ObjectSDF++: Improved Object-Compositional Neural Implicit Surfaces

Supplementary Material

Qianyi Wu[1] Kaisiyuan Wang[2] Kejie Li[3] Jianmin Zheng[4] Jianfei Cai[1]
[1]Monash University [2]University of Sydney [3]University of Oxford [4]Nanyang Technological University
{qianyi.wu, jianfei.cai}@monash.edu kaisiyuan.wang@sydney.edu.au
kejie.li@outlook.com asjmzheng@ntu.edu.sg

In this supplementary material, we will discuss the occlusion-aware object opacity design in Sec. A, more implementation details in Sec. B, the definition of evaluation metric in Sec. C, more experimental results in Sec. D and discussion about our framework E.

A. Discussion about occlusion-aware object opacity

We explore the distinctions between our design and other existing methods, providing additional details to enhance the readers' comprehension. Scene opacity is defined as $O_\Omega(v) = 1 - T_\Omega(v)$, the probability that a ray does hit a particle before reaching v, with rigorously derived PDF $\frac{dO_\Omega(v)}{dv}(v) = T_\Omega(v)\sigma_\Omega(\mathbf{r}(v))$ [15, 12]. 1) When extending to object opacity, a simple way is $O_{\mathcal{O}_i}(\mathbf{r}) = \int_{v_n}^{v_f} T_{\mathcal{O}_i}(v)\sigma_{\mathcal{O}_i}(\mathbf{r}(v))dv$, $(E1)$, which however ignores the occlusions among objects [14, 6]. To model the occlusion between objects, vMAP [6] requires ground truth depth to determine the integration range in $(E1)$ and requires a manual interval shrinkage design in volume rendering [9]. This is also similar to the 3D mask design in ObjectNeRF [14] 2) ObjectSDF [13] builds an additional semantic field $\mathbf{s}$ and render via $\int_{v_n}^{v_f} T_\Omega(v)\sigma_\Omega(\mathbf{r}(v))\mathbf{s}(\mathbf{r}(v))dv$ to handle occlusion. It introduces an extra tuning hyperparameter for semantic mapping. 3) In contrast, we rethink the occlusion issue and introduce this design: $O_{\mathcal{O}_i}(\mathbf{r}) = \int_{v_n}^{v_f} T_\Omega(v)\sigma_{\mathcal{O}_i}(\mathbf{r}(v))dv$, to approximate both visible and occluded object opacity. This design removes the dependency on ground truth depth data or additional semantic mapping and has shown effectiveness in experiments, which also offers fundamental insights.

B. More Implementation Details

Multi-Resolution Feature Grid Provoked by [8], we adopt the multi-resolution feature grid to compensate the fixed frequency position encoding used in vanilla NeRF [7] to accelerate the model convergence speed. Concretely, the 3D space will be represented by a $L = 16$ level of feature grid with resolution sampled in geometry space to combine different frequencies features:

$$R_l := \lfloor R_{min} b^l \rfloor, b := \exp(\frac{\ln R_{max} - \ln R_{min}}{L-1}), \quad (1)$$

where $R_{min} = 16, R_{max} = 2048$ are the coarsest and finest resolitions, respectively. Each grid includes up to T feature with a dimension of 2. In the coarse level where $R_l \leq T$, the feature grid is stored densely. For the finer level where $R_l > T$, we follow the Instant-NGP [8] to apply a spatial hashing function to index the feature vector from the hashing table:

$$h(x) = (\oplus_{i=1}^{3} x_i \pi_i) \text{mod} T \quad (2)$$

where $\oplus$ is the bit-wise XOR operation and π_i are unique, large prime numbers. The size of the feature vector table T is set as 2^{19} similar to [8, 16]. By concatenating the trilinear interpolated queried vector from each scale, we append it with the vanilla fixed frequency position embedding of the point coordinates as the input for SDF prediction network [16].

