NTO3D: Neural Target Object 3D Reconstruction with Segment Anything

Xiaobao Wei[1,2,3] Renrui Zhang[4] Jiarui Wu[1,3] Jiaming Liu[1]
Ming Lu[5] Yandong Guo[6] Shanghang Zhang[1†]
[1]National Key Laboratory for Multimedia Information Processing, School of Computer Science,
Peking University [2]Institute of Software, Chinese Academy of Sciences
[3]University of Chinese Academy of Sciences [4]Shanghai Artificial Intelligence Laboratory
[5]Intel Labs China [6]AI^2 Robotics

A. Overview

The supplementary material encompasses the subsequent components.

- Supplementary experiments
  - Experiments on BlendedMVS
    - Quantitative analysis
    - Qualitative analysis
  - Comparisons on segmentation models
- Network architecture
- Details of iterative mask lifting
- Implementation of side-by-side comparison

B. Supplementary experiments

B.1. Experiment on BlendedMVS

BlendedMVS [9] is also widely used for 3D reconstruction with more complex backgrounds. We select 7 challenged scenes from the dataset. The images in BlendedMVS have a resolution of 768 × 576. One-eighth of the images are held out as test sets. In this section, we quantitatively compare the render quality between NTO3D and baseline. Since ground truth meshes aren't provided by the datasets, we further qualitatively compare the reconstruction meshes between NTO3D and baseline. Since training with or without masks has a significant impact on reconstruction quality, we take NeRF [5] and NeuS [7] as our baselines and divide them into two settings.

Quantitative analysis As shown in Tab. 1, our proposed method NTO3D achieves competitive performance on novel view synthesis in the selected scenes compared with baseline. Training with masks often leads to higher rendering performance, since the background is noisy and significantly affects the learning of target objects. With the proposed techniques, the model can focus on the target object with masks obtained by the proposed 3D Occupancy Field. The results demonstrate that our iterative lifting method can produce high-quality masks that help the neural fields to converge better.

Table 1. Quantitative comparisons with other methods on the task of novel view synthesis. Mean represents the average value of PSNR and SSIM.

Scene	Bell	Clock	Statue	Shoe	Sculpture	Bread	Durian	Mean
				Train w/o mask setting				
PSNR(NeuS)	22.46	29.67	22.18	28.43	21.54	24.27	24.25	24.69
PSNR(NeRF)	26.14	26.57	20.32	23.57	18.85	22.90	33.92	24.61
SSIM(NeuS)	0.845	0.902	0.874	0.918	0.789	0.960	0.871	0.880
SSIM(NeRF)	0.941	0.875	0.841	0.839	0.732	0.901	0.981	0.872
				Train w/ mask setting				
PSNR(NeuS)	24.06	**34.77**	22.47	26.07	28.87	29.17	24.45	27.12
PSNR(Ours)	**29.29**	29.67	**32.74**	**35.39**	**31.63**	**34.87**	**29.41**	**31.86**
SSIM(NeuS)	**0.910**	**0.970**	0.887	0.888	**0.924**	0.916	**0.920**	0.916
SSIM(Ours)	0.899	0.903	**0.934**	**0.987**	0.892	**0.966**	0.898	**0.926**

Qualitative analysis As we can see in Fig. 1, for baselines that train without masks models background as well and generate meshes with a background. With the help of iterative mask lifting, NTO3D can output fine-grained masks that only segment the target objects. Thus the meshes generated by NTO3D are more precise. Additionally, we can witness that although NeuS contains a background model that helps to distinguish foreground objects, it fails when facing a complex reconstruction environment.

B.2. Comparisons on segmentation models

SemanticNeRF [11] lifts the 2D semantic map to 3D space while ours NTO3D utilizes a 3D SAM Feature Field to lift 2D features into 3D. To make a fair comparison, we take 2D foreground masks generated by SAM to supervise SemanticNeRF and compare surface reconstruction results with NTO3D.

