Multi-task View Synthesis with Neural Radiance Fields
Supplementary Material

In this supplementary material, we first present a demo video to show the multi-task view synthesis quality of the proposed MuvieNeRF in Section A. Next, in Section B we provide additional qualitative results, including more visualizations for the two main datasets and results on other out-of-distribution datasets. We conduct additional experimental evaluations to analyze the behavior of our model under different settings in Section C. We further present the multiple run results of our model and the compared methods in Section D, demonstrating that our MuvieNeRF consistently achieves the best performance. Finally, we include additional details about our model implementation and dataset processing in Section E.

A. Demo Video

We include a demo video in the supplementary zip file. This video shows the synthesis quality of our MuvieNeRF when performing a **zero-shot adaption** on testing scenes in the Replica dataset [20].

We regard the 50 views in the testing scene "room_0" as anchor views and use a linear interpolation on the camera pose of the 50 anchor views to obtain camera poses between the adjacent anchor views. For each pair of anchor views, we interpolate 4 views, thus making a total of 246 views. We render the RGB, surface normal, shading, edge, keypoint, and semantic label maps using our MuvieNeRF model for each of the 246 views at 16 FPS.

In the demo video, we also compare our performance with a state-of-the-art conventional discriminative multi-task learning method [25] in the hybrid setting, which is trained with ground-truth annotations and tuned with NeRF's predictions. Our model is able to make more accurate and consistent predictions. In comparison, the conventional multi-task learning method generates predictions that are noisy and inconsistent across different views, indicating the importance of our joint modeling strategy of different tasks and our designed CTA and CVA modules to foster multi-task information flow and cross-view consistency.

Datasets	ScanNet [4]	TartanAir [23]	LLFF [13]	BlendedMVS [24]
Number of scenes	4	4	8	2
Resolution	384×288	640×480	1008×756	768×576
Contents	Indoor	Indoor, Outdoor	Indoor, Outdoor, Object	Object

Table A. Detailed information about the four out-of-distribution datasets, which contain indoor, outdoor, and/or even object-centric scenes.

B. More Visualizations

We provide more qualitative results from the following two aspects: **(1)** visual comparison with other synthesis methods and **(2)** RGB synthesis results on other out-of-distribution datasets.

B.1. Comparison with Other Synthesis Methods

Additional qualitative comparisons for all the compared methods in the Replica and SceneNet RGB-D datasets are shown in Figure A and Figure B, respectively. Our MuvieNeRF outperforms other methods with clearer and more accurate contours of the objects in scenes. This is because MuvieNeRF utilizes the CTA and CVA modules to better take advantage of the shared knowledge across different downstream tasks and the cross-view information.

B.2. Out-of-distribution Generalization

In the main paper, we present a practical application of our proposed MuvieNeRF to show that the multi-task information learned from one dataset can be generalized to the scenes in other datasets. We use MuvieNeRF trained on the Replica dataset to perform a zero-shot adaption on out-of-distribution datasets: LLFF [13], TartanAir [23], ScanNet [4], and BlendedMVS [24] containing indoor, outdoor and even object-centric scenes. The detailed information of these four datasets is listed in Table A.

The RGB synthesis results on those out-of-distribution datasets are shown in Figure C. We can observe that our model renders higher-quality RGB images from novel views with sharper contours compared to the GeoNeRF baseline. The underlying reason lies in the joint modeling of edges and surface normal during training, which makes RGB prediction more precise even for the out-of-distribution datasets.

