# DeLiRa: Self-Supervised Depth, Light, and Radiance Fields – Supplementary Material –

Vitor Guizilini[1] Igor Vasiljevic[1] Jiading Fang[2] Rares Ambrus[1] Sergey Zakharov[1]
Vincent Sitzmann[3] Adrien Gaidon[1]

[1]Toyota Research Institute (TRI), Los Altos, CA
[2]Toyota Technological Institute of Chicago (TTIC), Chicago, IL
[3]Massachusetts Institute of Technology (MIT), Cambridge, MA

## 1. Dataset Details

We use ScanNet to evaluate our method. Several splits are popular in recent scene-level NeRF works, hence we consider two popular splits to compare to prior work.

**ScanNet-Frontal** follows the ScanNet split and evaluation protocol from NerfingMVS [6]: eight scenes (0000_01, 0079_00, 0158_00, 0316_00, 0521_00, 0553_00, 0616_00, and 0653_00) are selected, each with 40 images covering a local region. From these, 35 images are used for training and 5 are held out for testing. All images are resized to $484 \times 648$ resolution, and median ground truth scaling is used for depth evaluation.

**ScanNet-Rooms** follows the ScanNet split and evaluation protocol from DDP-NeRF [4]: three scenes (0710_00, 0758_00, and 0781_00) were selected, from which 18 to 20 training images and 8 testing images were extracted. All images are resized to $468 \times 624$, and median ground truth scaling is used for depth evaluation. The scenes considered are 0710_00, 0758_00, and 0781_00. To increase frame overlap, such that the multi-view photometric objective has a stronger self-supervised training signal, we included forward and backward context frames for each training image, using a stride of 5. All other methods were re-evaluated under these new conditions, using officially released open-source repositories and the guidelines described in [4].

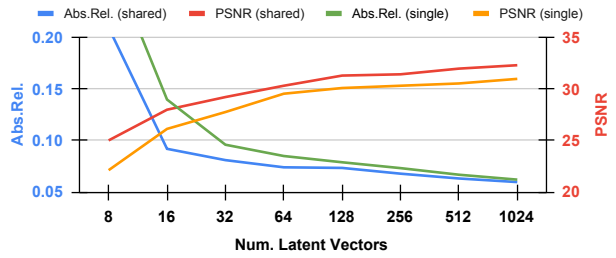
## 2. Implementation Details

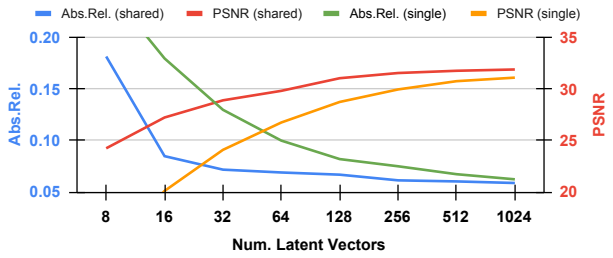
### 2.1. Training parameters

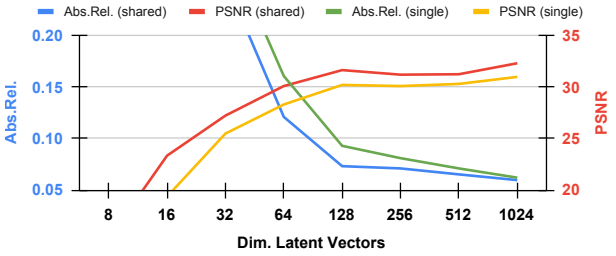
We implemented our models using PyTorch [3], with distributed training across eight V100 GPUs. We used grid search to select training parameters, including photometric loss weight $\alpha_p = 0.1$, virtual camera loss weight $\alpha_v = 0.5$, virtual camera projection noise $\sigma_v = 0.25$, depth guidance noise $\sigma_g = 0.1$, number of ray samples $K = 128$ and depth field guidance samples $K_g = 32$, minimum $d_{min} = 0.1$ and maximum $d_{max} = 5.0$ depth ranges, and a batch size of $b = 1$ per GPU. We use the AdamW optimizer [2], with standard parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, a weight decay of $w = 10^{-4}$, and an initial learning rate of $lr = 2 \cdot 10^{-4}$. We train for 4000 epochs, and multiply the learning rate by 0.8 at each 1000 epochs. A downsample of 4 is used during training for strided ray sampling, and at test time full resolution estimates are decoded. At each each iteration, 3 additional images are randomly sampled from the same scene to serve as context. Our self-supervised photometric objective includes auto-masking and minimum reprojection error, as introduced in [1].