As shown in Tab. 2, although SemanticNeRF utilizes masks from SAM to supervise the semantic branch, it fails

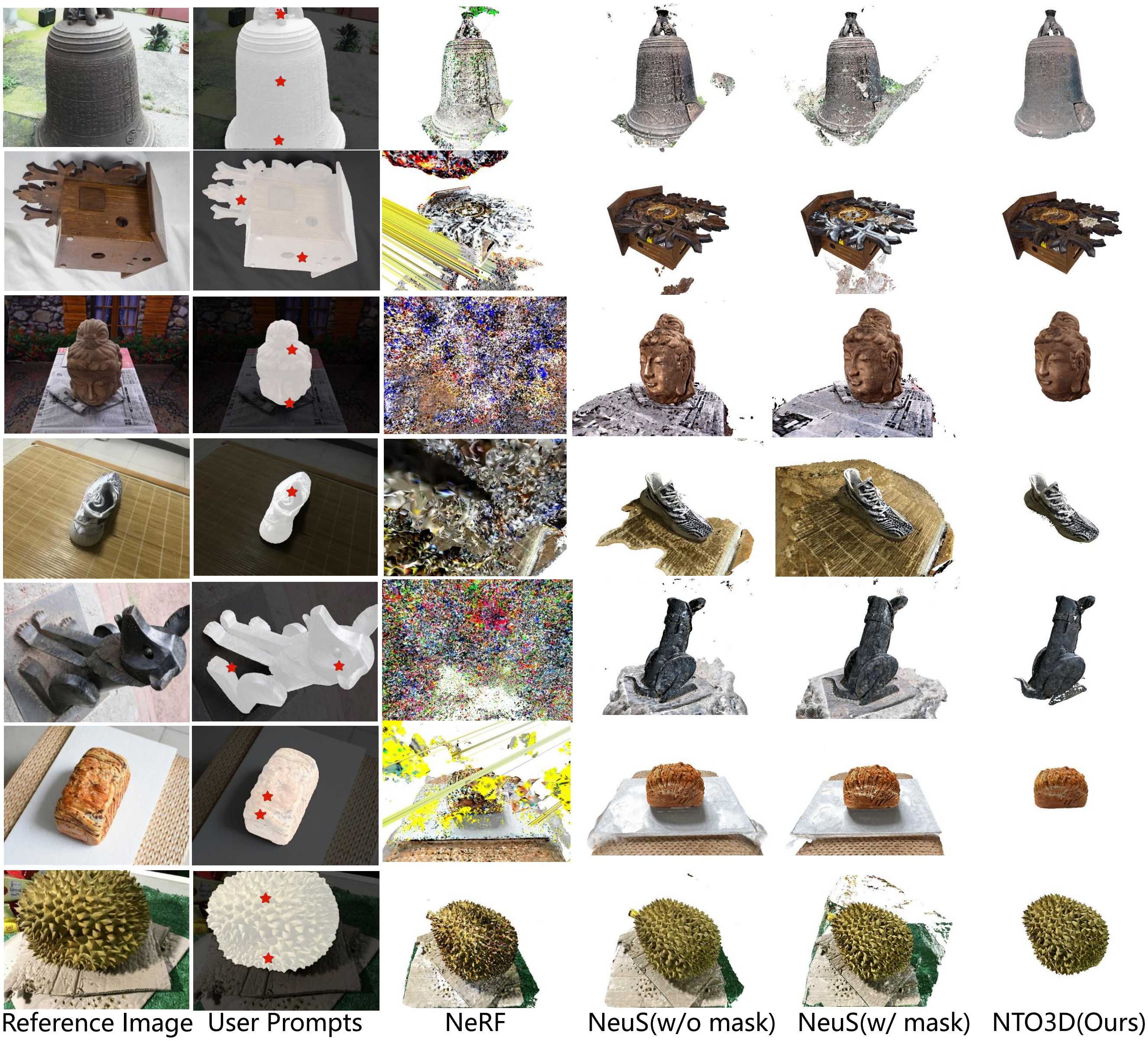

Figure 1. Qualitative comparison on BlendedMVS.