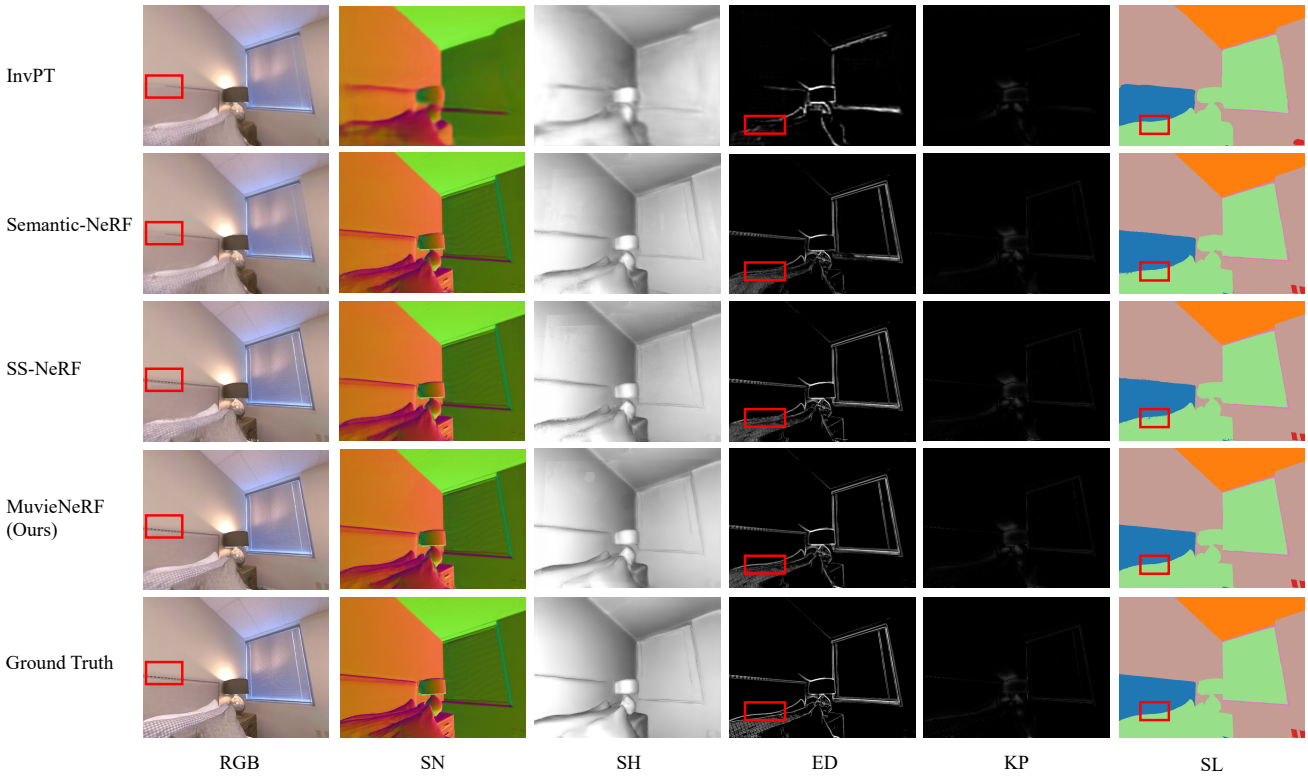

Figure A. Additional qualitative results on one testing scene in the Replica dataset. Our proposed MuvieNeRFoutperforms other methods with more accurate predictions and sharper boundaries, which demonstrates the effectiveness of the multi-task and cross-view information modeled by the CTA and CVA modules. **Zoom in to better see the comparison.**

Index	Training Scene Name	SN	SH	ED	KP	SL
1	apartment_0 (a)	×	×	✓	×	✓
2	apartment_0 (b)	✓	×	✓	✓	✓
3	apartment_1	✓	✓	✓	✓	×
4	apartment_2 (a)	×	✓	✓	✓	✓
5	apartment_2 (b)	✓	✓	✓	✓	✓
6	apartment_2 (c)	×	✓	×	✓	×
7	FRL_apartment_0 (a)	✓	✓	✓	✓	✓
8	FRL_apartment_0 (b)	✓	×	×	✓	✓
9	hotel_0 (a)	×	✓	✓	✓	×
10	hotel_0 (b)	✓	✓	✓	×	×
11	hotel_0 (c)	×	✓	✓	×	✓
12	hotel_0 (d)	✓	✓	×	✓	✓
13	office_0 (a)	✓	×	×	×	✓
14	office_0 (b)	×	✓	✓	×	✓
15	office_0 (c)	✓	✓	×	✓	×
16	office_2	×	✓	✓	✓	✓
17	room_2 (a)	✓	✓	✓	×	×
18	room_2 (b)	✓	×	×	✓	✓

Table B. Simulated *federated training* setting where some of the task annotations for certain training scenes are unavailable.

