of the shape completeness and viewpoint diversity scores:

$$Q = \lambda_s Q_{\text{shape}} + \lambda_v Q_{\text{view}}. \tag{4}$$

where the parameters λ_s and λ_v are weighting factors that balance the contribution of mask completeness and viewpoint diversity. After selecting the complete set of input images, we perform center cropping around the target object and feed the cropped images into TRELLIS [23].

A.3. Coarse Alignment

According to Kerbl et al. [11], the covariance of a 3D Gaussian primitive can be represented as $\Sigma = RSS^\top R^\top$, where R is a rotation matrix, and S is a diagonal scaling matrix containing the scaling factors along the three principal axes. Obviously, R and S are the original parameters of the primitive. When applying a rotation R' to the primitive, the parameters, position μ and rotation R, of the primitive are transformed as $\mu \leftarrow R'\mu, R \leftarrow R'R$. Additionally, the SH is transformed according to https://github.com/graphdeco-inria/gaussian-splatting/issues/176 from the official Gaussian Splatting [11] repository. Especially, when a scale S' is applied to a 3D Gaussian primitive, the parameters are transformed as $\mu \leftarrow S'\mu, S \leftarrow SS'$.

Since the point cloud formed by the positions of all Gaussian primitives in the degraded object G^{deg} is incomplete and non-uniform, the Iterative Closest Point (ICP) [3] method used in the initial alignment has a high probability of converging to a local optimal solution or failing. To address this limitation, we adopt a simple heuristic method to find the transformation for initial alignment. First, we randomly generate 128^2 quaternions, each corresponding to a

potential rotation matrix. Then, we traverse and select 128 rotation matrices that form the largest angles with the already selected rotation matrices. These selected matrices serve as the initial rotations for the ICP process. Moreover, the initial translation is nearly **0** as a result of the alignment of the centroids between the proxy object and the degraded object. Next, we perform ICP optimization for each initial rotation matrix and finally select the transformation result with the best matching index as the transformation for initial alignment. We use the ICP implementation in Open3D [28], setting the maximum correspondence points-pair distance to 0.16 times the arithmetic mean on the dimension of the 3D bounding box of the degraded object G^{deg} mentioned, and maximum number of iterations to 400.

A.4. Matching-Based Pose Adjustment

As for the matching process of a single object, before inputting the masked target image I^{deg} and the rendered image I^{prx} into MASt3R [14] for object matching, we need to preprocess these two images. First, to reduce the subsequent computational load, we crop the images to focus on the area where the object mask or its projection is located. Next, we apply a padding of 200 pixels to the cropped images. This padding is intended to maintain a visually coherent appearance of the object in the image, which aids in the feature matching process of MASt3R.

For a pair of target image I^{deg} and rendered image I^{prx} in a dataset, we select the top 16 matching points based on the confidence. However, when the number of images $|\mathcal{I}|$ in a scene is large, we further reduce the time overhead caused by matching by employing an equidistant sampling strategy with a step size of $|\mathcal{I}|/15$. Due to the low quality of the original images from certain viewpoints or occlusions of the target object, the final number of point pairs projected back into 3D space using depth maps and camera parameters will be close to N_c (i.e., $\mathcal{P}^{\text{prx}}, \mathcal{P}^{\text{deg}} \in \mathbb{R}^{3\times N_c}$).

Notice that the problem addressed by Umeyama [20] is a least-squares problem, motivating the use of RANSAC [7] to improve robustness. In our implementation, we have developed our own version of RANSAC, setting the maximum number of iterations to 2000 and the inlier distance threshold to 0.01 times the arithmetic mean on the dimension of the 3D bounding box of the degraded object G^{deg}.

A.5. Scale-Undistorted Shape Refinement

As for a rotation matrix $R = \exp(\theta[u]_\times)$, $u = [v_1, v_2, v_3]^\top$ is the rotation axis and θ is the rotation angle, where $||u||^2 = (v_1^2 + v_2^2 + v_3^2) = 1$. $|||[u]_\times||_F^2 = \text{Tr}([u]_\times^\top [u]_\times)$ equals the sum of the squares of all entries of $[u]_\times$,

$$\begin{aligned} ||[u]_\times||^2 &= \sum_{i,j} ([u]_\times)_{i,j}^2 \\ &= 2(v_1^2 + v_2^2 + v_3^2) = 2 \end{aligned} \tag{5}$$

Therefore, $\mathcal{L}_R$ is the square of the rotation angle:

$$\begin{aligned} \mathcal{L}_R &= \tfrac{1}{2}||\log R||_F^2 = \tfrac{1}{2}||\theta[u]_\times||^2 \\ &= \theta^2 \cdot \tfrac{1}{2}|||[u]_\times||^2 = \theta^2, \end{aligned} \tag{6}$$

, limiting the angle of the rotation matrix. In the parameterization of the shape refinement problem, to make the scale parameter $S = \text{diag}(s_1, s_2, s_3)$ more controllable, we set minimum and maximum values for the scale, denoted as s_{min} and s_{max}, respectively. Specifically, for an original parameter $s_o \in \mathbb{R}$, we use the sigmoid function $f : x \mapsto \frac{1}{1+e^{-x}}$ to map it to the interval $(0, 1)$. Then, we apply a simple linear transformation $s = s_{\text{min}} + (s_{\text{max}} - s_{\text{min}})f(s_o)$ to map it to the interval $(s_{\text{min}}, s_{\text{max}})$. This operation is considered as a part of the regularization term $\mathcal{L}_S$ for S. We set $s_{\text{min}} = 0.75$, $s_{\text{max}} = 1.5$, $\lambda_S = 2 \times 10^{-5}$, and $\lambda_R = 1 \times 10^{-4}$ in our implementation. To solve the proposed optimization objective, Adam [13] optimizer is used with 3000 iterations, a learning rate of 10^{-3}, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. We utilize PyTorch for the optimizer and automatic differentiation.