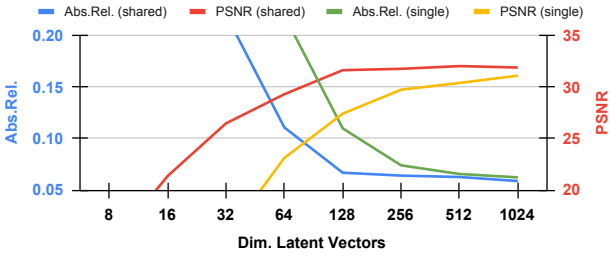
### 2.2. Architecture Details

We use $K_o = K_r = K_x = 16$ as the number of Fourier frequencies for geometric embeddings (camera center, viewing rays, and sampled 3D points respectively), with maximum resolution $\mu_o = \mu_r = \mu_x = 64$. Our volumetric ($\mathcal{E}_o \oplus \mathcal{E}_r = \mathcal{E}_{vol}$) and ray ($\mathcal{E}_o \oplus \mathcal{E}_x = \mathcal{E}_{ray}$) embeddings both have dimensionality $126 + 126 = 252$. The latent space $\mathcal{S}$ used to encode scene information is of dimensionality $N_l \times D_l = 1024 \times 1024$ (an ablation study regarding this design choice can be found in Sec. 3). Our decoder is composed of a single cross-attention layer, with GeLU as the hidden activation function, dropout of 0.1, and 2 attention heads. A single linear layer is then used to project the cross-attention output from 252 channels to the desired task dimensionality: $O_r = 4$ for radiance, $O_l = 3$ for light, and $O_d = 1$ for depth fields. Alternatively, we experimented with the deep residual network of [5] as the decoder, achieving significant improvements in light field novel view synthesis at the expense of slower inference times (Tab. 5 in the main paper).




(a) Volumetric rendering predictions

(b) Depth and light field predictions

Figure 1: **Depth and view synthesis performance** on *ScanNet* (scene 0653_00), with varying latent space shapes (larger values were not considered due to computational constraints). Blue and red lines correspond to predictions decoded from a shared latent space, and green and yellow lines to predictions decoded from latent spaces with a single representation. We observe that sharing the latent space between representations not only does not degrade results, but in fact leads to overall improvements in both view synthesis and depth estimation. These improvements are more noticeable in smaller latent spaces, particularly for depth and light field estimates, indicating that both representations are compatible for multi-task decoding.

## 3. Latent Space Dimensionality

Here we analyze the impact that changing the dimensions of the latent space $\mathcal{S}$ has on performance, both in terms of view synthesis (PSNR) and depth estimation (Abs.Rel.). Two variables are considered: the number $N_l$ of latent vectors, and the dimensionality $D_l$ of these vectors. The results of this analysis are shown in Fig. 1 (blue and red lines), where we can see that larger latent spaces indeed leads to an improvement in performance (i.e. better view synthesis PSNR and absolute relative depth error), albeit with diminishing returns. To achieve optimal results without excessive computational cost, in all experiments we used a $1024 \times 1024$ latent space. When experimenting with smaller dimensionalities, we noticed a gradual decrease in performance, followed by a steep change around $N_l = 16$ and $D_l = 128$. This sudden "phase transition" indicates the point at which the latent space becomes unable to properly encode the scene representation.