C. Additional Experimental Evaluation

We provide additional experimental evaluations to understand the behavior of MuvieNeRFand its capability from various aspects: (1) we evaluate our model in a *federated training* setting; (2) we ablate the CTA module with a lightweight choice of the cross-stitch module [14]; (3) we report the results with a half-sized training set; (4) we provide an additional comparison for the discriminative models with extra data; and (6) we ablate the contributions of the proposed CTA and CVA modules in the more challenging setting formulated by Equation 2.

C.1. Federated Training with Partial Annotations

In the real-world regime, it is not always possible to get access to all the different types of annotations to train a model. In this scenario, federated training [10] is widely used. To simulate the real-world regime, we propose such a setting where *every task annotation for each training scene has a 30% probability of being unavailable*. The detailed setting is shown in Table B, where only 2 scenes get access to all annotations. We train our MuvieNeRF on this subset and compare against the two NeRF-based baselines Semantic-NeRF [29] and SS-NeRF [28] trained on the full

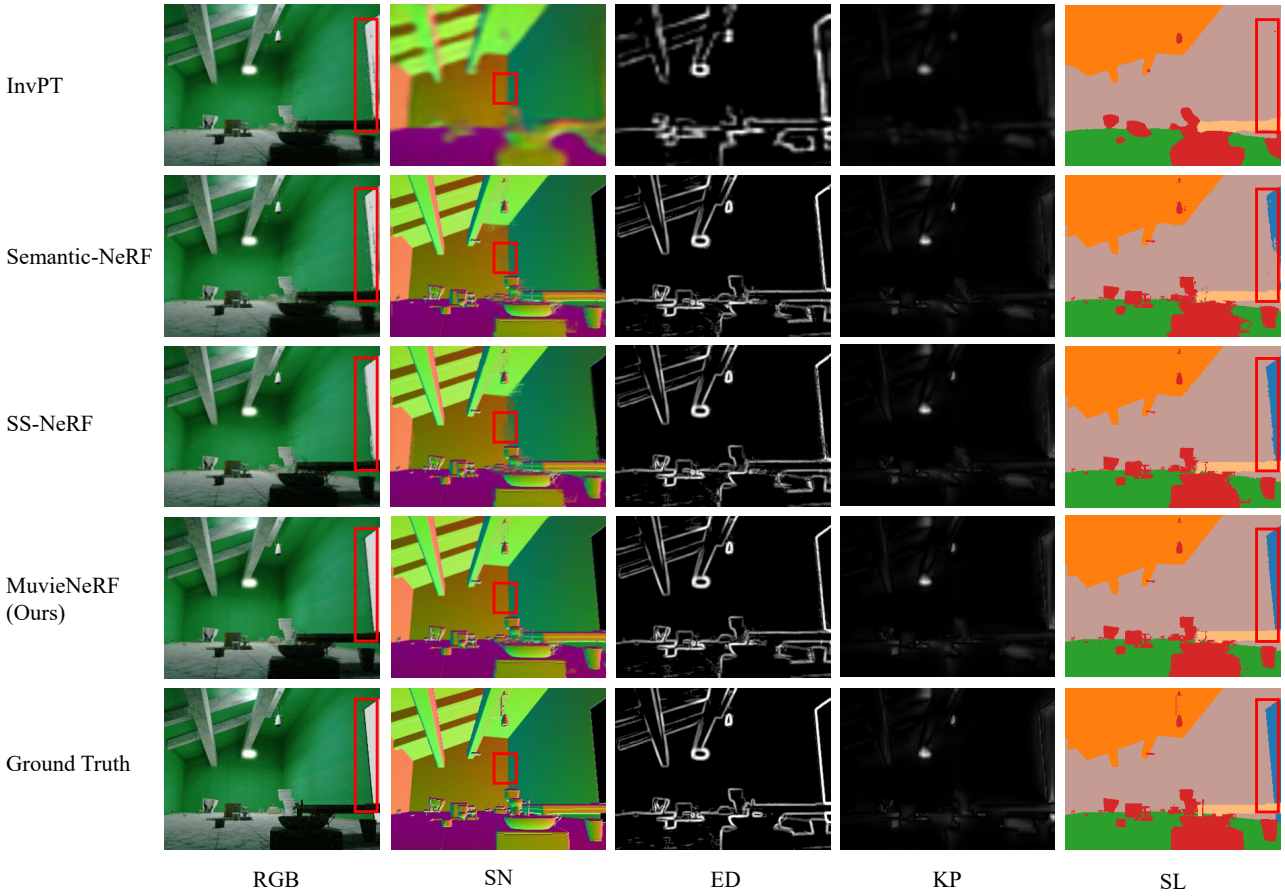

Figure B. Additional qualitative results on one testing scene in the SceneNet RGB-D dataset. Our proposed MuvieNeRFoutperforms other methods, indicating that our model benefits from the multi-task and cross-view information with the designed CTA and CVA modules. The black regions in the surface normal visualizations are due to the missing depth values in those regions. **Zoom in to better see the comparison.**

training set. The evaluation is conducted on the same testing scenes in Replica.

The results are shown in Table C. We have the following observations. First, our model still outperforms the two baselines even with missing annotations, indicating that leveraging multi-task and cross-view information in our proposed MuvieNeRF is the key to the success. Second, we still achieve comparable results to the model trained with the full training set, showing the robustness and label-efficiency of our method.

Settings	RGB (↑)	SN (↓)	SH (↓)	ED (↓)	KP (↓)	SL (↑)
Full set + Semantic-NeRF	27.08	0.0221	0.0418	0.0212	0.0055	0.9417