A.6. Iterative Optimization

Note that the initial pose and scale alignment would be so rough that the results of 2D-2D correspondences and further 3D-3D correspondences may be inaccurate. Thus, the Correspondence Matching and Optimization are iteratively conducted to improve the matching and alignment results. Usually, there are 6 iterations overall, the first 3 iterations and the last are used for Pose Adjustment, and the remaining 2 iterations are used for Shape Refinement. Alg. 1 summarizes the Iterative Optimization procedure for Pose Adjustment and Shape Refinement of the proxy object.

A.7. Pose-Constrained Appearance Refinement

To ensure that the surface texture of G^{prx} remains consistent with the training views under visible viewpoints, we perform a fine-tuning step. In order to preserve the previously established alignment of pose and shape, we adjust only the spherical harmonic coefficients of G^{prx} without applying any adaptive density control, keeping the number and orientation of Gaussian points unchanged. The fine-tuning is conducted for a total of 600 iterations.

B. Additional Results

B.1. Quantitative Comparison Results

Tab. 1 presents the quantitative evaluation of our method alongside baseline approaches on the comprehensive dataset described in Sec. C.1, where all data represent averages computed over the three difficulty levels discussed in the paper. Our results demonstrate that our method outperforms all baselines, including 3DGS [11], 2DGS [8],

Algorithm 1 Gaussian Model Alignment

Ensure:
G^{deg}: GS-based original degraded object
G^{prx}: GS-based proxy object after coarse alignment
$\mathcal{T} = \{T_i\}_{i=1}^{N_c}$: Camera extrinsic set
K: Camera intrinsic, $\mathcal{I} = \{I_i\}_{i=1}^{N_c}$: Input images
$\mathcal{M} = \{M_{I_i}\}_{i=1}^{N_c}$: Binary masks for the target object
$T_{\max}$: Max iterations
$\mathscr{T}$: Set of Shape Refinement iteration serial numbers
Require: Aligned proxy object G^{prx}

for $t \leftarrow 1$ **to** $N_{\max}$ **do**
  $\mathcal{C} \leftarrow \varnothing$
  for $n \leftarrow 1$ **to** N_c **do**
    $I_n^{\text{prx}}, D_n^{\text{prx}} \leftarrow \text{Render}(T_n, G^{\text{prx}})$
    $\{(u_i^{\text{prx}}, u_i^{\text{deg}})\}_{i=1}^{N_m} \leftarrow \text{Match}(I_n^{\text{prx}}, M_{I_n} \odot I_n)$
    $\{(P_i^{\text{prx}}, P_i^{\text{deg}})\}_{i=1}^{N_m} \leftarrow \text{UnProject}(\{(u_i^{\text{prx}}, u_i^{\text{deg}})\}_{i=1}^{N_m}, D_n^{\text{prx}}, D_n^{\text{deg}}, T_n, K)$
    $\{P_i^{\text{prx}})\}_{i=1}^{N_m} \leftarrow \text{UnProject}(\{u_i^{\text{prx}}\}_{i=1}^{N_m}, D_n^{\text{prx}}, T_n, K)$
    $\{P_i^{\text{deg}})\}_{i=1}^{N_m} \leftarrow \text{UnProject}(\{u_i^{\text{deg}}\}_{i=1}^{N_m}, D_n^{\text{deg}}, T_n, K)$
    $\mathcal{C} \leftarrow \mathcal{C} \cup \{(P_i^{\text{prx}}, P_i^{\text{deg}})\}_{i=1}^{N_m}$
  end for
  if $t \notin \mathscr{T}$ **then**
    $(\hat{R}, \hat{t}, \hat{S}) \leftarrow \underset{R,t,s}{\text{argmin}} \sum_{(P^{\text{prx}}, P^{\text{deg}}) \in \mathcal{C}} \left\| sRP^{\text{prx}} + t - P^{\text{deg}} \right\|^2$
    $\hat{R}' \leftarrow I$
  else
    $(\hat{R}, \hat{t}, \hat{S}, \hat{R}') \leftarrow \underset{R,t,S,R'}{\text{argmin}} \Big\{ \sum_{(P^{\text{prx}}, P^{\text{deg}}) \in \mathcal{C}} \left\| RR'^{\top} S R' P^{\text{prx}} + t - P^{\text{deg}} \right\|^2 + \lambda_R \mathcal{L}_R + \lambda_S \mathcal{L}_S \Big\}$
  end if
  $G^{\text{prx}} \leftarrow \text{Rotate}(G^{\text{prx}}, \hat{R}')$
  $G^{\text{prx}} \leftarrow \text{Scale}(G^{\text{prx}}, \hat{S})$
  $G^{\text{prx}} \leftarrow \text{Rotate}(G^{\text{prx}}, \hat{R}'^{\top})$
  $G^{\text{prx}} \leftarrow \text{Rotate}(G^{\text{prx}}, \hat{R})$
  $G^{\text{prx}} \leftarrow \text{Translate}(G^{\text{prx}}, \hat{t})$
end for
return G^{prx}

DNGaussian [15], and GenFusion [21], across nearly all metrics.

B.2. Qualitative Comparison Results

Fig. 1 illustrates additional qualitative results obtained from various methods applied to the same scene under medium difficulty (*i.e.*, with the removal of 4/5 of the images). Although our method does not guarantee pixel-level correspondence with the ground truth, it consistently maintains a high level of visual quality and overall coherence. Figs. 2, 3, 4, and 5 provide further qualitative comparisons between our approach and other baseline methods, demonstrating that our approach offers significant advantages in terms of geometric quality.

Table 1. Additional quantitative comparisons of rendering quality were obtained using SCORP alongside several baseline methods. The best, the second best, and the third best are highlighted. Due to table linewidth constraints, partial method names and scene names have been abbreviated (*e.g.*, "DN." denotes DNGassian [15] and "show_r." represents the scene *show_rack*).

