NeRF-RPN: A general framework for object detection in NeRFs Supplementary Material

Benran Hu[1*] Junkai Huang[1*] Yichen Liu[1*] Yu-Wing Tai[1,2] Chi-Keung Tang[1]
[1]The Hong Kong University of Science and Technology [2]Kuaishou Technology

Please watch the supplementary video for visualizing our 3D region proposals on NeRFs at https://youtu.be/M8_4Ih1CJjE.

1. More Details on Dataset Construction

Hypersim As mentioned, we perform extensive cleaning based on the NeRF reconstruction quality. The number of camera poses on each trajectory in the Hypersim dataset is limited to 100, which is too sparse for NeRF training for many larger scenes, and usually produces fuzzy NeRF results strewn with a lot of dangling reconstruction errors or "floaters" to be removed. To remove these unsatisfactory scenes, we train NeRF models on all the scenes, and use a subset of training poses together with randomly interpolated poses as validation camera poses to examine the NeRF quality. We use the NeRF implementation from instant-NGP [4] and run at least 10k training iterations for each scene. By manually checking the NeRF rendering results, we filter out the following types of scenes: 1) scenes containing no objects; 2) scenes where a significant number of object bounding boxes are missing; 3) scenes that are too blurry, or the objects which cannot be clearly separated from floaters. After cleaning the scenes, we further clean the object bounding boxes based on the criteria aforementioned in the paper.

3D-FRONT We spent much effort to split complex scenes into individual rooms and cleaning up bounding boxes. In order to obtain data with suitable size for NeRF training, we first manually partition each scene into individual rooms according to the given layout of the scene. For each selected room, we generate 200∼300 camera poses, including 100∼150 general views randomly distributed in the room, and 15∼20 close-up views for each object within the given room. With these poses, we use [1] to render 2D images for NeRF training.

Read-World Data SceneNN is a real-world indoor dataset with around 100 scenes, where RGB-D images with predicted poses, bounding boxes of objects and reconstructed meshes are provided for each scene. We first filter the images by choosing the image with highest sharpness (variance of Laplacian) among every 20 consecutive frames. Then, we project bounding boxes onto chosen images using camera poses to determine camera pose correctness and eliminate incorrect camera poses manually. A total of 16 scenes survive the above, and we use [4] to reconstruct them.

2. Ablation on NeRF Sampling Strategies

To investigate how view-dependent radiance information from NeRF affects the performance of our method, we experiment the following sampling patterns:

1. use density only;
2. in addition to density, use the average radiance sampled from 18 fixed viewing directions in the form of $(\cos(\phi)\cos(\theta), \cos(\phi)\sin(\theta), \sin(\phi))$, where $\phi \in \{\frac{\pi}{3}, 0, -\frac{\pi}{3}\}$, $\theta \in \{\frac{k\pi}{3} \mid k \in \mathbb{N}, 0 \leq k \leq 5\}$;
3. in addition to density, use the average radiance sampled from all training camera viewing directions;
4. similar as 3) above, but only average from training camera views of which the viewing frustum contains the sample point. If a sample point is invisible in all frustums, we use the same scheme as 3) above;
5. in addition to density, use the coefficients of the Spherical Harmonics (SH) at the sample point up to degree $l = 3$. The SH function is fitted similarly as in [5] by uniformly sampling radiance from 300 directions on a sphere.

Table 1 shows the results of different sampling methods above on the 3D-FRONT test set, using VGG19 as the backbone and the anchor-free RPN head. The results might be counter-intuitive as finely-curated radiance information impairs the performance. However, we speculate that density alone is sufficient for the region proposal task as it involves only a binary classification between objects and background

*Equal contribution. The order of authorship was determined alphabetically.

†This research is supported in part by the Research Grant Council of the Hong Kong SAR under grant no. 16201420.