Full set + SS-NeRF	27.22	0.0224	0.0405	0.0196	0.0053	0.9483
Full set + MuvieNeRF	28.55	0.0201	0.0408	0.0162	0.0051	0.9563
w/o Full set + MuvieNeRF	27.86	0.0212	0.0422	0.0185	0.0053	0.9526

Table C. Comparison between our model and two baselines in a federated training setting. Our MuvieNeRF model still outperforms the two baselines and even achieves comparable results to the model trained with the full set, indicating the effectiveness and robustness of the proposed method.

C.2. Comparison with Lightweight Cross-task Modules

The novel CVA and CTA modules are designed to facilitate multi-task and cross-view information interaction, which improves the performance of MuvieNeRF. In this section, we provide an additional ablation with a lightweight choice of CTA modules. We show in the following that although the simpler module reaches comparable performance when modeling RGB together with two additional tasks, its performance significantly lags behind our novel design when handling the more challenging setting with RGB modeled with five additional tasks. It demonstrates that the designed CVA and CTA modules in our MuvieNeRF have a larger capacity for modeling multiple tasks.

Concretely, we adopt the cross-stitch [14] module for experimental evaluation. The cross-stitch module takes a simple strategy of performing a learned combination of task-

GeoNeRF MuvieNeRF (Ours) Ground Truth

Figure C. Additional qualitative RGB synthesis results on out-of-distribution datasets. **From top to bottom:** ScanNet [4], TartanAir [23], and LLFF [13]. Our MuvieNeRF yields better visual quality, demonstrating that the multi-task and cross-view knowledge learned during training can be generalized and applied to out-of-distribution datasets. **Zoom in to better see the comparison.**

specific features. More specifically, when applied in our MuvieNeRF pipeline, it functions after the "separate decoders" as

$$F_{\text{out}} = \mathbf{W} F_{\text{in}}, \tag{1}$$

where $F_{\text{in}}, F_{\text{out}} \in \mathbb{R}^{K \times V \times c}$ are the input and output of the cross-stitch module, respectively. $\mathbf{W} \in \mathbb{R}^{K \times K}$ is a learnable weight matrix with an L_2 regularization for each row. Each weight value w_{ij} measures the information the j-th component obtained from the i-th component.