	SSIM↑					PSNR↑					LPIPS↓				
	3DGS	2DGS	DN.	Gen.	Ours	3DGS	2DGS	DN.	Gen.	Ours	3DGS	2DGS	DN.	Gen.	Ours
bonsai	0.951	0.950	0.949	0.938	0.927	27.169	26.831	26.848	25.650	19.987	0.046	0.045	0.049	0.067	0.065
garden	0.819	0.859	0.846	0.758	0.842	19.053	20.989	21.193	18.115	17.095	0.168	0.139	0.163	0.238	0.157
kitchen	0.883	0.907	0.890	0.847	0.888	24.800	24.669	23.737	21.901	20.026	0.100	0.091	0.104	0.160	0.109
scene1	0.751	0.774	0.852	0.738	0.868	9.648	10.507	14.389	8.617	16.259	0.237	0.219	0.140	0.251	0.136
scene2	0.805	0.835	0.784	0.682	0.861	18.550	19.897	18.012	14.801	16.662	0.215	0.186	0.229	0.327	0.164
Bench	0.905	0.955	0.955	0.944	0.969	18.316	23.227	23.311	22.264	24.776	0.099	0.052	0.056	0.071	0.029
Desk	0.867	0.885	0.877	0.844	0.885	17.498	18.762	17.641	16.350	17.225	0.165	0.142	0.155	0.188	0.131
donuts	0.907	0.936	0.900	0.861	0.973	17.332	19.430	16.281	15.149	26.443	0.112	0.078	0.114	0.172	0.030
figurines	0.933	0.940	0.939	0.929	0.957	20.466	21.395	20.965	19.519	22.640	0.079	0.069	0.072	0.088	0.049
shoe_r.	0.940	0.956	0.948	0.943	0.962	22.556	25.248	23.951	22.870	24.430	0.068	0.049	0.059	0.070	0.040
teatime	0.928	0.949	0.950	0.926	0.968	17.828	20.313	20.014	18.127	25.539	0.084	0.060	0.061	0.087	0.043
Avg.	0.881	0.904	0.899	0.855	0.918	19.383	21.024	20.577	18.488	21.007	0.125	0.103	0.109	0.156	0.087

	MUSIQ↑					CLIPS↑					mIoU↑				
	3DGS	2DGS	DN.	Gen.	Ours	3DGS	2DGS	DN.	Gen.	Ours	3DGS	2DGS	DN.	Gen.	Ours
bonsai	48.711	51.993	44.882	20.048	49.205	0.921	0.920	0.905	0.875	0.882	0.769	0.798	0.740	0.652	0.394
garden	55.415	61.179	54.888	42.411	63.410	0.845	0.896	0.845	0.797	0.892	0.611	0.702	0.667	0.524	0.731
kitchen	48.983	51.518	46.280	33.304	54.907	0.920	0.922	0.917	0.833	0.947	0.553	0.743	0.615	0.550	0.838
scene1	55.485	57.513	33.900	50.191	59.004	0.765	0.769	0.623	0.754	0.863	0.481	0.527	0.001	0.458	0.636
scene2	42.301	48.475	39.973	33.004	61.706	0.799	0.825	0.773	0.735	0.887	0.594	0.693	0.584	0.470	0.800
Bench	38.889	44.997	41.130	30.470	58.641	0.866	0.909	0.890	0.868	0.956	0.405	0.722	0.697	0.625	0.819
Desk	47.242	52.417	47.392	41.457	65.700	0.821	0.833	0.835	0.795	0.873	0.703	0.765	0.744	0.654	0.724
donuts	35.328	40.032	34.682	25.904	53.335	0.829	0.861	0.836	0.824	0.922	0.264	0.395	0.259	0.184	0.648
figurines	34.804	40.740	35.066	20.399	48.797	0.871	0.880	0.878	0.848	0.899	0.465	0.510	0.495	0.409	0.448
shoe_r.	33.428	40.262	34.970	23.596	50.221	0.879	0.908	0.890	0.879	0.901	0.399	0.444	0.472	0.322	0.408
teatime	27.961	35.051	30.117	20.654	50.358	0.823	0.837	0.841	0.827	0.893	0.290	0.408	0.434	0.255	0.426
Avg.	42.595	47.653	40.298	31.040	55.935	0.849	0.869	0.839	0.821	0.901	0.503	0.610	0.519	0.464	0.625

Table 2. More Quantitative results of the ablation study on the rendering quality. Metrics on appearance are computed over three scenes (3DGS-CD-*Bench*, 3DGS-CD-*Desk*, LERF-*Donuts*). best, second best, third best and false case are highlighted. Since the Appearance Refinement branch only optimizes color, the mIoU of (a) and (g) are the same and therefore omitted.

SSIM↑	Bench				Desk				Donuts				Avg.			
	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.
a	0.961	0.962	0.963	0.962	0.889	0.881	0.858	0.876	0.974	0.968	0.972	0.971	0.942	0.937	0.931	0.936
b	0.963	0.968	0.970	0.967	0.000	0.848	0.000	0.283	0.973	0.000	0.000	0.324	0.645	0.605	0.323	0.525
c	0.964	0.964	0.963	0.964	0.886	0.884	0.885	0.885	0.974	0.969	0.971	0.971	0.941	0.939	0.940	0.940
d	0.920	0.700	0.309	0.643	0.770	0.774	0.622	0.722	0.951	0.935	0.915	0.934	0.881	0.803	0.615	0.766
e	0.908	0.698	0.309	0.638	0.772	0.775	0.634	0.727	0.950	0.935	0.913	0.933	0.877	0.803	0.619	0.766
f	0.957	0.963	0.964	0.961	0.883	0.875	0.867	0.875	0.972	0.964	0.972	0.969	0.937	0.934	0.934	0.935
g	0.970	0.968	0.969	0.969	0.895	0.881	0.880	0.885	0.975	0.971	0.973	0.973	0.947	0.940	0.941	0.943

