Exploring Reconstructive Latent-Space Neural Radiance Fields

Tristan Aumentado-Armstrong[1,2,4*] Ashkan Mirzaei[1,2*] Marcus A. Brubaker[1,3,4]
Jonathan Kelly[2] Alex Levinshtein[1] Konstantinos G. Derpanis[1,3,4] Igor Gilitschenski[2]
[1]Samsung AI Centre Toronto [2]University of Toronto [3]York University [4]Vector Institute for AI

Abstract

Neural Radiance Fields (NeRFs) have proven to be powerful 3D representations, capable of high quality novel view synthesis of complex scenes. While NeRFs have been applied to graphics, vision, and robotics, problems with slow rendering speed and characteristic visual artifacts prevent adoption in many cases. In this work, we investigate combining an autoencoder (AE) with a NeRF, in which features (instead of colours) are rendered and then convolutionally decoded. The resulting latent-space NeRF can produce novel views with higher quality than standard colour-space NeRFs, as the AE can correct certain visual artifacts, while rendering three times faster. Further, we can control the tradeoff between efficiency and image quality by shrinking the AE architecture, achieving over 13 times faster rendering with only a small drop in performance.

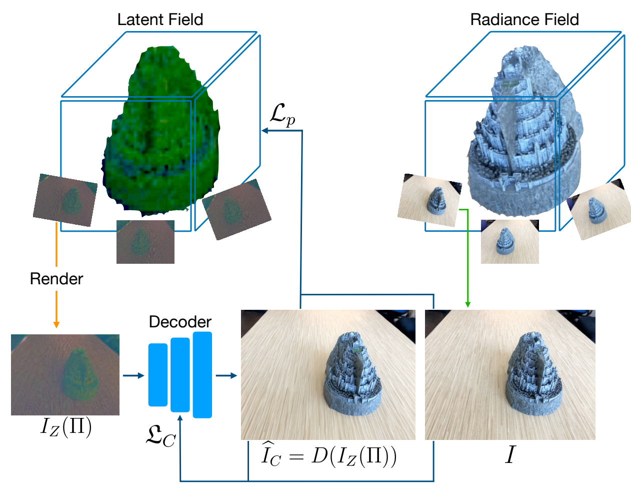

Figure 1. An overview of the ReLS-NeRF training loop. The radiance (colour) field is fit to RGB captures, as in the standard NeRF [42]. Given camera parameters, Π, ReLS-NeRF renders feature maps in the latent Z-space defined by a convolutional autoencoder (AE), $D \circ E$, for which arbitrary views can be decoded into image space. The discrepancy between the decoded rendered latents and the corresponding images (from a colour-space NeRF or real images) enables training the Z-space NeRF and the AE.

1. Introduction

Neural rendering techniques [70] continue to grow in importance, particularly Neural Radiance Fields [42] (NeRFs), which achieve state-of-the-art performance in novel view synthesis and 3D-from-2D reconstruction. As a result, NeRFs have been utilized for a variety of applications, not only in content creation [22, 88, 44, 43], but also for many robotics tasks, including 6-DoF tracking [81], pose estimation [29], surface recognition [52] or reconstruction [37], motion planning [49, 35, 1], reinforcement learning [14, 60], tactile sensing [93], and photorealistic simulation [66]. However, slow rendering and the qualitative artifacts of NeRFs impede further use cases in production.

To render a single pixel, one major bottleneck is the need for multiple forward passes of a multilayer perceptron (MLP). Replacing or augmenting the MLP with alternative representations (e.g., voxel grids [57] or feature hash-tables [47]) has been used to improve both training and inference speed. To reduce test-time rendering speed specifically, baking NeRFs into other primitive representations has been a popular approach [25, 11, 55]. Separately, alternative sampling methods [68, 5, 3, 4], different radiance models [74], and scene contraction functions [89, 4] have been proposed to reduce artifacts (e.g., "floaters" [80]). Despite these advancements, NeRFs still suffer from visual flaws and low rendering frame-rates.

In this paper, we propose an orthogonal approach for improving test-time speed and visual quality of NeRFs. By leveraging convolutional autoencoders (AEs), we can define a "NeRF" operating in latent feature space (rather than colour space), such that *low*-resolution *latent* renders can be decoded to *high*-resolution RGB renders (see Fig. 1). This offloads expensive MLP-based rendering computations to the low-cost AE. Thus, we extend the standard NeRF architecture to return point-wise latent vectors, in addition to colors and densities. As it is used for scene reconstruction, we denote the resulting combined field a Reconstructive

* Authors contributed equally.

Latent-Space NeRF (ReLS-NeRF). Beyond faster rendering, the AE can also act as an image prior, fixing some of the artifacts associated with direct NeRF renders. Empirically, our model is able to render views three times faster, while improving in multiple image and video quality metrics. Further, we demonstrate a tradeoff between visual quality and rendering efficiency: by reducing the AE size, we obtain a 13-fold speed-up, with only a small drop in quality.