The experimental comparison of our MuvieNeRF and the simpler cross-stitch implementation is shown in Table D. When modeling with only two additional tasks, the cross-stitch module could reach comparable performance to our method. However, when the number of tasks jointly learned with RGB increases to five, the cross-stitch implementation fails to serve as an efficient multi-task learning strategy. This indicates that although the simpler cross-stitch module can afford to benefit the information exchange in the easier cases when two tasks beyond RGB are jointly modeled, it does not have enough capacity to handle the more complicated relationships of five tasks along with RGB. In comparison, our design of the CVA and CTA modules is superior, which leads to the success of modeling more tasks.

Tasks	RGB (↑)	SN (↓)	SL (↑)
Cross-stitch (RGB + 2 Tasks)	**27.16**	0.0242	**0.9519**
MuvieNeRF (RGB + 2 Tasks)	26.97	**0.0229**	0.9476
Cross-stitch (RGB + 5 Tasks)	27.57	0.0219	0.9459
MuvieNeRF (RGB + 5 Tasks)	**28.55**	**0.0201**	**0.9563**

Table D. Comparison between our MuvieNeRF model design and a simpler cross-stitch [14] multi-task module. The results are averaged over the testing scenes on the Replica dataset. The simpler cross-stitch implementation can reach comparable results when the target is easier (RGB + 2 tasks), but fails to achieve satisfactory results when the target becomes more challenging (RGB + 5 tasks). In comparison, our model is able to achieve better performance with more tasks learned together.

C.3. Results with a Half-sized Training Set

We further investigate the robustness of our model by decreasing the number of scenes in the training dataset to only half of the original size. The results are shown in Table F. We could observe a similar phenomenon as the federated training result in Table C: when the number of training scenes reduces, the performance of our model only drops slightly while still outperforming the other compared methods, demonstrating the robustness and sample-efficiency of our method.

C.4. Additional Comparisons with Discriminative Models

We add the following two sets of comparisons for discriminative models on Table E: (1) we use 15K images rendered from the Replica dataset for training; (2) we use a pre-trained checkpoint (Taskgrouping-4M, the only available multi-task one with 4 tasks) on Taskonomy (∼4M data) for initialization and finetune it on Replica. All these variants still cannot outperform our MuvieNeRF, indicating that the discriminative models still **lack the ability of multi-view reasoning** even when the training data increases.

C.5. Contributions of CTA and CVA Modules with the More Challenging Setting

In Table 4 in the main paper, we dissect the individual contributions of the proposed CTA and CVA modules with our primary setting. We additionally ablate their contributions of them in the more challenging setting formulated by Equation 2. The results in Table H show similar conclusions

Model	NeRF's Images (No Tuned)					NeRF's Images (Tuned)					GT Images (Upper Bound)				
	SN (↓)	SH (↓)	ED (↓)	KP (↓)	SL (↑)	SN (↓)	SH (↓)	ED (↓)	KP (↓)	SL (↑)	SN (↓)	SH (↓)	ED (↓)	KP (↓)	SL (↑)
Taskgrouping (15k)	0.0464	0.0757	0.0418	0.0088	0.6633	0.0479	0.0531	0.0388	0.0087	0.7193	0.0438	0.0509	0.0284	0.0058	0.7509
MTI-Net (15k)	0.0533	0.0676	0.0414	0.0089	0.5509	0.0463	0.0581	0.0314	0.0079	0.6821	0.0462	0.0500	0.0271	0.0050	0.7555
InvPT (15k)	0.0463	0.0580	0.0417	0.0079	0.7157	0.0399	0.0477	0.0272	0.0057	0.7719	0.0402	0.0472	0.0257	0.0047	0.7981
Taskgrouping-4M	0.0451	-	0.0350	0.0079	0.6692	0.0313	-	0.0311	0.0066	0.7818	0.0231	-	0.0112	0.0040	0.8376
MuvieNeRF	**0.0201**	**0.0408**	**0.0162**	**0.0051**	**0.9563**	-	-	-	-	-	-	-	-	-	-