PSNR↑	Bench				Desk				Donuts				Avg.			
	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.
a	22.568	22.967	23.072	22.869	17.133	16.132	14.724	15.997	26.235	24.976	25.197	25.469	21.979	21.358	20.998	21.445
b	23.484	24.130	24.639	24.084	0.000	13.549	0.000	4.516	26.356	0.000	0.000	8.785	16.613	12.560	8.213	12.462
c	23.316	22.540	22.349	22.735	16.794	16.778	17.540	17.038	26.479	25.268	25.612	25.786	22.196	21.529	21.833	21.853
d	21.185	16.310	13.420	16.972	14.498	14.744	9.440	12.894	23.471	21.486	17.223	20.726	19.718	17.513	13.361	16.864
e	18.954	15.935	13.437	16.109	14.079	14.349	9.410	12.613	22.853	21.066	16.893	20.271	18.629	17.117	13.247	16.331
f	20.222	21.747	22.281	21.416	16.676	16.518	15.461	16.218	26.179	24.336	26.039	25.518	21.026	20.867	21.260	21.051
g	25.122	24.306	24.901	24.776	17.635	17.106	16.935	17.225	27.032	25.681	26.615	26.443	23.263	22.364	22.817	22.815

LPIPS↓	Bench				Desk				Donuts				Avg.			
	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.
a	0.035	0.035	0.034	0.035	0.133	0.134	0.154	0.141	0.028	0.038	0.032	0.032	0.065	0.069	0.074	0.069
b	0.039	0.031	0.029	0.033	1.000	0.154	1.000	0.718	0.030	1.000	1.000	0.677	0.356	0.395	0.676	0.476
c	0.036	0.035	0.035	0.036	0.130	0.130	0.134	0.131	0.029	0.034	0.032	0.032	0.065	0.067	0.067	0.066
d	0.076	0.322	0.760	0.386	0.424	0.294	0.635	0.451	0.055	0.078	0.104	0.079	0.185	0.231	0.500	0.305
e	0.085	0.330	0.756	0.390	0.435	0.295	0.635	0.455	0.056	0.080	0.105	0.080	0.192	0.235	0.499	0.309
f	0.042	0.037	0.035	0.038	0.140	0.145	0.145	0.144	0.031	0.040	0.031	0.034	0.071	0.074	0.070	0.072
g	0.029	0.030	0.030	0.030	0.121	0.134	0.134	0.130	0.026	0.032	0.032	0.030	0.059	0.066	0.066	0.063

mIoU↑	Bench				Desk				Donuts				Avg.			
	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.
a	\	\	\	\	\	\	\	\	\	\	\	\	\	\	\	\
b	0.773	0.747	0.804	0.775	0.000	0.299	0.000	0.100	0.636	0.000	0.000	0.212	0.470	0.349	0.268	0.362
c	0.803	0.681	0.674	0.719	0.675	0.663	0.735	0.691	0.656	0.508	0.582	0.582	0.711	0.617	0.664	0.664
d	0.432	0.128	0.048	0.203	0.343	0.463	0.227	0.345	0.473	0.335	0.217	0.342	0.416	0.309	0.164	0.296
e	0.432	0.128	0.048	0.203	0.343	0.463	0.227	0.345	0.473	0.335	0.217	0.342	0.416	0.309	0.164	0.296
f	0.494	0.662	0.718	0.625	0.747	0.630	0.599	0.659	0.615	0.485	0.608	0.569	0.619	0.592	0.642	0.618
g	0.862	0.781	0.813	0.819	0.754	0.698	0.720	0.724	0.712	0.606	0.627	0.648	0.776	0.695	0.720	0.730

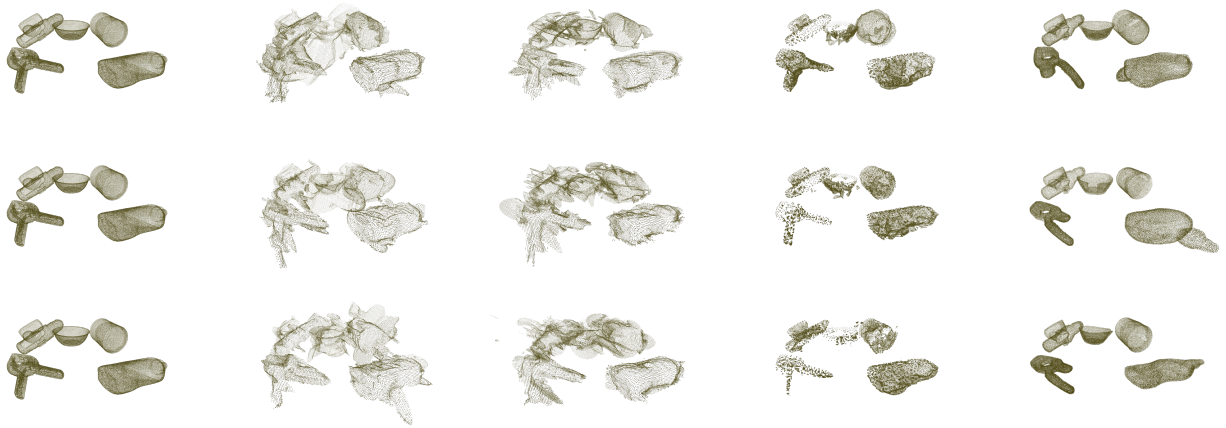
Figure 1. Qualitative results of rendered images of target objects produced by SCORP and several baseline methods. From top to bottom, images are selected from Mip360-*garden* `DSC08018 DSC08038 DSC08130` with "soccer ball" and "pinecones vase", Mip360-*kitchen* `DSCF0740 DSCF0759 DSCF0860` with "lego bulldozer", ToyDesk-*scene2* `0052 0056 0145` with "onigir", "dinosaur", "grass block", "rubik" and "pink candy", and 3DGS-CD-*Desk* `frame_00020 frame_00090 frame_00162` with "yellow bottle", "red bag" and "light green bowl".

GT DNGaussian [15] Genfusion [21] ComPC [9] Ours

Figure 2. Qualitative results of point cloud data for Scene 1 in our synthetic dataset. From top to bottom: easy, medium, and hard settings. All point clouds are downsampled to 16384 points to keep similar visual effect.