Geometry initialization for object compositional neural implicit surfaces To train a model which takes coordinates position as input and then predicts SDF, a good initialization could serve an important role in the optimization. A commonly used technique is the geometry initialization proposed in [1]. The key design lies in the initial weight to create an SDF field of a sphere in 3D space. In our object-compositional setting, we improve it by manipulating the bias term in the last layer of MLP to create a different radius of the sphere for objects and backgrounds. Specifically, we set the bias term in the channel of common objects as half of that in the channel of background SDF. This design will make sure the objects lie inside the background at the beginning of model optimization. We noticed that this could help in alleviating some object-missing issues during model training. The default radius set for the background object is 0.6-0.9 to cover the camera trajectory. To make sure the

minimum operation stays meaningful, we set the region inside the sphere as positive for the background object SDF so that it won't influence the inner object SDF.

Details about Normal and Depth loss As the monocular depth extracted from pre-trained model [3] is not a metric depth, MonoSDF adopts a scale-invariant loss [4, 10] by solving a least-square problem:

$$(w, q) = \arg\min_{w,q} = \sum_{\mathbf{r}\in\mathcal{R}} (w\hat{D}(\mathbf{r}) + q - \bar{D}(\mathbf{r}))^2. \quad (3)$$

Here the $\bar{D}(\mathbf{r})$, $\hat{D}(\mathbf{r})$ are the pesudo depth and rendered depth, respectively. This equation has a closed-form solution when sampling larger than 2 points. We solved w, q individually at each iteration for a batch of randomly sampled rays within a single image. The main reason behind it is the depth map predicted by the pre-trained model may differ in scale and shift and the predicted geometry will change at each iteration. Then the depth loss can be defined as:

$$\mathcal{L}_{depth} = \sum_{\mathbf{r}\in\mathcal{R}} \|w\hat{D}(\mathbf{r}) + q - \bar{D}(\mathbf{r})\|^2. \quad (4)$$

As for the normal loss, we not only force the scale of the normal vector but also the angle similarity for predicted normal and pseudo-normal. The predicted normal vector $\hat{N}$ can also be obtained from the volume rendering result of the gradient of the SDF field, similar to depth and RGB color. The loss can be defined as follow:

$$\mathcal{L}_{normal} = \sum_{\mathbf{r}\in\mathcal{R}} \|\hat{N}(\mathbf{r}) - \bar{N}(\mathbf{r})\|_1 + \|1 - \hat{N}(\mathbf{r})\bar{N}(\mathbf{r})\|_1, \quad (5)$$

where $\hat{N}(\mathbf{r})$, $\bar{N}(\mathbf{r})$ are the rendered normal and pseudo normal from OmniData [3] respectively.

Object Distinction Loss We also provide the idea of the implementation of object distinction regularization loss. Because we only apply this loss to object SDFs which is not the minimum value at this point. We first get the SDF vector $\mathbf{d}(\mathbf{p}) = (d_{\mathcal{O}_1}(\mathbf{p}), d_{\mathcal{O}_2}(\mathbf{p}), \ldots, d_{\mathcal{O}_K}(\mathbf{p}))$, then we use the minimum operation to get the scene SDF, $d_\Omega(\mathbf{p})$. We adopt a simple trick to eliminate the influence by subtracting the output of that from the minimum SDF in the loss:

$$\begin{aligned} &\sum_{d_{\mathcal{O}_i}(\mathbf{p})\neq d_\Omega(\mathbf{p})} \text{ReLU}(-d_{\mathcal{O}_i}(\mathbf{p}) - d_\Omega(\mathbf{p}))] \\ &= \sum_{i=1,\ldots,K} [\text{ReLU}(-d_{\mathcal{O}_i}(\mathbf{p}) - d_\Omega(\mathbf{p}))] \\ &\quad -\text{ReLU}(-\min \mathbf{d}(\mathbf{p}) - d_\Omega(\mathbf{p})), \end{aligned} \quad (6)$$

Due to the minimum operation being differentiable, we are able to calculate this loss and backpropagate the gradient.

C. Evaluation Metric

We provide the definition of the evaluation metrics we used in the main document.