Table E. Additional comparison with discriminative models. Training discriminative models with a larger amount of data still cannot outperform our MuvieNeRF, indicating that the discriminative models still lack the ability of multi-view reasoning even when the training data increases.

Settings	RGB (↑)	SN (↓)	SH (↓)	ED (↓)	KP (↓)	SL (↑)
Full + Semantic-NeRF	27.08	0.0221	0.0418	0.0212	0.0055	0.9417
Full + SS-NeRF	27.22	0.0224	0.0405	0.0196	0.0053	0.9483
Full + MuvieNeRF	28.55	0.0201	0.0408	0.0162	0.0051	0.9563
Half + MuvieNeRF	28.11	0.0211	0.0427	0.0168	0.0054	0.9562

Table F. Comparison of training with only half training scenes in Replica. Our model still achieves relatively satisfactory results when the number of training scenes reduces to only half, indicating the sample-efficiency and robustness of our method.

to our main table and validate the proposed CVA and CTA modules are universally beneficial.

D. Multiple Runs

To further validate the robustness and good performance of our model against other methods, we show the results of multiple runs on the Replica dataset in Table G. Our MuvieNeRF consistently outperforms the single-task Semantic-NeRF [29] and the multi-task SS-NeRF [28] baselines, demonstrating the effectiveness of our model design.

E. Implementation Details

We provide the architecture of the conditional NeRF encoders and the additional U-Net [16] discriminative module we used for Section 4.5. More details of the training procedure and dataset processing are also included.

E.1. Conditional NeRF Encoders

GeoNeRF [7] encoder first uses a feature pyramid network [9] to encode input views of the scene to cascaded cost volumes [5]. Next, it masks out the input view features when the depth of the current 3D point is larger than the estimated depth in the corresponding input view. Finally, four cross-view attention operations are used to process the multi-view tokens. We refer to the official repository [1] of GeoNeRF for our implementation.

MVSNeRF [2] encoder takes a similar architecture to the GeoNeRF encoder only without the cross-view attention modules. We refer to the released codes [2] for implementation.

PixelNeRF [26] encoder uses ResNet-34 [6] as the backbone of its feature extractor. It chooses the features prior to the first four pooling layers and upsamples them to be in the same shape as the input RGB images to obtain the multi-scale features. Next, the sampled points are projected to the image planes of the input views to obtain the projected feature from the V source views. We implement it based on the official repository [3].

GNT [21] encoder [4] also adopts ResNet-34 as the feature encoder to obtain the multi-view features from multi-view RGB inputs. We apply the same strategy as the PixelNeRF encoder to obtain the features for single 3D points. Notice that, in the original GNT model which is solely designed for RGB synthesis, the multi-view features further go through a view transformer [22]. However, the output of their transformer is not compatible with our designed decoder pipeline so we only treat the ResNet part as the encoder. Therefore, the GNT encoder serves as the single-scale version of the PixelNeRF encoder in our experiments and it can explain the reason why the GNT encoder performs the worst in our main paper.

E.2. The Additional Discriminative Module

In Section 3.4 and 4.5 we introduce the model MuvieNeRF$_D$ for the more challenging problem setting with unknown nearby-view annotations. We take the encoder-decoder structure used in [19] for the U-Net shaped module F_{UNet}, which takes RGB images as the input and predicts pixel-level scene properties.

Concretely, for the U-Net module, we use a shared encoder with the Xception [3] as the backbone and apply K light-weighted deconvolutional layers [15] to predict multiple scene properties. After the predictions, we use the 3D coordinate of the queried point to project the sampled points to the input image planes to obtain the single-pixel scene properties for the weighted sum.