To further evaluate the properties of our learned implicit representation, we performed similar experiments in which two latent spaces are optimized, one containing only a volumetric representation, and another only a light and depth field representation. For a fair comparison, both latent spaces are still trained jointly (i.e., light and depth predictions benefit from virtual volumetric supervision, and volumetric predictions benefit from depth field guidance). The green and yellow lines in Fig. 1 show results using this setting. Interestingly, we observe that maintaining separate latent spaces for each representation not only leads to worse performance than using a single latent space (as we show in the main paper), but also that this performance gap increases when smaller latent spaces are used.

This is particularly noticeable in the case of depth and light field predictions, that experience the "phase transition" at significantly higher dimensionalities: $128 \times 256$, compared to $16 \times 128$ when using a shared latent space. We attribute this behavior to the regularization effect that the volumetric representation has on light and depth field predictions. As we show in the main paper (Sec. 4.4.2), jointly learning a volumetric representation has a similar effect to virtual camera augmentation, promoting the learning of a multi-view consistent representation for light and depth field predictions. With smaller model complexities, this multi-view consistency becomes a key factor in the learning of a useful representation for accurate predictions from novel viewpoints.

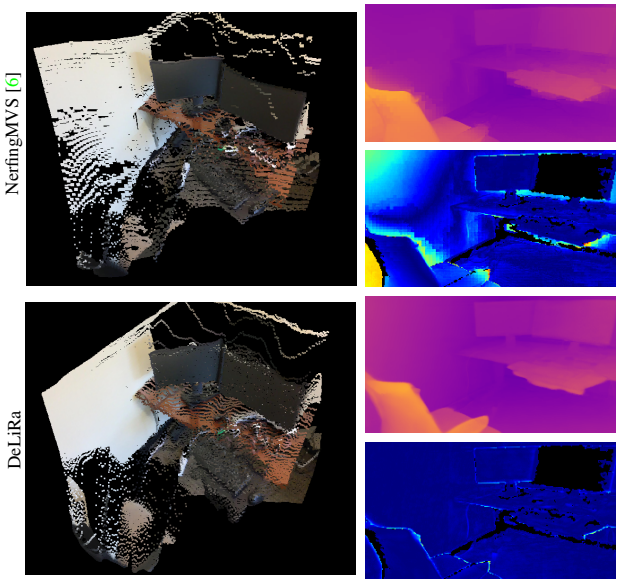
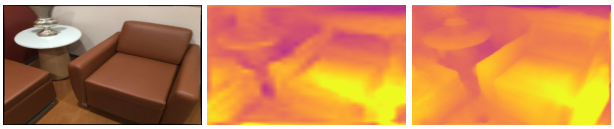



Figure 2: **Qualitative comparison between DeLiRa and NerfingMVS [6]**, for depth estimation from novel viewpoints. We show predicted depth maps (top right), depth error maps (bottom right), and reconstructed pointclouds using predicted depth and colors (left). Our approach leads to sharper depth maps, with errors concentrated on discontinuities around object boundaries, as well as better reconstruction of planar surfaces.

## 4. Additional Qualitative Results

We also include additional qualitative results to complement the ones provided in the main paper. In Fig. 2 we compare depth estimation results from DeLiRa and those produced by NerfingMVS [6], the previous state of the art in *ScanNet-Frontal*. As we can see, our predictions are sharper, with errors concentrated in discontinuities around object boundaries. DeLiRa also improves upon NerfingMVS in terms of reconstructing planar surfaces, such as the left wall and the right computer monitor. These improvements are particularly meaningful given that NerfingMVS (and most other current approaches) rely on depth priors from pre-trained networks, while DeLiRa is trained using only information from the observed scene.