2. Related Work

NeRF efficiency. While NeRFs produce results of extraordinary quality, the speed of fitting (training) and rendering (inference) remains a bottleneck for adoption in a variety of applications (e.g., [4, 66, 72]). This has prompted a myriad of approaches to increasing their efficiency. Feature grids have proven effective in expediting fitting convergence (e.g., [78, 63, 64, 5, 9, 10, 57, 47]). Other approaches include utilizing depth [13], better initializations [67], and pretraining conditional fields (e.g., [87, 79, 30]). Such improvements can be readily utilized in our own framework. Similarly, a number of methods have been proposed to enhance the efficiency of the volume rendering operation, which relies on an expensive Monte Carlo integration involving many MLP calls per pixel. These include architectural modifications [19, 76, 54, 36, 86], "baking" (precomputing and storing network outputs) [25, 55], improved sampling strategies [51, 16, 48, 38, 34], or altering the integration method itself [39, 83]. Finally, several works eschew volume rendering itself. Several representations [61, 62, 85, 17, 2, 27] use only a single sample per pixel, but struggle with geometric consistency and scalability. Similarly, one can move to a mesh-based representation and use rasterization instead [11, 21, 77]; however, this loses certain properties, such as amenability to further optimization or differentiable neural editing. Though our approach improves rendering efficiency, it is orthogonal to these methods, as it reduces the number of MLP calls per image by changing the output space of the NeRF itself.

Feature-space NeRFs. Other models have utilized *neural feature fields* (NFFs), as opposed to "radiance" fields, where rendering is altered to output learned features instead. Some NFFs [71, 33] learn to produce the outputs of pretrained 2D feature extractors; similarly, several works have considered the use of language-related features [31, 6, 59] and other segmentation signals [92, 91, 45, 44] to embed semantics into the NFF. More closely related to our work are generative modelling NFFs that decode rendered features into images via generative adversarial networks [20, 50, 84] or diffusion models [40, 58, 8]. In contrast, this paper considers the scene reconstruction problem, using a latent representation potentially amenable to downstream tasks, and investigates issues related to view consistency.

3. Methods

3.1. ReLS-NeRF Neural Architecture

Our model includes two neural modules: (i) a modified NeRF, f, which outputs a latent vector (in addition to its standard outputs), and (ii) an autoencoder (AE), with encoder and decoder networks, E and D.

We first extend the standard colour-density field of NeRF to include a latent feature vector, z, via $f(x,r) = (\sigma \in \mathbb{R}_+, c \in [0,1]^3, z \in \mathbb{R}^n)$, where x and r represent the input position and direction, and σ and c represent the output density and colour. We refer to the σ and c fields as an "RGB-NeRF", to distinguish them from the latent component of the ReLS-NeRF. Volume rendering is unchanged: for a single feature at a pixel position, p, we use

$$Z(p) = \int_{t_{\min}}^{t_{\max}} \mathcal{T}(t)\sigma(t)z(t)\,dt, \tag{1}$$

to obtain the feature value at p, where $\mathcal{T}(t)$ is the transmittance [65], and $z(t) = z(x(t), r(t))$ is obtained by sampling the ray defined by p. For camera parameters Π, we denote the latent image rendering function as $\mathcal{R}(\Pi|f) = I_Z(\Pi)$, where $I_Z[p] = Z(p)$. Replacing $z(t)$ with $c(t)$, for instance, would render colour in the standard manner, giving a colour image, $I_C(\Pi)$ (that does *not* use z). To obtain a colour image from I_Z, we simply pass it to D; i.e., view synthesis is simply $\widehat{I}_C(\Pi) = D(I_Z(\Pi))$, which can be viewed as a form of *neural rendering* (e.g., [50, 69, 15]). The benefit of using $\widehat{I}_C$ is that significantly fewer pixels need to be rendered, assuming D is an upsampler, compared to $I_C(\Pi)$; it also enables placing a prior on $\widehat{I}_C$ by choosing D appropriately.

We considered two choices of AE: (i) the *pretrained* VAE from Stable Diffusion [56], which we denote SD-VAE, and (ii) a smaller residual block-based AE [23, 28] (R32, when using a 32D latent space) that is randomly initialized. Both encoders provide an $8\times$ downsampling of the image.

3.2. Fitting Process

Setup. As in the standard NeRF scenario, we expect only a training set of multiview posed images, $S_I = \{(I_i, \Pi_i)\}_i$. The optimization proceeds in three stages: (A) AE training, (B) joint NeRF fitting, and (C) decoder fine-tuning.

AE training (A). The first phase simply trains (or fine-tunes) the AE to reconstruct the training images of the scenes, using the mean-squared error.