GT DNGaussian [15] Genfusion [21] ComPC [9] Ours

Figure 3. Qualitative results of point cloud data for Scene 2 in our synthetic dataset.



GT DNGaussian [15] Genfusion [21] ComPC [9] Ours

Figure 4. Qualitative results of point cloud data for Scene 3 in our synthetic dataset.

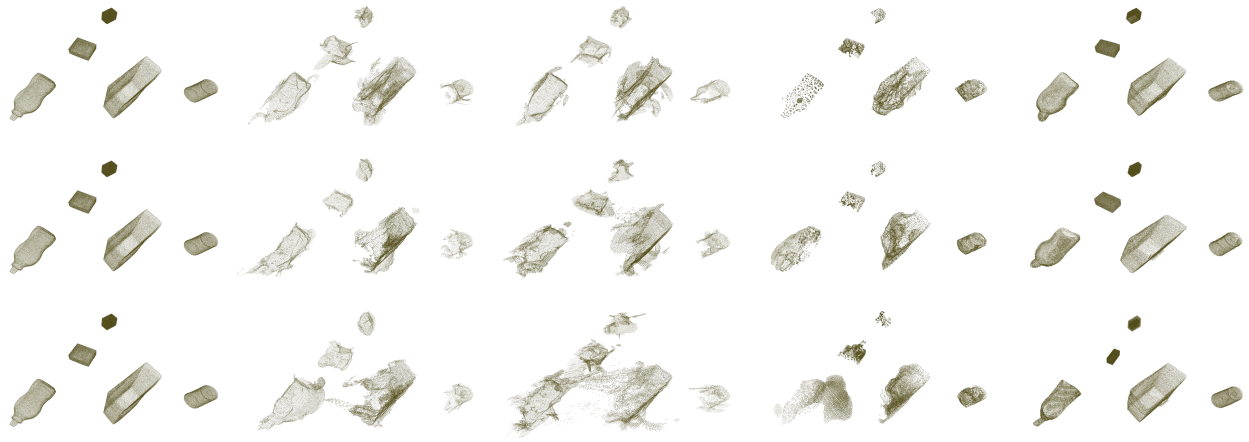


Figure 5. Qualitative results of point cloud data for Scene 4 in our synthetic dataset.

B.3. Quantitative Ablation Results

Tab. 2 and Tab. 3 report the outcomes of the ablation experiments described in the main paper, evaluated across different scenes using appearance and geometry metrics, respectively. Besides, the failure cases associated with the method proposed by Chatrasingh et al. [4] are explicitly presented.

Table 3. More Quantitative results of the ablation study on the geometry quality. Metrics on appearance are computed over 3 randomly selected scenes from the constructed scenarios. Performance is evaluated using Chamfer Distance (CD) [cm] and Earth Mover Distance (EMD) [cm]. Results under the worst-performing ablation condition of the same scenario and difficulty level are included in the average calculation when false cases are present. The value 5.000 is chosen for the false case since it is the upper bound value among the remaining values. Ablation (a), (e) are not included in the table since the remaining ones are sufficient to prove the effectiveness of the modules of SCORP.

CD↓	Scene 1				Scene 2				Scene 3				Avg.			
	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.
b	5.000	1.201	4.511	3.571	1.965	1.954	2.547	2.155	1.302	1.362	1.881	1.515	1.733	1.506	2.980	2.073
c	1.869	1.999	1.174	1.681	1.597	1.720	2.305	1.874	1.318	1.863	1.425	1.535	1.594	1.860	1.634	1.696
d	1.819	1.284	1.903	1.669	1.576	1.956	2.543	2.025	1.422	1.362	1.881	1.555	1.605	1.534	2.109	1.749
f	1.558	2.940	4.520	3.006	1.186	1.234	3.306	1.909	1.250	3.304	2.778	2.444	1.331	2.492	3.534	2.452
g	1.932	1.802	1.684	1.806	1.192	1.162	1.684	1.346	1.176	1.269	1.355	1.267	1.433	1.411	1.574	1.472

EMD↓	Scene 1				Scene 2				Scene 3				Avg.			
	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.	2/3	4/5	6/7	Avg.
b	5.000	1.406	4.442	3.616	2.223	2.581	3.224	2.676	1.337	1.825	2.515	1.892	2.052	1.937	3.393	2.461
c	2.595	1.932	1.717	2.081	2.070	1.954	3.271	2.432	1.296	2.033	1.623	1.651	1.987	1.973	2.203	2.054
d	2.474	1.485	2.057	2.005	1.878	2.585	3.219	2.561	1.664	1.822	2.509	1.998	2.005	1.964	2.595	2.188
f	1.791	2.933	4.462	3.062	1.614	1.599	3.812	2.342	1.307	3.518	3.104	2.643	1.570	2.683	3.792	2.682
g	1.840	1.902	2.021	1.921	1.880	1.729	1.803	1.804	1.299	1.498	1.671	1.489	1.673	1.709	1.831	1.738

B.4. Qualitative Ablation Results

We conducted additional ablation experiments to evaluate the effectiveness of various components in our Shape Refinement solution. These components include (i) initial alignment using only ICP, (ii) w/o the rotation term R', (iii) w/o the regularization terms $\mathcal{L}_R$ and $\mathcal{L}_S$, (iv) w/o R', $\mathcal{L}_R$, and $\mathcal{L}_S$, and (v) the full model. Fig. 6 presents the qualitative results of these ablation experiments. Configuration (i) demonstrates that the proxy object maintains its shape following generation, whereas (ii) and (iii) result in the shape distortion after alignment. In (iv), additional deformation and unrealistic object dimensions are observed. These results indicate that the full model consistently produces correctly proportioned objects and achieves distortion-free alignment.