In Fig. 4 we show predicted RGB images and depth maps obtained using different DeLiRa decoders (cf. Fig. 3 in the main paper). We also provide error maps for both predictions, in the form of normalized absolute differences. As a baseline, we show results produced by a model trained without our contributions (i.e., the multi-view photometric objective and the joint learning of depth, light, and radiance fields). Interestingly, this baseline model achieves novel view synthesis results comparable to our proposed architecture, however depth estimates are considerably worse. These are examples of the shape-radiance ambiguity, in which accurate novel view synthesis can still be achieved even with degenerated learned geometries, especially in cases of limited viewpoint diversity. By introducing the multi-view photometric objective as additional regularization, we promote convergence to the proper scene geometry, improving depth estimation and, by extension, novel view synthesis. Furthermore, our learned latent representation can be queried both in the form of volumetric renderings, via the radiance field decoder, as well as direct depth color estimates, via the depth and light field decoders.



(a) RGB (b) Monodepth2 (c) DeLiRa

Figure 3: **Qualitative example of depth predictions** between DeLiRa and a traditional monocular depth network.

Moreover, in Fig. 5 we show additional point clouds generated from novel viewpoints using different DeLiRa decoders, relative to the ground truth point cloud. Each point cloud is generated by lifting pixel colors to 3D space, using camera intrinsics and depth information. Ground truth point clouds use provided RGB images and depth maps, while predicted pointclouds use estimates for specific decoders (radiance for volumetric renderings, and depth and light fields for single-query renderings).

## 5. Comparison with Monodepth

Our multi-view photometric regularization is inspired by the self-supervised loss used in monocular depth estimation. For illustrative purposes, we show in Fig. 3 a qualitative comparison of depth maps from DeLiRa and monodepth2 [1], a traditional monocular depth network. Self-supervised depth estimation requires a large amount of training data to learn accurate predictions, since the multi-view photometric objective is highly ambiguous and has several local failure cases (e.g., reflective surfaces, non-Lambertian objects, textureless areas). In the indoor setting, where these types of surfaces are common, it is thus highly challenging, and for the example in Fig. 3 the self-supervised depth network fails to properly capture the observed scene geometry. In contrast, our method maintains a volumetric representation, which attenuates the effect of the self-supervised photometric loss, thus allowing for the network to more accurately reconstruct non-Lambertian surfaces, using the multi-view photometric loss only as a geometric regularizer that gradually vanishes over time.

## References

[1] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J. Brostow. Digging into self-supervised monocular