Joint NeRF fitting (B). In the second phase, we train the RGB and Latent components of the NeRF in conjunction with the decoder, D. Our total loss function,

$$\mathfrak{L}_B = \mathcal{L}_r + \lambda_d \mathcal{L}_d + \lambda_{\text{gr}} \mathcal{L}_{\text{gr}} + \mathcal{L}_p, \tag{2}$$

consists of the standard RGB loss on random rays, $\mathcal{L}_r$, the DS-NeRF [13] depth loss, $\mathcal{L}_d$, the geometry regularizing distortion loss [4], $\mathcal{L}_{\text{gr}}$, and a patch-based loss for training the latent component, $\mathcal{L}_p$. Given a posed image, (I, Π), the latter loss is simply the error between a sample patch, $\mathcal{P} \sim I$, and the corresponding rendered then decoded patch,

$$\mathcal{L}_p = \mathbb{E}_{\mathcal{P}\sim I,(I,\Pi)\sim S_I} \text{MSE}(\mathcal{P}, D(I_Z(\Pi))). \tag{3}$$

Decoder fine-tuning (C). Finally, we fine-tune D, utilizing a combination of the multiview posed images, S_I, and renders from the RGB component of the ReLS-NeRF. First, we sample random renders, $\widetilde{S}_I = \{(I_C(\Pi_s), \Pi_s) \,|\, \Pi_s \sim \Gamma(S_\Pi)\}_s$, where $\Gamma(S_\Pi)$ samples camera extrinsics by interpolation between a random triplet in S_Π. Optimizing

$$\mathfrak{L}_C = \gamma\delta(S_I) + (1-\gamma)\delta(\widetilde{S}_I), \tag{4}$$

where $\delta(S) = \mathbb{E}_{(I,\Pi)\sim S} \text{MSE}(I, \widehat{I}_C(\Pi))$ and $\gamma \in [0,1]$ is a weight, distills information from the RGB-NeRF into the latent renderer. Note that real training images, S_I, are used; hence, the RGB-NeRF is not a strict performance ceiling (further, D has different generalization properties).

3.3. Implementation Details

We utilize the neural graphics primitives [47] architecture, via the `tiny-cuda-nn` library [46]. All phases use Adam [32]. Note that the loss gradient from the latent component of the NeRF (i.e., from $\mathcal{L}_p$) is not back-propagated to the colour, c, and density, σ, fields. Further, we use separate features for the latent feature vector, z, and c, but render with the same σ. In other words, RGB-NeRF training is unaffected by z. (See our appendix for further details.)

3.4. Evaluation Metrics

Pixelwise and perceptual distances. We measure performance with novel view synthesis on held-out test views. In addition to the standard pixelwise peak signal-to-noise ratio (PSNR), we use perceptual losses to measure similarity as well, including LPIPS [90] and DreamSim [18]. LPIPS provides more human-like responses to low-level distortions (e.g., noise, small colour/spatial shifts), whereas DreamSim is designed to be "mid-level" metric, better able to capture large-scale and semantic differences than LPIPS (without being as high-level as, e.g., CLIP-based metrics [53, 7, 75]).
Local consistency. When examining generative models of NeRFs that use decoders, we can qualitatively see a "shimmering" effect in time (e.g., [50, 20]), which is also reminiscent of generative video model artifacts (e.g., [26, 24]). This jittering appears related to local appearance inconsistencies: since each z pixel corresponds to an RGB *patch*, as Π changes, interpolating in z-space does not perfectly approximate the correct appearance changes. Since this flaw is distinct from the artifacts observed in standard NeRFs, we devise a simple metric to detect it: the *Reprojective*

	Reference-based			Reference-free		
NeRF	PSNR↑	LPIPS↓	DS↓	DoA↑	DoT↑	RCC↑
RGB	23.52	0.37	**1.18**	80.2	72.9	**25.6**
Ours-SD	**23.81**	**0.35**	1.44	**81.5**	**77.3**	25.5
Ours-R32	23.37	0.40	1.71	76.4	74.3	25.3

Table 1. Test-view evaluation on eight LLFF scenes [41]. Reference-based metrics include PSNR, LPIPS [90], and DreamSim (DS; $\times 10$) [18]. For reference-free metrics, we use DOVER-technical (DoT), DOVER-aesthetic (DoA), and our reprojective colour consistency (RCC) measure, computed on rendered videos. Different models (rows) correspond to the standard RGB NeRF, the SDVAE-based ReLS-NeRF, and the R32-based ReLS-NeRF. ReLS-NeRF-SDVAE outperforms the RGB-space NeRF on the lower-level reference-based (PSNR and LPIPS) and reference-free (DoT) metrics, but performs similarly on the more semantic metrics (DS and DoA). Our RCC metric, designed to detect the "shimmer" present in decoded (neural rendered) videos, detects slightly more inconsistency with ReLS-NeRF. Using R32 reduces accuracy, but enables much faster rendering time (see Table 2).

NeRF	Rendering Time	Fitting Time (A)	(B)	(C)
RGB	132.1s [1×]	–	**1h**	–
Ours-SD	43.1s [3×]	10m	2h	2.5h
Ours-R32	**10.2s [13×]**	40m	1.5h	1.5h

Table 2. Timings of inference (rendering a 120 frames) and fitting for various NeRF types. Simply changing the decoder architecture, D, trades off between efficiency and image quality. We measure the RGB-NeRF rendering time without the latent component.