B.5. Performance on Visible Viewpoints

We conducted an additional experiment that evaluates the PSNR, SSIM, and LPIPS specifically on visible views, comparing SCORP against 3DGS [11]. As shown in Tab. 4, SCORP achieves comparable performance in visible views due to the effective Appearance Refinement (AR) module. As some cases in the dataset contain complex geometry, such as the "lego bulldozer" with so complex detailed structures including "buckets", "tracks", "connecting rod structures", and "cabs" in Mip360-*kitchen* scene that TRELLIS [23] could not fully capture, SCORP may not always achieve the higher PSNR, resulting in a lower average PSNR. Moreover, mathematically, the same ΔPSNR could be hard to perceive when PSNR is high enough. Since SCORP achieves a similar performance on appearance to the baseline in regions that are actually observed during training, we believe that our method still faithfully reconstructs the geometry and appearance of known regions in most cases.

B.6. Robot Grasping

We further validate the application of our method in the robotic grasping task. In real-world scenarios, missing observations from specific viewpoints often lead to degraded reconstruction quality and reduced grasping performance under those views. As shown in Fig. 7, compared to the

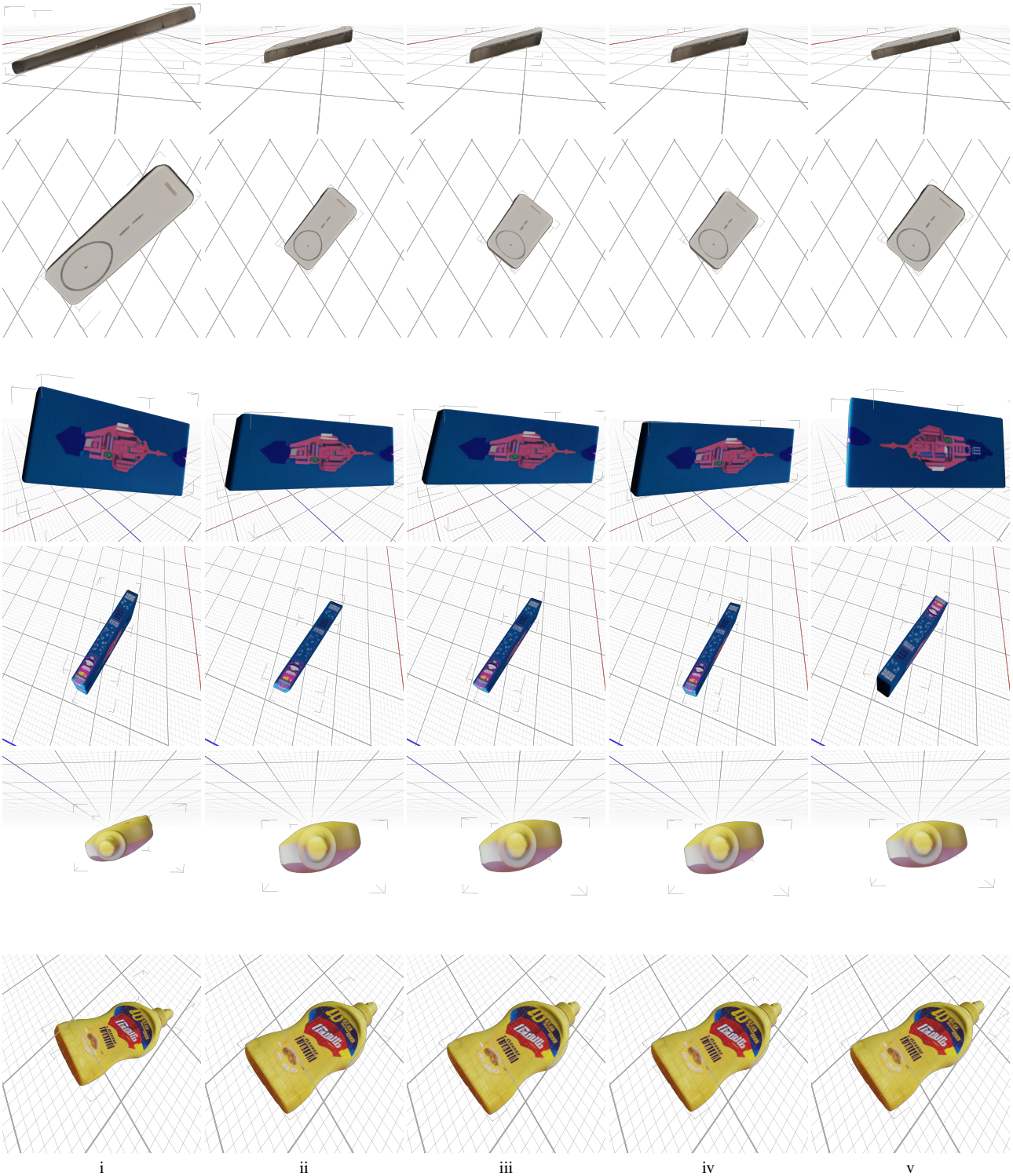


Figure 6. Qualitative ablations on Scale-Undistorted Shape Refinement for a real-world object “wireless charger”, “blue box” from 3DGS-CD-*Bench*, and “yellow bottle” from 3DGS-CD-*Desk*.

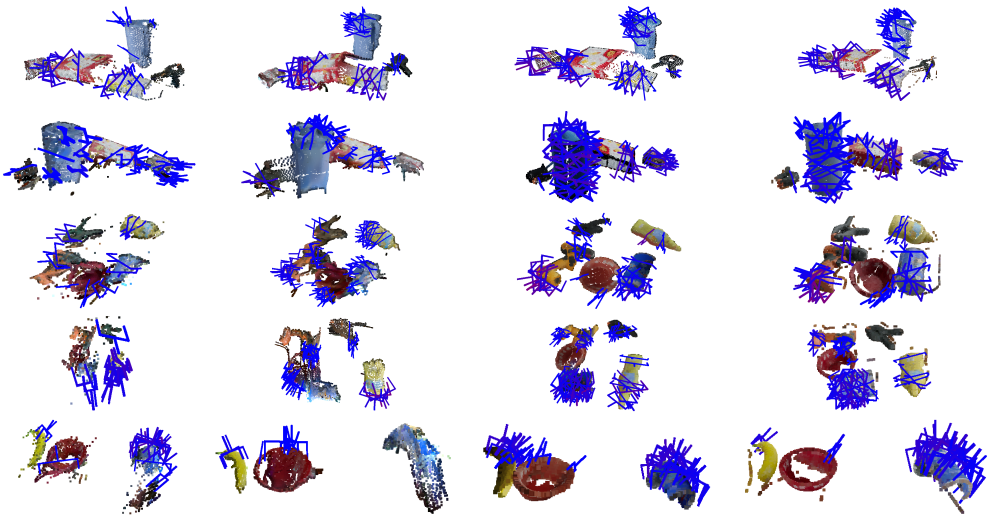


Figure 7. More AnyGrasp [6] results on the rendering depth of test view.