Methods	Recall		AP	
	0.25	0.50	0.25	0.50
Density only	95.6	**82.4**	**87.9**	**71.7**
Fixed directions	**96.3**	77.9	84.1	66.3
All cameras	95.6	75.7	86.4	64.0
Filtered cameras	**96.3**	71.3	86.5	62.1
SH coefficients	95.6	69.9	83.2	57.3

Table 1. Ablation results of NeRF sampling methods. Reported metrics are calculated on the top 2500 proposals after NMS. Filtered cameras refer to removing training camera views where the sample is outside of the viewing frustums.

which relies less on object semantics. Additionally, in this case, such extra radiance information may lead to more severe over-fitting and thus lower performance on a relatively small dataset such as 3D-FRONT. Nevertheless, the semantics carried in radiance information may be helpful for downstream classification tasks or the detection of secondary object structures.

3. Objectness Classification

As mentioned in Section 3.4, we implement a binary objectness classification model. We choose Swin-S [2] as the backbone in the experiments and use the top 2,500 proposals from RPN after NMS with an IoU threshold of 0.3. We fine-tune the feature extractor trained on RPN with AdamW [3], an initial learning rate of 0.0001, and a weight decay of 0.0001. We set $\lambda = 5.0$ in the loss function and also adopt the same augmentation strategy in RPN training. During testing, we use ROIs with objectness scores larger than 0.5 to calculate the average precision (AP). Table 2 illustrates our results. We find that the APs do not increase as expected and we speculate that this results from the limited resolution of the feature volumes. The ROIs projected onto the coarser-level feature volumes can have similar or smaller sizes compared to our ROI pooling output, while a rotated interpolation over these low-resolution feature volumes can lead to high resampling errors and cannot produce precise features for each rotated ROI. Moreover, the quality of the NeRF models can also affect the ROI quality and the objectness classification performance. However, our objectness classification architecture might still be useful in many downstream tasks like object detection where the ROI features are required, especially when a higher-resolution feature pyramid is supplied.

4. 2D Projection

We project the 3D bounding boxes as aforementioned. However, we believe 3D features from NeRF already contain sufficient information for precise 3D bounding box regression, which renders the 2D projection loss redundant. This is corroborated by the results in Table 3, where introducing the extra loss does not help with performance. Therefore, we do not use 2D projection loss for other results presented in this paper. The 2D projection loss may however still be helpful when 3D supervision is unavailable.

Methods	Hypersim		3D-FRONT	
	AP_{25}	AP_{50}	AP_{25}	AP_{50}
Anchor-based	**24.6**	**6.2**	**51.8**	**26.6**
+Objectness cls.	12.1	1.2	36.0	7.4
Anchor-free	**27.7**	**7.7**	**78.7**	**41.0**
+Objectness cls.	14.7	2.5	44.7	16.8

Table 2. Ablation of the objectness classification component on Hypersim and 3D-FRONT, using Swin-S as the backbone.

Methods	Recall		AP	
	0.25	0.50	0.25	0.50
Anchor-based	**98.5**	63.2	51.8	**26.6**
+2D proj. loss	97.1	**65.4**	**58.4**	22.2
Anchor-free	**96.3**	**62.5**	**78.7**	41.0
+2D proj. loss	**96.3**	57.4	78.2	**41.3**

Table 3. Ablation of the 2D projection loss run on 3D-FRONT, using Swin-S as the backbone.

5. Video Results

Please watch the supplementary video for moving 3D demonstration of our test-time region proposal results and heat maps in various examples.

References

[1] Maximilian Denninger, Martin Sundermeyer, Dominik Winkelbauer, Youssef Zidan, Dmitry Olefir, Mohamad Elbadrawy, Ahsan Lodhi, and Harinandan Katam. Blenderproc. *arXiv preprint arXiv:1911.01911*, 2019. 1

[2] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2

[3] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*. OpenReview.net, 2019. 2

[4] 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. 1

[5] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. PlenOctrees for real-time rendering of neural radiance fields. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 1