Colour Consistency (RCC) metric. The RCC measures sudden changes in appearance as Π changes, relying on the NeRF geometry to obtain correspondences. Specifically, we simply reproject one render, I_i, into the reference frame of another, I_{i+1}, using the NeRF depth, D_i, so

$$\text{RCC} = \text{PSNR}\left(\mathbb{E}_i[\text{MSE}(I_{i+1}, \text{Reproj}_{D_i,\Pi_{i+1}} I_i)]\right), \tag{5}$$

where I_i and I_{i+1} are adjacent video frames. Notice that occlusions and view-dependent lighting effects will confound the RCC; however, these effects will (i) be relatively minimal across adjacent frames and (ii) be shared for the same scene, enabling it to be a fair comparative metric.
Video quality. As noted above, adding a temporal dimension can make certain artifacts more perceptually detectable. We therefore applied a recent video quality metric, DOVER [82], to NeRF-rendered videos. DOVER has two components: DOVER-aesthetic (DoA), which focuses on high-level semantics, and DOVER-technical (DoT), which detects low-level distortions (e.g., blur and noise).

4. Discussion

Results. We display our evaluation in Table 1, as well as timing measurements in Table 2, using the eight LLFF

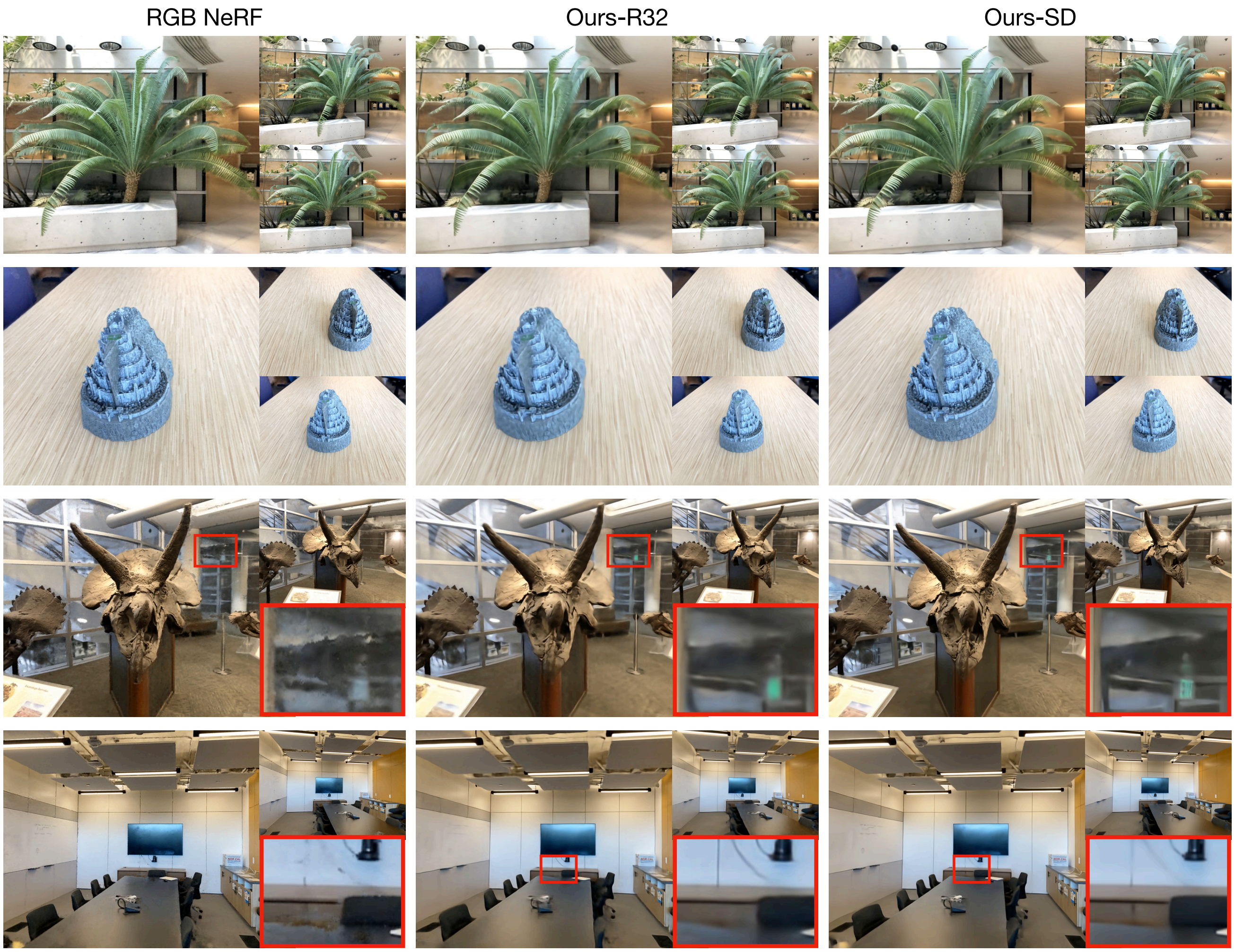

Figure 2. Qualitative comparison of NeRF renders. In the zoomed insets, we show how ReLS-NeRF-SD fixes some of the artifacts of the RGB-NeRF, despite being trained in part on its outputs. One can also see the slight blur incurred by using the faster R32 AE.

scenes [41]*, at 1008×756 resolution. We see that ReLS-NeRF (i.e., decoding a rendered latent feature map) with the SDVAE actually has superior novel view image quality, while having superior inference speed (three times faster). In particular, the low-level metrics, including PSNR, LPIPS, and DoT, prefer ReLS-NeRF-SD over the standard colour NeRF. This is likely due to the fine-tuned decoder fixing artifacts incurred by the colour NeRF, as can be seen in Fig. 2. The higher-level, more semantic metrics are more mixed: DreamSim prefers the RGB-NeRF, while DoA slightly favours ReLS-NeRF-SD. Similarly, the RCC slightly prefers the RGB-NeRF; though it is hard to see in still images, ReLS-NeRF has temporal "jittering" artifacts, which the RCC is designed to detect. We can also control the tradeoff between efficiency and quality by changing the AE architecture: using R32 reduces inference time by $\sim$92%, while decreasing test-view PSNR by only 0.15.