Table 4. Quantitative comparison of appearance on visible views against 3DGS [11] among different settings. Our method achieves comparable performance to 3DGS on seen views

		1/3	1/5	1/7	Avg.
PSNR↑	3DGS	33.938	33.907	34.834	34.226
	Ours	33.114	33.401	33.630	33.381
SSIM↑	3DGS	0.9877	0.9883	0.9896	0.9885
	Ours	0.9833	0.9855	0.9855	0.9848
LPIPS↓	3DGS	0.0197	0.0185	0.0170	0.0184
	Ours	0.0221	0.0203	0.0187	0.0203

fragmented objects from baseline, rendered RGB-D of objects refined by SCORP enable the grasping model [6] to generate denser and more plausible grasp candidates. Results highlight our method's sim-to-real potential application with inferred high-quality grasps on limited views.

Table 5. Time cost and peak VRAM usage statistics on Mip360-*kitchen* among different difficulty settings, which are shown in "Time [min]↓ / VRAM [MB]↓" columns respectively. Since CA uses none of the VRAM, and PA obviously uses less VRAM than SR, we do not report the VRAM usage of CA and PA.

	3DGS Init.	3D Seg.	OS	CA	PA	SR	AR
1/3	8.55 / 1344.2	0.58 / 827.7	1.62 / 11048.2	1.65 / -	3.47 / -	2.93 / 6061.3	1.57 / 2120.7
1/5	7.40 / 1310.3	0.37 / 823.5	1.60 / 11203.6	1.73 / -	3.58 / -	2.88 / 5726.9	1.55 / 2149.7
1/7	7.22 / 1302.5	0.20 / 813.0	1.08 / 11098.4	1.77 / -	3.83 / -	2.47 / 5556.8	1.25 / 2130.0

B.7. Efficiency Analysis

We report wall-clock time and peak GPU memory for each stage of SCORP on a single Tesla V100 GPU with 32 GB on the high-resolution (3115x2078) kitchen scene with a single target object. The results are summarized in Tab. 5, which provides a breakdown of VRAM usage and time cost for each component, including 3DGS Scene Initialization (Scene Init.), 3D Segmentation (3D Seg.), Object Synthesis (OS), Coarse Alignment (CA), Pose Adjustment (PA), Scale Refinement (SR), and Appearance Refinement (AR). There are 4 iterations in PA and 2 iterations in SR, which could be adjusted as needed on custom scenes. The results indicate that the peak VRAM consumption in our pipeline consistently arises during the OS stage.

Additionally, we compare both the computation time and the memory usage between our method and a baseline, GenFusion [21], in Tab. 6. This comparison highlights the practical efficiency of our approach and provides a more comprehensive understanding of its performance characteristics. During our experiments, we tried to maintain the default settings of the baseline method and fixed the number of Gaussian training iterations to 10k across methods to ensure a fair comparison. It is worth noting that GenFusion resizes all images used in the diffusion model to 960x512, which is consistent with its default settings. Besides, we noticed that our method's time cost on 3DGS-CD-*Bench* is slightly higher than Mip360-*kitchen* since there are 3 target objects in the *Bench* scene, which triple (3x) the time cost of CA, PA, and SR in our method compared to the single object in the same scene. In the high-resolution *kitchen* scene, our method achieves significantly lower VRAM usage and competitive computation time cost compared to GenFusion.

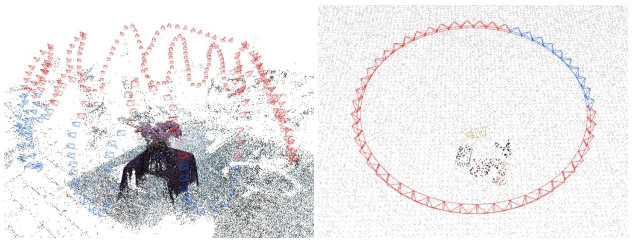
Table 6. Quantitative comparison of time cost and peak VRAM usage between our method and GenFusion [21], on 3DGS-CD-*Bench* (1006x753) and Mip360-*kitchen* (3115x2078). Consistently, our method achieves a lower VRAM usage and a better computation time cost. The better score is highlighted in **bold**.

		Bench-1/3	*Bench*-1/5	*Bench*-1/7	*kitchen*-1/3	*kitchen*-1/5	*kitchen*-1/7
#images	\	52	32	26	93	59	45
Time [min]↓	Gen.	51.98	49.45	48.70	82.68	78.18	75.70
	Ours	**22.45**	**20.67**	**22.27**	**20.45**	**19.20**	**18.03**
VRAM [GB]↓	Gen.	20.70	20.60	20.55	20.58	20.56	20.56
	Ours	**9.76**	**9.59**	**9.79**	**10.79**	**10.94**	**10.84**

C. Dataset Details

C.1. Appearance

To evaluate the appearance quality of SCORP, we selected scenes from the NVS task datasets, Mip360 [2] and LERF [12]. Due to the specific requirements of the task, the chosen scenes were required to include images captured around significant objects to serve as ground truth, which posed challenges in dataset selection. Thus, we also extracted some scenes from other task datasets, ToyDesk [25] and 3DGS-CD [17]. Ultimately, we obtained 11 scenes with images captured around objects, providing us with sufficient ground truth. Specifically, we selected the *bonsai*, *garden*, and *kitchen* scenes from Mip360 [2], the *donuts*, *figurines*, *show_rack*, and *teatime* scenes from LERF [12], the *scene1* and *scene2* scenes from ToyDesk [25], and the *Bench* and *Desk* scenes from 3DGS-CD [17]. For evaluation, we randomly select a start view and remove the n nearest views in joint position–orientation space as unseen views (*i.e.*, test views). Fig. 8 illustrates the resulting view splits on a real scenario and on a constructed scenario.



kitchen *scene 3*

Figure 8. Visualization of the split of seen views and unseen views for evaluation, where the views with red cameras are unseen views and the views with blue cameras are seen views. The point cloud is for 3DGS training initialization. The left is from Mip360-*bonsai* and the right is from *scene 3* of constructed scenarios.