Table 2. Chamfer distance comparison on different segmentation models.

Scan ID	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
SemanticNeRF	1.65	2.36	1.21	1.34	2.31	1.28	1.78	2.65	1.36	1.71	0.96	1.74	1.21	1.35	1.24	1.61
NTO3D(Ours)	0.82	1.14	0.60	0.35	1.01	0.53	0.63	1.31	0.86	0.73	0.51	1.15	0.45	0.42	0.46	0.73

to reconstruct high-quality surfaces since it focuses on neural rendering instead of neural reconstruction. It is also worth noting that NTO3D is dedicated to jointly segmenting and reconstructing target objects while SemanticNeRF focuses on semantic segmentation.

C. Network Architecture

We introduce a Neural Target Object 3D Reconstruction model called NTO3D, to efficiently leverage the benefits of both neural field and SAM to reconstruct objects indicated by humans. NTO3D is based on the neural surface reconstructor Instant-NSR [10], which is similar to Instant-NGP [6]. Our NTO3D consists of three concatenated MLPs: a 5-hidden-layer SDF MLP M_s, a 3-hidden-layer color MLP M_c, and a 5-hidden-layer feature MLP M_f both 64 neurons wide. The same as Instant-NSR, we replace the original ReLU activation with Softplus and set β = 100 for the activation functions of all the hidden layers in

M_s. The input of the M_s is the concatenation of the 3 input spatial location values of each 3D sampled point and the 32 output values from the hash encoded position. Then, we apply a truncated function to the output SDF value which maps it to $[-1, 1]$ using the sigmoid activation. M_c also adds view-dependent color variation by spherical harmonics encoding function. The input of M_c is the concatenation of the 3 input spatial location values of each 3D sampled point, the 3 estimated normal values from the approximated SDF gradient by finite difference function, the 16 output values of the SDF MLP, and the view direction decomposed onto the first 16 coefficients of the spherical harmonics basis up to degree 4. We further apply a sigmoid activation to map the output RGB color values into the range $[0, 1]$. As for the M_f, to ensure pixel features correspond with voxel features, we take the SDF features and color features from M_s and M_c as M_f input. Then we add linear normalization in the last layer and reshape the output of M_f into the size of SAM encoder features. As for the 3D Occupancy Field, we just output the features after hash encoding and apply the max mechanism to satisfy the assumption in the main text. Finally, they iteratively optimize until converge.

D. Details of Iterative Mask Lifting

In this paper, we propose a 3D Occupancy Field to iterative lift 2D masks generated by SAM [3]. Before the iterative optimization, we need to annotate the target object in one view. We don't deliberately choose the initial viewpoint but just randomly select a view that contains the target object without occlusion. Thanks to the generalization ability of the 3D Occupancy Field and SAM, the selection of the initial view does not have much impact on the convergence of the whole network. During training, we notice that the number of aggregation points affects the quality of prompts. Intuitively, more points are needed to help the SAM segment in complex scenarios. Thus in scenes with more boundaries, we need to increase the number of points to better prompt SAM. However, more points can lead to worse segmentation results due to redundant information. In practice, no more than 10 points are needed as prompts to fully describe the target object. With the help of the proposed techniques, we can obtain fine-grained masks in a short time.

E. Implementation of side-by-side comparison

Qualitative comparison with the SA3D. The side-by-side qualitative comparison can be seen in Fig. 2. It can be witnessed that NTO3D obtains higher reconstruction quality. Although SA3D [1] works well on rendering, it fails to impose constraints on the geometry of target objects. NTO3D effectively strikes a balance between neural rendering and reconstruction, thereby achieving results that surpass SA3D [1].