*Images in Fig. 1,2 available in LLFF [41] under a CC BY 3.0 License.

Ablations. We find that removing phase C is devastating to ReLS-NeRF, causing PSNR to drop to 22.85 (SD) and 20.87 (R32). Since the SDVAE is pretrained, ablating phase A has little effect with SD; however, doing so for R32 reduces PSNR by 0.1. Note that the latter case trains the decoder, D, alongside the NeRF and then alone, in phases B and C.

Conclusion. We have shown that ReLS-NeRF can improve image quality, while being several times faster to render. Further, we have demonstrated a tradeoff between efficiency and quality, which can be controlled by the architecture of the AE. Importantly, to obtain its speedup, ReLS-NeRF does not "bake" the scene or transform to a mesh; hence, e.g., it can be continually trained online in the standard fashion. We expect useful future directions to include utilizing different AEs for task-specific biases, applying ReLS-NeRF for online learning, and better customizing the rendering process to latent space rendering (e.g., using a learned mapping instead of volume integration).

References

[1] Michal Adamkiewicz, Timothy Chen, Adam Caccavale, Rachel Gardner, Preston Culbertson, Jeannette Bohg, and Mac Schwager. Vision-Only Robot Navigation in a Neural Radiance World. *IEEE Robotics and Automation Letters*, 2022. 1

[2] Tristan Aumentado-Armstrong, Stavros Tsogkas, Sven Dickinson, and Allan D Jepson. Representing 3D shapes with probabilistic directed distance fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[3] Jonathan T. Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P. Srinivasan. Mip-NeRF: A multiscale representation for anti-aliasing neural radiance fields. *International Conference on Computer Vision (ICCV)*, 2021. 1

[4] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-NeRF 360: Unbounded anti-aliased neural radiance fields. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2, 3

[5] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Zip-NeRF: Anti-aliased grid-based neural radiance fields. *arXiv preprint arXiv:2304.06706*, 2023. 1, 2

[6] Kenneth Blomqvist, Francesco Milano, Jen Jen Chung, Lionel Ott, and Roland Siegwart. Neural implicit vision-language feature fields. *arXiv preprint arXiv:2303.10962*, 2023. 2

[7] Caroline Chan, Frédo Durand, and Phillip Isola. Learning to generate line drawings that convey geometry and semantics. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 3

[8] Eric R Chan, Koki Nagano, Matthew A Chan, Alexander W Bergman, Jeong Joon Park, Axel Levy, Miika Aittala, Shalini De Mello, Tero Karras, and Gordon Wetzstein. Generative novel view synthesis with 3D-aware diffusion models. *arXiv preprint arXiv:2304.02602*, 2023. 2

[9] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. TensoRF: Tensorial radiance fields. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[10] Anpei Chen, Zexiang Xu, Xinyue Wei, Siyu Tang, Hao Su, and Andreas Geiger. Factor fields: A unified framework for neural fields and beyond. *arXiv preprint arXiv:2302.01226*, 2023. 2

[11] Zhiqin Chen, Thomas Funkhouser, Peter Hedman, and Andrea Tagliasacchi. MobileNeRF: Exploiting the polygon rasterization pipeline for efficient neural field rendering on mobile architectures. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2

[12] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (ELUs). *arXiv preprint arXiv:1511.07289*, 2015. 9

[13] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised NeRF: Fewer views and faster training for free. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022. 2

[14] Danny Driess, Ingmar Schubert, Pete Florence, Yunzhu Li, and Marc Toussaint. Reinforcement learning with neural radiance fields. *Neural Information Processing Systems (NeurIPS)*, 2022. 1

[15] S. M. Ali Eslami, Danilo Jimenez Rezende, Frederic Besse, Fabio Viola, Ari S. Morcos, Marta Garnelo, Avraham Ruderman, Andrei A. Rusu, Ivo Danihelka, Karol Gregor, David P. Reichert, Lars Buesing, Theophane Weber, Oriol Vinyals, Dan Rosenbaum, Neil Rabinowitz, Helen King, Chloe Hillier, Matt Botvinick, Daan Wierstra, Koray Kavukcuoglu, and Demis Hassabis. Neural scene representation and rendering. *Science*, 360(6394):1204–1210, 2018. 2

[16] Jiemin Fang, Lingxi Xie, Xinggang Wang, Xiaopeng Zhang, Wenyu Liu, and Qi Tian. Neusample: Neural sample field for efficient view synthesis. *arXiv preprint arXiv:2111.15552*, 2021. 2