Metric	Definition
Accuracy	$\text{mean}_{\mathbf{p}\in\mathbf{P}}(\min_{\mathbf{q}\in\mathbf{Q}} \|\mathbf{p}-\mathbf{q}\|_1)$
Completeness	$\text{mean}_{\mathbf{q}\in\mathbf{Q}}(\min_{\mathbf{p}\in\mathbf{P}} \|\mathbf{p}-\mathbf{q}\|_1)$
Chamfer-L1	0.5 * (Accuracy + Completeness)
Precision	$\text{mean}_{\mathbf{p}\in\mathbf{P}}(\min_{\mathbf{q}\in\mathbf{Q}} \|\mathbf{p}-\mathbf{q}\|_1) < 0.05$
Recall	$\text{mean}_{\mathbf{q}\in\mathbf{Q}}(\min_{\mathbf{p}\in\mathbf{P}} \|\mathbf{p}-\mathbf{q}\|_1) < 0.05$
F-score	$2 * \text{Precision} * \text{Recall}/(\text{Precision+Recall})$

Table 1. **Evaluation Metric Calculation**. We provide the equation for computing the quantitative metric used in the experiment. Given the sampled point cloud from ground-truth $\mathbf{P}$ and predicted result $\mathbf{Q}$, all the metrics can be calculated as shown above.

D. More Experimental Results

D.1. More Details about Experimental Setting

We use 8 scenes from Replica [11] following [16, 6] and 4 scenes from ScanNet [2] following [16, 5] for evaluation. The groud-truth meshes of vMap [6] are from here[1] and MonoSDF [16] are from here[2]. vMAP also provides the ground truth object mesh of Replica dataset. We also evaluate our object reconstruction results compared with these data.

D.2. More results on Replica Dataset

We also provide the variant of ObjectSDF* with distinction regularization loss in Tab 2. We notice that adding the object distinction regularization loss into ObjectSDF* can further improve the quantitative results. However, the semantic field design limits the surface reconstruction quality, and the quantitative result from 'ObjSDF w reg' is still worse than 'Ours w/o reg' on the Replica dataset. It demonstrates the effectiveness of occlusion-aware object opacity rendering in improving surface reconstruction quality. We provide more results in Table. 4.

D.3. More results on ScanNet

We provide more quantitative results of Scannet. The results of ObjectSDF*, Ours w/o reg, and Ours are provided in Tab. 3. We found that replacing the semantic field design with the occlusion-aware object opacity training scheme could also show superiority in scene reconstruction quality. The object distinction loss also performs an important role in further improving the quantitative results and making them achieve state-of-the-art performance. It suggests that the combination of object distinction loss and occlusion-aware object opacity rendering scheme is necessary. Besides that, we also find simpling applying the design of ObjectSDF* has already improved the result of MonoSDF (Multi-Res Grid) by a clear margin. It further reassures the

[1] https://github.com/kxhit/vMAP#results

[2] https://github.com/autonomousvision/monosdf/blob/main/scripts/download_meshes.sh

Method	Model Components			Scene Reconstruction		Object Reconstruction	
	Object Guidance	Mono Cue	Regularizer	Chamfer-L1↓	F-score ↑	Chamfer-L1↓	F-score ↑
ObjSDF [13]	Semantic			22.8	25.74	7.05	59.91
ObjSDF*	Semantic	✓		4.14	78.34	4.65	74.06
ObjSDF* w reg	Semantic	✓	✓	3.96	80.58	4.18	76.82
Ours w/o reg	Occlusion Opacity	✓		3.60	85.59	3.78	79.51
Ours	Occlusion Opacity	✓	✓	3.58	85.69	3.74	80.10

Table 2. **The quantitative average results from 8 Replica scenes evaluated on scene and object reconstruction**. We show more results of the ablation study.

	Accuracy ↓	Completeness ↓	Chamfer-L1 ↓	Precision ↑	Recall ↑	F-Score ↑
MonoSDF (Multi-Res Grids) [16]	0.072	0.057	0.064	0.660	0.601	0.626
ObjectSDF* (Multi-Res Grids)	0.065	0.048	0.057	0.661	0.672	0.669
Ours w/o reg (Multi-Res Grids)	0.065	0.045	0.055	0.667	0.704	0.685
Ours (Multi-Res Grids)	0.047	0.045	0.046	0.749	0.707	0.726

Table 3. **The quantitative results of the scene reconstruction on ScanNet**. We show the ablation results of ObjectSDF++ compared with multi-resolution grid-based MonoSDF. With the introduction of object-compositional modeling, we found the scene reconstruction quality can get significant improvement.