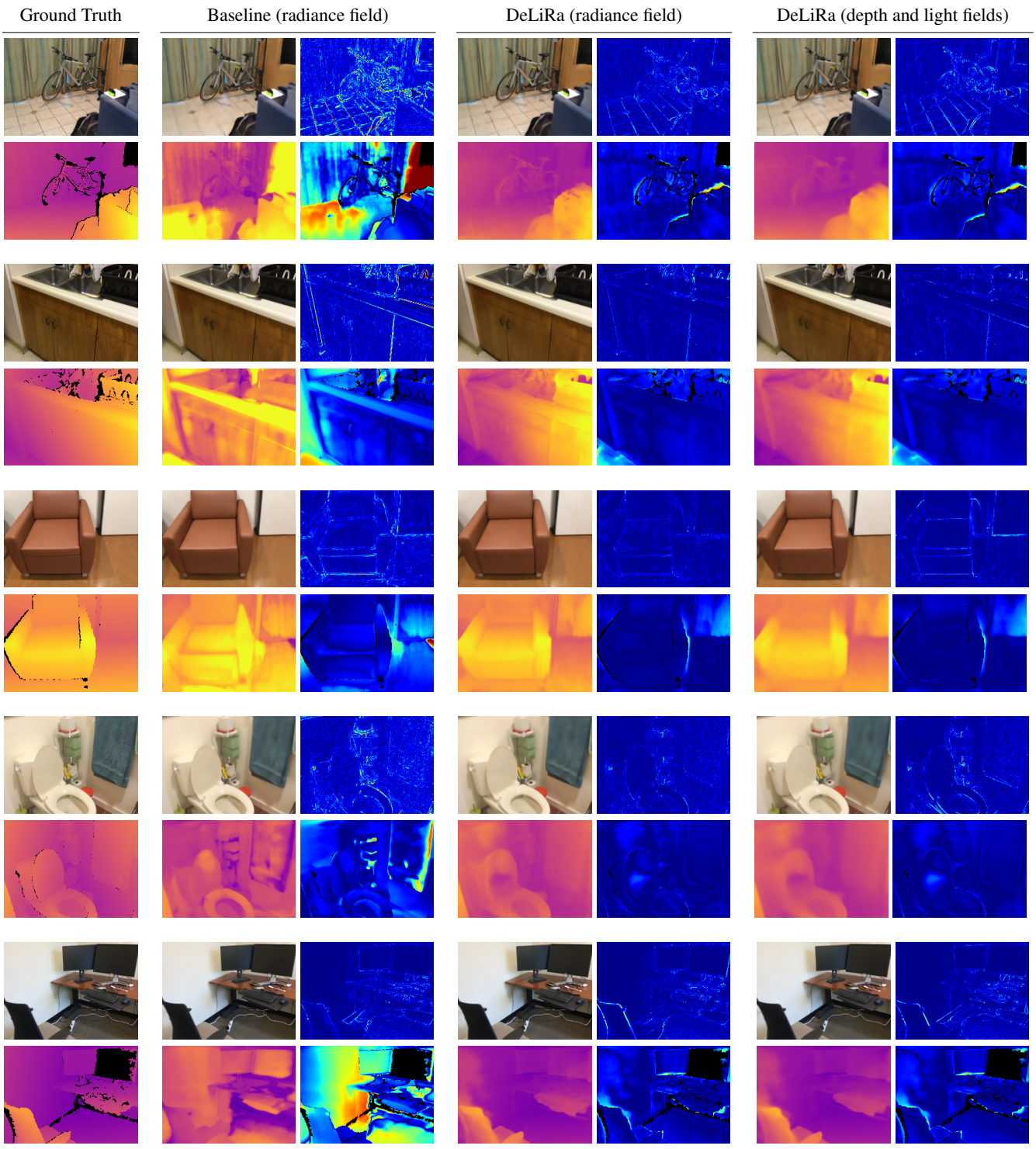



Figure 4: **Additional qualitative depth and view synthesis results** from unseen viewpoints, using different DeLiRa decoders. As a baseline, we show predictions obtained from a model trained without our contributions, leading to a degenerate learned geometry due to shape-radiance ambiguity (i.e., accurate view synthesis with poor depth predictions). RGB and depth error maps are calculated as absolute differences and respectively normalized between $[0.0, 0.5]$ and $[0.0, 1.0]$.


Figure 5: **Qualitative depth and view synthesis results** from unseen viewpoints, using different DeLiRa decoders. The first column shows ground truth point clouds, while the second and third columns show respectively pointclouds generated using radiance field predictions, and depth and light field predictions.

depth prediction. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2019. 1, 3

[2] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. 1

[3] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*. 2019. 1

[4] Barbara Roessle, Jonathan T. Barron, Ben Mildenhall, Pratul P. Srinivasan, and Matthias Nießner. Dense depth priors for neural radiance fields from sparse input views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022. 1

[5] Huan Wang, Jian Ren, Zeng Huang, Kyle Olszewski, Menglei Chai, Yun Fu, and Sergey Tulyakov. R2l: Distilling neural radiance field to neural light field for efficient novel view synthesis. In *European Conference on Computer Vision*, 2022. 1

[6] Yi Wei, Shaohui Liu, Yongming Rao, Wang Zhao, Jiwen Lu, and Jie Zhou. NerfingMVS: Guided optimization of neural radiance fields for indoor multi-view stereo. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021. 1, 3