[17] Brandon Y Feng, Yinda Zhang, Danhang Tang, Ruofei Du, and Amitabh Varshney. PRIF: Primary ray-based implicit function. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[18] Stephanie Fu*, Netanel Tamir*, Shobhita Sundaram*, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. DreamSim: Learning new dimensions of human visual similarity using synthetic data. *arXiv preprint arXiv:2306.09344*, 2023. 3

[19] Stephan J Garbin, Marek Kowalski, Matthew Johnson, Jamie Shotton, and Julien Valentin. FastNeRF: High-fidelity neural rendering at 200fps. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[20] Jiatao Gu, Lingjie Liu, Peng Wang, and Christian Theobalt. StyleNeRF: A style-based 3D-aware generator for high-resolution image synthesis. *arXiv preprint arXiv:2110.08985*, 2021. 2, 3

[21] Yuan-Chen Guo, Yan-Pei Cao, Chen Wang, Yu He, Ying Shan, Xiaohu Qie, and Song-Hai Zhang. VMesh: Hybrid volume-mesh representation for efficient view synthesis. *arXiv preprint arXiv:2303.16184*, 2023. 2

[22] Ayaan Haque, Matthew Tancik, Alexei Efros, Aleksander Holynski, and Angjoo Kanazawa. Instruct-NeRF2NeRF: Editing 3D scenes with instructions. *arXiv preprint arXiv:2303.12789*, 2023. 1

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 9

[24] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity video generation with arbitrary lengths. *arXiv preprint arXiv:2211.13221*, 2022. 3

[25] Peter Hedman, Pratul P Srinivasan, Ben Mildenhall, Jonathan T Barron, and Paul Debevec. Baking neural radiance fields for real-time view synthesis. In *International Conference on Computer Vision (ICCV)*, 2021. 1, 2

[26] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben

Poole, Mohammad Norouzi, David J Fleet, and Tim Salimans. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 3

[27] Trevor Houchens, Cheng-You Lu, Shivam Duggal, Rao Fu, and Srinath Sridhar. NeuralODF: Learning omnidirectional distance fields for 3D shape representation. *arXiv preprint arXiv:2206.05837*, 2022. 2

[28] Huaibo Huang, Zhihang Li, Ran He, Zhenan Sun, and Tieniu Tan. IntroVAE: Introspective variational autoencoders for photographic image synthesis. *Neural Information Processing Systems (NeurIPS)*, 2018. 2

[29] Muhammad Zubair Irshad, Sergey Zakharov, Rares Ambrus, Thomas Kollar, Zsolt Kira, and Adrien Gaidon. ShAPO: Implicit representations for multi object shape appearance and pose optimization. *European Conference on Computer Vision (ECCV)*, 2022. 1

[30] Mohammad Mahdi Johari, Yann Lepoittevin, and François Fleuret. GeoNeRF: Generalizing nerf with geometry priors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[31] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. LeRF: Language embedded radiance fields. *arXiv preprint arXiv:2303.09553*, 2023. 2

[32] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations (ICLR)*, 2015. 3

[33] Sosuke Kobayashi, Eiichi Matsumoto, and Vincent Sitzmann. Decomposing NeRF for editing via feature field distillation. *Neural Information Processing Systems (NeurIPS)*, 2022. 2

[34] Naruya Kondo, Yuya Ikeda, Andrea Tagliasacchi, Yutaka Matsuo, Yoichi Ochiai, and Shixiang Shane Gu. VaxNeRF: Revisiting the classic for voxel-accelerated neural radiance field. *arXiv preprint arXiv:2111.13112*, 2021. 2

[35] Mikhail Kurenkov, Andrei Potapov, Alena Savinykh, Evgeny Yudin, Evgeny Kruzhkov, Pavel Karpyshev, and Dzmitry Tsetserukou. NFOMP: Neural field for optimal motion planner of differential drive robots with nonholonomic constraints. *IEEE Robotics and Automation Letters*, 2022. 1

[36] Andreas Kurz, Thomas Neff, Zhaoyang Lv, Michael Zollhöfer, and Markus Steinberger. AdaNeRF: Adaptive sampling for real-time rendering of neural radiance fields. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[37] Soomin Lee, Chen Le, Wang Jiahao, Alexander Liniger, Suryansh Kumar, and Fisher Yu. Uncertainty guided policy for active robotic 3D reconstruction using neural radiance fields. *IEEE Robotics and Automation Letters*, 2022. 1

[38] Haotong Lin, Sida Peng, Zhen Xu, Yunzhi Yan, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Efficient neural radiance fields for interactive free-viewpoint video. In *SIGGRAPH Asia 2022 Conference Papers*, 2022. 2

[39] David B Lindell, Julien NP Martel, and Gordon Wetzstein. AutoInt: Automatic integration for fast neural volume rendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[40] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-NeRF for shape-guided generation of 3D shapes and textures. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[41] 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, 4

[42] 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 *European Conference on Computer Vision (ECCV)*, 2020. 1

[43] Ashkan Mirzaei, Tristan Aumentado-Armstrong, Marcus A. Brubaker, Jonathan Kelly, Alex Levinshtein, Konstantinos G. Derpanis, and Igor Gilitschenski. Reference-guided controllable inpainting of neural radiance fields. In *International Conference on Computer Vision (ICCV)*, 2023. 1