E.3. Training Details

We set the weights for the six chosen tasks as $\lambda_{\text{RGB}} = 1$, $\lambda_{\text{SN}} = 1$, $\lambda_{\text{SL}} = 0.04$, $\lambda_{\text{SH}} = 0.1$, $\lambda_{\text{KP}} = 2$, and

[1] https://github.com/idiap/GeoNeRF
[2] https://github.com/apchenstu/mvsnerf
[3] https://github.com/sxyu/pixel-nerf
[4] https://github.com/VITA-Group/GNT

Tasks		RGB (↑)	SN (↓)	SH (↓)	ED (↓)	KP (↓)	SL (↑)
Training scene evaluation	Semantic-NeRF	33.79 (±0.1579)	0.0231 (±0.0013)	0.0400 (±0.0005)	0.0127 (±0.0003)	0.0037 (±0.0000)	0.9522 (±0.0017)
	SS-NeRF	34.07 (±0.2572)	0.0212 (±0.0008)	0.0379 (±0.0007)	0.0113 (±0.0005)	0.0035 (±0.0000)	0.9528 (±0.0023)
	MuvieNeRF	**34.85 (±0.1440)**	**0.0197 (±0.0003)**	**0.0352 (±0.0006)**	**0.0102 (±0.0003)**	**0.0034 (±0.0000)**	**0.9589 (±0.0009)**
Testing scene evaluation	Semantic-NeRF	26.94 (±0.3180)	0.0219 (±0.0004)	0.0410 (±0.0005)	0.0195 (±0.0018)	0.0054 (±0.0001)	0.9502 (±0.0053)
	SS-NeRF	27.65 (±0.6055)	0.0216 (±0.0010)	0.0405 (±0.0004)	0.0184 (±0.0016)	0.0053 (±0.0001)	0.9503 (±0.0070)
	MuvieNeRF	**28.50 (±0.2127)**	**0.0200 (±0.0002)**	**0.0402 (±0.0006)**	**0.0164 (±0.0004)**	**0.0051 (±0.0001)**	**0.9586 (±0.0033)**

Table G. Results of all the compared models with four multiple runs on the Replica dataset. Our MuvieNeRF consistently has better performance and overall smaller deviation among multiple runs than the single-task Semantic-NeRF [29] and the multi-task SS-NeRF [28], demonstrating the effectiveness of our model design.

Model	SN (↓)	ED (↓)	KP (↓)
MuvieNeRF$_{\text{w/o CTA}}$	0.0694	0.0256	0.0079
MuvieNeRF$_{\text{w/o CVA}}$	0.0668	0.0246	0.0076
MuvieNeRF$_D$	**0.0605**	**0.0230**	**0.0074**

Table H. Ablation study with CTA and CVA modules on Replica [20] dataset with the more challenging setting. MuvieNeRF$_{\text{w/o CTA}}$ is the variant without CTA module; MuvieNeRF$_{\text{w/o CVA}}$ is the variant without CVA module. The proposed CVA and CTA modules are universally beneficial for both problem settings.

$\lambda_{\text{ED}} = 0.4$ based on empirical observations. We use the Adam [8] optimizer with an initial learning rate of 5×10^{-4} and set $\beta_1 = 0.9, \beta_2 = 0.999$. During training, each iteration contains a batch size of 1024 rays randomly sampled from all training scenes. The number of input views is set to 5. Following [14], we adopt a two-stage training strategy. We first train all the parameters except for the self-attention modules in the CTA module for 5×10^3 iterations. Afterwards, we train the parameters in the self-attention modules along with other parameters for 1×10^3 iterations. We train our model on a single NVIDIA A100 with 40GB memory for around 2.5 hours.