	ObjectSDF	ObjectSDF*	ObjectSDF++
room0	17.96/5.26	2.76/3.40	2.68/3.08
room1	29.29/8.59	3.94/5.07	3.37/4.66
room2	28.62/6.15	3.30/5.07	3.03/4.02
office0	20.10/7.73	5.72/4.38	6.00/3.14
office1	31.56/11.87	6.67/4.50	4.07/3.79
office2	15.98/5.25	4.47/4.27	3.70/3.62
office3	10.29/5.25	3.35/5.01	3.10/3.88
office4	31.89/10.10	2.75/6.53	2.72/4.03

Table 4. **The quantitative results of the Chamfer distance in individual scenes on Replica, including scene/object.**. We show the individual results from ObjectSDF [13], its variant ObjectSDF* and our framework on different scenes in Replica

benefit of object-compositional modeling in improving surface reconstruction ability.

D.4. Additional Experimental Results

To solve the object compositional representation, one simple baseline is to reconstruct each object separately and recombine them together in the final scene. Therefore, we conducted a simple experiment using the same scene as Fig.3. The results show that independently learning SDF from the mask doesn't correctly give occlusion relationships, resulting in poor reconstruction. Moreover, such a baseline is laborious if there are many objects in a scene.

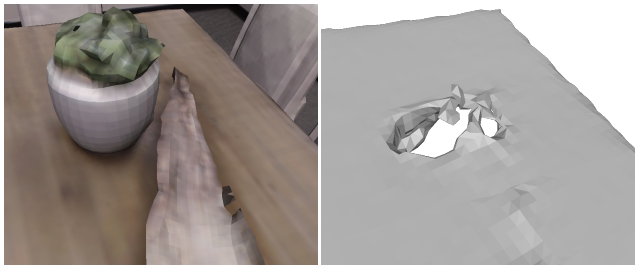

Figure 1. Reconstruct Object separately. Left: reference image, Right: reconstructed desk

E. Limitation about this framework

We improve the quality of ObjectSDF by rethinking the process of opacity rendering and object collision issues. It still exists some space to further improve it. Firstly, although we adopt the multi-resolution grid for accelerating the model convergence speed, the main focus of ObjectSDF++ is not a real-time framework for object-compositional neural implicit surfaces. The estimated training time for one scene is still about 16 hours on Pytorch (depending on how many objects are inside the scene) in a single GPU. We will explore this direction in the future. Secondly, the SDF-based representation is suitable for closed surfaces. It would be better to further extend it to support some open surfaces such as clothes *etc*. Thirdly, the underline assumption of the density transition function is that all objects are solid. Therefore, it is also a good direction to explore whether to represent transparent/semi-transparent objects in the neural implicit surface framework. We also point out that the mask used in this work is a temporally consis-

tent mask. Using an online segmentation mask could enhance the framework's applicability but require additional design for mask association between different frames. We leave it for our future work.

References

[1] Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[2] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017.

[3] Ainaz Eftekhar, Alexander Sax, Jitendra Malik, and Amir Zamir. Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3d scans. In *ICCV*, 2021.

[4] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NeurIPS*, 2014.

[5] Haoyu Guo, Sida Peng, Haotong Lin, Qianqian Wang, Guofeng Zhang, Hujun Bao, and Xiaowei Zhou. Neural 3d scene reconstruction with the manhattan-world assumption. In *CVPR*, 2022.

[6] Xin Kong, Shikun Liu, Marwan Taher, and Andrew J Davison. vmap: Vectorised object mapping for neural field slam. In *CVPR*, 2023.

[7] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020.

[8] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM TOG.*, 41(4):102:1–102:15, July 2022.

[9] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *ICCV*, 2021.

[10] Ashutosh Saxena, Sung Chung, and Andrew Ng. Learning depth from single monocular images. In *NeurIPS*, 2005.

[11] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke M. Strasdat, Renzo De Nardi, Michael Goesele, Steven Lovegrove, and Richard Newcombe. The Replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019.

[12] Andrea Tagliasacchi and Ben Mildenhall. Volume rendering digest (for nerf), 2022.

[13] Qianyi Wu, Xian Liu, Yuedong Chen, Kejie Li, Chuanxia Zheng, Jianfei Cai, and Jianmin Zheng. Object-compositional neural implicit surfaces. In *ECCV*, 2022.

[14] 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 *ICCV*, October 2021.

[15] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *NeurIPS*, 2021.

[16] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. In *NeurIPS*, 2022.