[44] Ashkan Mirzaei, Tristan Aumentado-Armstrong, Konstantinos G. Derpanis, Jonathan Kelly, Marcus A. Brubaker, Igor Gilitschenski, and Alex Levinshtein. SPIn-NeRF: Multiview segmentation and perceptual inpainting with neural radiance fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2

[45] Ashkan Mirzaei, Yash Kant, Jonathan Kelly, and Igor Gilitschenski. LaTeRF: Label and text driven object radiance fields. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[46] Thomas Müller. Tiny CUDA neural network framework, 2021. https://github.com/nvlabs/tiny-cuda-nn. 3

[47] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (TOG)*, 2022. 1, 2, 3

[48] Thomas Neff, Pascal Stadlbauer, Mathias Parger, Andreas Kurz, Joerg H Mueller, Chakravarty R Alla Chaitanya, Anton Kaplanyan, and Markus Steinberger. DONeRF: Towards real-time rendering of compact neural radiance fields using depth oracle networks. In *Computer Graphics Forum*, 2021. 2

[49] Ruiqi Ni and Ahmed H. Qureshi. NTFields: Neural time fields for physics-informed robot motion planning. *International Conference on Learning Representations (ICLR)*, 2023. 1

[50] Michael Niemeyer and Andreas Geiger. GIRAFFE: Representing scenes as compositional generative neural feature fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 3

[51] Martin Piala and Ronald Clark. TermiNeRF: Ray termination prediction for efficient neural rendering. In *International Conference on 3D Vision (3DV)*, 2021. 2

[52] Ri-Zhao Qiu, Yixiao Sun, Joao Marcos Correia Marques, and Kris Hauser. Real-time semantic 3D reconstruction for high-touch surface recognition for robotic disinfection. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*, 2022. 1

[53] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, 2021. 3

[54] Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. KiloNeRF: Speeding up neural radiance fields with thousands of tiny MLPs. In *International Conference on Computer Vision (ICCV)*, 2021. 2

[55] Christian Reiser, Richard Szeliski, Dor Verbin, Pratul P. Srinivasan, Ben Mildenhall, Andreas Geiger, Jonathan T. Barron, and Peter Hedman. MERF: Memory-efficient radiance fields for real-time view synthesis in unbounded scenes. In *arXiv preprint arXiv:2302.12249*, 2023. 1, 2

[56] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[57] Sara Fridovich-Keil and Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2

[58] Hoigi Seo, Hayeon Kim, Gwanghyun Kim, and Se Young Chun. DITTO-NeRF: Diffusion-based iterative text to omnidirectional 3D model. *arXiv preprint arXiv:2304.02827*, 2023. 2

[59] Nur Muhammad Mahi Shafiullah, Chris Paxton, Lerrel Pinto, Soumith Chintala, and Arthur Szlam. CLIP-fields: Weakly supervised semantic fields for robotic memory. *arXiv preprint arXiv:2210.05663*, 2022. 2

[60] Dongseok Shim, Seungjae Lee, and H Jin Kim. SNeRL: Semantic-aware neural radiance fields for reinforcement learning. *arXiv preprint arXiv:2301.11520*, 2023. 1

[61] Vincent Sitzmann, Semon Rezchikov, Bill Freeman, Josh Tenenbaum, and Fredo Durand. Light field networks: Neural scene representations with single-evaluation rendering. *Neural Information Processing Systems (NeurIPS)*, 2021. 2

[62] Cameron Omid Smith, Hong-Xing Yu, Sergey Zakharov, Fredo Durand, Joshua B Tenenbaum, Jiajun Wu, and Vincent Sitzmann. Unsupervised discovery and composition of object light fields. *Transactions on Machine Learning Research*, 2023. 2

[63] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[64] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Improved direct voxel grid optimization for radiance fields reconstruction. *arXiv preprint arXiv:2206.05085*, 2022. 2

[65] Andrea Tagliasacchi and Ben Mildenhall. Volume rendering digest (for NeRF). *arXiv preprint arXiv:2209.02417*, 2022. 2

[66] Matthew Tancik, Vincent Casser, Xinchen Yan, Sabeek Pradhan, Ben Mildenhall, Pratul P Srinivasan, Jonathan T Barron, and Henrik Kretzschmar. Block-NeRF: Scalable large scene neural view synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2

[67] Matthew Tancik, Ben Mildenhall, Terrance Wang, Divi Schmidt, Pratul P Srinivasan, Jonathan T Barron, and Ren Ng. Learned initializations for optimizing coordinate-based neural representations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[68] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Justin Kerr, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, David McAllister, and Angjoo Kanazawa. NeRFstudio: A modular framework for neural radiance field development. In *Proceedings of SIGGRAPH*, 2023. 1

[69] Ayush Tewari, Ohad Fried, Justus Thies, Vincent Sitzmann, Stephen Lombardi, Kalyan Sunkavalli, Ricardo Martin-Brualla, Tomas Simon, Jason M. Saragih, Matthias Nießner, Rohit Pandey, Sean Ryan Fanello, Gordon Wetzstein, Jun-Yan Zhu, Christian Theobalt, Maneesh Agrawala, Eli Shechtman, Dan B. Goldman, and Michael Zollhofer. State of the art on neural rendering. *Computer Graphics Forum*, 39(2):701–727, 2020. 2