E.4. Datasets Details

Replica dataset [20] is a synthetic dataset which has accurate 3D mesh, semantic annotations and depth information. For semantic labels (SL), we map the original 88-class semantic labels in Replica dataset to the commonly-used 13-class annotation defined in NYUv2-13 [18]. For surface normal (SN), we derive it from depth:

$$\text{SN}(x, y, z) = (-\frac{dz}{dx}, -\frac{dz}{dy}, 1), \tag{2}$$

where (x, y, z) is the *3D* coordinate and $\frac{dz}{dx}$, $\frac{dz}{dy}$ are the gradients of x and y with respect to z, respectively. Edge (ED) and keypoint (KP) are rendered with Canny [1] edge detector and SIFT [11]. Shadings (SH) are obtained by XTConsistency [27] which are pre-trained on indoor scenes. To better satisfy the multi-task setting in the real world with unknown camera poses, we generate the poses of each scene with COLMAP [17].

SceneNet RGB-D dataset [12] is a large-scale photorealistic dataset that allows rendering RGB images along with pixel-wise semantic and depth annotations. We use the same strategy as the Replica dataset to obtain the semantic labels and surface normal for SceneNet RGB-D. We also use Canny and SIFT to render the ED and KP annotations. The pre-trained model for SH failed to work on this dataset; therefore, we discard shadings for the evaluation on SceneNet RGB-D.

References

[1] John Canny. A computational approach to edge detection. *TPAMI*, 8(6), 1986. 6

[2] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. MVSNeRF: Fast generalizable radiance field reconstruction from multi-view stereo. In *ICCV*, 2021. 5

[3] François Chollet. Deep learning with depthwise separable convolutions. In *CVPR*, 2017. 5

[4] 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. 1, 4

[5] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *CVPR*, 2020. 5

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[7] Mohammad Mahdi Johari, Yann Lepoittevin, and François Fleuret. GeoNeRF: Generalizing NeRF with geometry priors. In *CVPR*, 2022. 5

[8] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[9] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 5

[10] Yang Liu, Anbu Huang, Yun Luo, He Huang, Youzhi Liu, Yuanyuan Chen, Lican Feng, Tianjian Chen, Han Yu, and Qiang Yang. Fedvision: An online visual object detection platform powered by federated learning. In *AAAI*, 2020. 2

[11] David G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60, 2004. 6

[12] John McCormac, Ankur Handa, Stefan Leutenegger, and Andrew J Davison. Scenenet RGB-D: 5M photorealistic images of synthetic indoor trajectories with ground truth. In *ICCV*, 2017. 6

[13] 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. *TOG*, 38(4), 2019. 1, 4

[14] Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. In *CVPR*, 2016. 2, 3, 4, 6

[15] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *ICCV*, 2015. 5

[16] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 5

[17] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 6

[18] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In *ECCV*, 2012. 6

[19] Trevor Standley, Amir Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? In *ICML*, 2020. 5

[20] 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. 1, 6

[21] Mukund Varma T, Peihao Wang, Xuxi Chen, Tianlong Chen, Subhashini Venugopalan, and Zhangyang Wang. Is attention all NeRF needs? In *ICLR*, 2023. 5

[22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 5

[23] Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. TartanAir: A dataset to push the limits of visual SLAM. In *IROS*, 2020. 1, 4

[24] 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 *CVPR*, 2020. 1

[25] Hanrong Ye and Dan Xu. Inverted pyramid multi-task transformer for dense scene understanding. In *ECCV*, 2022. 1

[26] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelNeRF: Neural radiance fields from one or few images. In *CVPR*, 2021. 5

[27] Amir R. Zamir, Alexander Sax, Nikhil Cheerla, Rohan Suri, Zhangjie Cao, Jitendra Malik, and Leonidas J Guibas. Robust learning through cross-task consistency. In *CVPR*, 2020. 6

[28] Mingtong Zhang, Shuhong Zheng, Zhipeng Bao, Martial Hebert, and Yu-Xiong Wang. Beyond RGB: Scene-property synthesis with neural radiance fields. In *WACV*, 2023. 2, 5, 6

[29] Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew Davison. In-place scene labelling and understanding with implicit scene representation. In *ICCV*, 2021. 2, 5, 6