C.2. Geometry

To facilitate geometric evaluation in physically plausible and visually realistic environments, we construct a synthetic dataset tailored to multi-object 3D reconstruction tasks. Preliminary investigations reveal that existing datasets widely used in appearance-focused tasks, such as Mip360 and LERF, are unsuitable due to the lack of precise ground truth geometry. Likewise, datasets that focus solely on single isolated objects, such as DTU [10], BlenderMVS [26], and OmniObject [22], fail to capture the complex inter-object occlusions and collisions typical in real-world scenes. Although scene-level RGB-D datasets such as SUN RGB-D [27], and Replica [19] offer realistic sensor data, they lack accurate and complete 3D ground truth and are thus excluded. Similarly, point cloud completion datasets like Redwood [5] are omitted due to the absence of accompanying RGB imagery, which is essential for our reconstruction objectives.

Due to these limitations, we build upon the YCB-Video [24] dataset, which provides high-quality object meshes, to generate a set of synthetic scenes with controlled geometric and visual properties. For each scene, we first generate a horizontal base plane equipped with a collision volume to serve as the physical support. Multiple object meshes are then randomly instantiated within a bounded volume above the plane and assigned individual collision bodies. A rigid-body physics simulation is performed, allowing the objects to fall under gravity and interact naturally until a stable configuration is reached. After convergence, we simulate a camera moving along a circular trajectory around the scene center. From discrete camera poses, we render corresponding RGB images and depth maps, associated with camera extrinsics, along with the complete ground truth scene mesh. This procedure ensures consistent, high-fidelity data suitable for quantitative geometric analysis under realistic multi-object occlusion and physical interaction.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 1

[2] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5470–5479, 2022. 9

[3] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures*, pages 586–606. Spie, 1992. 1

[4] Maria Chatrasingh, Cholatip Wiratkapun, and Jackrit Suthakorn. A generalized closed-form solution for 3d registration of two-point sets under isotropic and anisotropic scaling. *Results in Physics*, 51:106746, 2023. 6

[5] Sungjoon Choi, Qian-Yi Zhou, Stephen Miller, and Vladlen Koltun. A large dataset of object scans. *arXiv:1602.02481*, 2016. 9

[6] Hao-Shu Fang, Chenxi Wang, Hongjie Fang, Minghao Gou, Jirong Liu, Hengxu Yan, Wenhai Liu, Yichen Xie, and Cewu Lu. Anygrasp: Robust and efficient grasp perception in spatial and temporal domains. *IEEE Transactions on Robotics*, 39(5):3929–3945, 2023. 8

[7] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 2

[8] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 conference papers*, pages 1–11, 2024. 2, 4

[9] Tianxin Huang, Zhiwen Yan, Yuyang Zhao, and Gim Hee Lee. Compc: Completing a 3d point cloud with 2d diffusion priors. In *The Thirteenth International Conference on Learning Representations*, 2024. 5, 6

[10] 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. 9

[11] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 1, 2, 4, 6, 8

[12] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. Lerf: Language embedded radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 19729–19739, 2023. 9

[13] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *Computer Science*, 2014. 2

[14] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European Conference on Computer Vision*, pages 71–91. Springer, 2024. 2

[15] Jiahe Li, Jiawei Zhang, Xiao Bai, Jin Zheng, Xin Ning, Jun Zhou, and Lin Gu. Dngaussian: Optimizing sparse-view 3d gaussian radiance fields with global-local depth normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20775–20785, 2024. 3, 4, 5, 6

[16] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *Computer Vision – ECCV 2024*, pages 38–55, Cham, 2025. Springer Nature Switzerland. 1

[17] Ziqi Lu, Jianbo Ye, and John Leonard. 3dgs-cd: 3d gaussian splatting-based change detection for physical object rearrangement. *IEEE Robotics and Automation Letters*, 2025. 9

[18] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollar, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. In *The Thirteenth International Conference on Learning Representations*, 2025. 1

[19] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 9

[20] Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 13(04): 376–380, 1991. 2

[21] Sibo Wu, Congrong Xu, Binbin Huang, Andreas Geiger, and Anpei Chen. Genfusion: Closing the loop between reconstruction and generation via videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6078–6088, 2025. 3, 4, 5, 6, 8

[22] Tong Wu, Jiarui Zhang, Xiao Fu, Yuxin Wang, Jiawei Ren, Liang Pan, Wayne Wu, Lei Yang, Jiaqi Wang, Chen Qian, et al. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 803–814, 2023. 9

[23] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21469–21480, 2025. 1, 6

[24] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. In *Robotics: Science and Systems (RSS)*, 2018. 9

[25] Bangbang Yang, Yinda Zhang, Yinghao Xu, Yijin Li, Han Zhou, Hujun Bao, Guofeng Zhang, and Zhaopeng Cui. Learning object-compositional neural radiance field for editable scene rendering. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13779–13788, 2021. 9

[26] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1790–1799, 2020. 9

[27] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. *Advances in neural information processing systems*, 27, 2014. 9

[28] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3d: A modern library for 3d data processing. *arXiv preprint arXiv:1801.09847*, 2018. 2