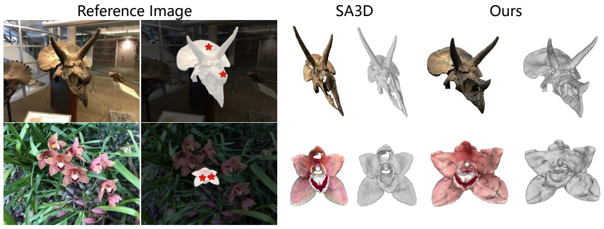

Figure 2. Side-by-side qualitative comparison on LLFF [4]. SA3D [1] lacks geometric constraints on the object surface, leading to abundant artifacts. Ours NTO3D achieves better reconstruction quality.

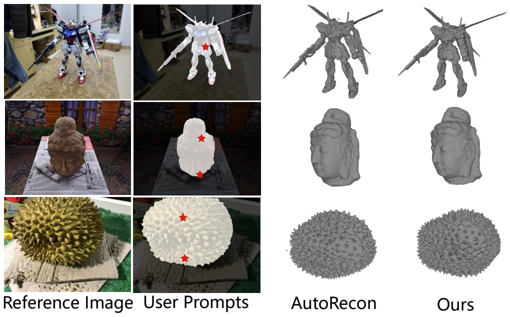

Figure 3. Qualitative comparison of NTO3D and AutoRecon [8] on BlendedMVS [9]. Please zoom in for more details. Mesh generated by NTO3D captures more details than AutoRecon [8]. In addition, NTO3D outpaces AutoRecon [8] in terms of processing speed.

Qualitative comparison with the AutoRecon. We further implement a visualization comparison between NTO3D and AutoRecon [8], which is shown in Fig. 3. Furthermore, NTO3D takes about 5 minutes to train 3D Occupancy Field in Stage-1 and takes 15 minutes to train the whole pipeline and converge in Stage-2. NTO3D is faster than AutoRecon [8], which takes about 1 hour to extract features and train to converge.

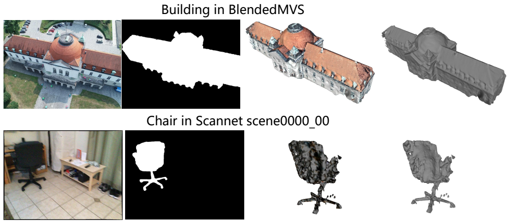

Figure 4. Results in complex scenes. NTO3D succeeds in producing high-quality segmentation and reconstruction results. The chair only appears in 11 out of 56 blur images. Sparse views and motion blur impact the geometric quality to some extent. Despite lower data quality in Scannet [2], NTO3D produces competitive results.

Reconstruction results in more complex scenes. We further implement an outdoor building in BlendedMVS [9] and a chair in Scannet [2] "scene0000_00" in Fig. 4. It can be witnessed that NTO3D also applies to the reconstruction of more complex scenes.

References

[1] Jiazhong Cen, Zanwei Zhou, Jiemin Fang, Chen Yang, Wei Shen, Lingxi Xie, Dongsheng Jiang, Xiaopeng Zhang, and Qi Tian. Segment anything in 3d with nerfs. In *NeurIPS*, 2023. 3

[2] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 3

[3] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 3

[4] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 2019. 3

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

[6] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 2

[7] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021. 1

[8] Yuang Wang, Xingyi He, Sida Peng, Haotong Lin, Hujun Bao, and Xiaowei Zhou. Autorecon: Automated 3d object discovery and reconstruction. In *CVPR*, 2023. 3

[9] 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. 1, 3

[10] Fuqiang Zhao, Yuheng Jiang, Kaixin Yao, Jiakai Zhang, Liao Wang, Haizhao Dai, Yuhui Zhong, Yingliang Zhang, Minye Wu, Lan Xu, et al. Human performance modeling and rendering via neural animated mesh. *arXiv preprint arXiv:2209.08468*, 2022. 2

[11] Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew J Davison. In-place scene labelling and understanding with implicit scene representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15838–15847, 2021. 1