[70] Ayush Tewari, Justus Thies, Ben Mildenhall, Pratul Srinivasan, Edgar Tretschk, Yifan Wang, Christoph Lassner, Vincent Sitzmann, Ricardo Martin-Brualla, Stephen Lombardi, Tomas Simon, Christian Theobalt, Matthias Niessner, Jonathan T. Barron, Gordon Wetzstein, Michael Zollhoefer, and Vladislav Golyanik. Advances in neural rendering. In *Proceedings of SIGGRAPH*, 2021. 1

[71] Vadim Tschernezki, Iro Laina, Diane Larlus, and Andrea Vedaldi. Neural feature fusion fields: 3D distillation of self-supervised 2D image representations. *arXiv preprint arXiv:2209.03494*, 2022. 2

[72] Haithem Turki, Deva Ramanan, and Mahadev Satyanarayanan. Mega-NeRF: Scalable construction of large-scale nerfs for virtual fly-throughs. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[73] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016. 9

[74] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T. Barron, and Pratul P. Srinivasan. Ref-NeRF: Structured view-dependent appearance for neural radiance fields. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1

[75] Yael Vinker, Yuval Alaluf, Daniel Cohen-Or, and Ariel Shamir. CLIPascene: Scene sketching with different types and levels of abstraction. *arXiv preprint arXiv:2211.17256*, 2022. 3

[76] Krishna Wadhwani and Tamaki Kojima. SqueezeNeRF: Further factorized FastNeRF for memory-efficient inference. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[77] Ziyu Wan, Christian Richardt, Aljaž Božič, Chao Li, Vijay Rengarajan, Seonghyeon Nam, Xiaoyu Xiang, Tuotuo Li, Bo Zhu, Rakesh Ranjan, and Jing Liao. Learning neural duplex radiance fields for real-time view synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[78] Peng Wang, Yuan Liu, Zhaoxi Chen, Lingjie Liu, Ziwei Liu, Taku Komura, Christian Theobalt, and Wenping Wang. F^2-NeRF: Fast neural radiance field training with free camera trajectories. *arXiv preprint arXiv:2303.15951*, 2023. 2

[79] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. IBRNet: Learning multi-view image-based rendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[80] Frederik Warburg, Ethan Weber, Matthew Tancik, Aleksander Holynski, and Angjoo Kanazawa. Nerfbusters: Removing ghostly artifacts from casually captured NeRFs. *arXiv preprint arXiv:2304.10532*, 2023. 1

[81] Bowen Wen, Jonathan Tremblay, Valts Blukis, Stephen Tyree, Thomas Muller, Alex Evans, Dieter Fox, Jan Kautz, and Stan Birchfield. BundleSDF: Neural 6-DoF tracking and 3D reconstruction of unknown objects. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1

[82] Haoning Wu, Erli Zhang, Liang Liao, Chaofeng Chen, Jingwen Hou Hou, Annan Wang, Wenxiu Sun Sun, Qiong Yan, and Weisi Lin. Exploring video quality assessment on user generated contents from aesthetic and technical perspectives. In *International Conference on Computer Vision (ICCV)*, 2023. 3

[83] Liwen Wu, Jae Yong Lee, Anand Bhattad, Yu-Xiong Wang, and David Forsyth. Diver: Real-time and accurate neural radiance fields with deterministic integration for volume rendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[84] Yang Xue, Yuheng Li, Krishna Kumar Singh, and Yong Jae Lee. GIRAFFE-HD: A high-resolution 3D-aware generative model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[85] Tarun Yenamandra, Ayush Tewari, Nan Yang, Florian Bernard, Christian Theobalt, and Daniel Cremers. FIRe: Fast inverse rendering using directional and signed distance functions. *arXiv preprint arXiv:2203.16284*, 2022. 2

[86] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. Plenoctrees for real-time rendering of neural radiance fields. In *International Conference on Computer Vision (ICCV)*, 2021. 2

[87] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. PixelNeRF: Neural radiance fields from one or few images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[88] Kai Zhang, Nick Kolkin, Sai Bi, Fujun Luan, Zexiang Xu, Eli Shechtman, and Noah Snavely. ARF: Artistic radiance fields. In *European Conference on Computer Vision (ECCV)*, 2022. 1

[89] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. NeRF++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020. 1

[90] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[91] Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew J Davison. In-place scene labelling and understanding with implicit scene representation. In *International Conference on Computer Vision (ICCV)*, 2021. 2

[92] Shuaifeng Zhi, Edgar Sucar, Andre Mouton, Iain Haughton, Tristan Laidlow, and Andrew J Davison. iLabel: Interactive neural scene labelling. *arXiv preprint arXiv:2111.14637*, 2021. 2

[93] Shaohong Zhong, Alessandro Albini, Oiwi Parker Jones, Perla Maiolino, and Ingmar Posner. Touching a NeRF: Leveraging neural radiance fields for tactile sensory data generation. In *Conference on Robot Learning*, 2023. 1