DeVRF: Fast Deformable Voxel Radiance Fields for Dynamic Scenes
Supplementary Material

Jia-Wei Liu[1]*, **Yan-Pei Cao**[2], **Weijia Mao**[1], **Wenqiao Zhang**[4], **David Junhao Zhang**[1], **Jussi Keppo**[5,6], **Ying Shan**[2], **Xiaohu Qie**[3], **Mike Zheng Shou**[1]†

[1] Show Lab, National University of Singapore [2] ARC Lab, [3] Tencent PCG
[4] National University of Singapore [5] Business School, National University of Singapore
[6] Institute of Operations Research and Analytics, National University of Singapore

This supplementary material is organized as follows:

- Section A provides more implementation details of the proposed DeVRF.
- Section B presents additional results of the per-scene evaluation as well as evaluations on an additional forward-facing real-world deformable scene.
- Section C conducts additional ablations to further verify the effectiveness of our low-cost data capture process and the proposed DeVRF model.

In addition to this supplementary material, it is worth noting that we also provide a **supplementary video** to better visualize and compare our results to other SOTA approaches on all synthetic and real-world deformable scenes.

A Implementation Details

We use the PyTorch [6] deep learning framework to conduct all our experiments on a single NVIDIA GeForce RTX3090 GPU.

3D canonical space optimization. During training, we set the voxel resolution of 3D canonical space, *i.e.*, density grid $\mathbf{V}_{\text{density}}$ and color grid $\mathbf{V}_{\text{color}}$, to $160 \times 160 \times 160$ for inward-facing scenes and $256 \times 256 \times 128$ for forward-facing scenes, and we use a shallow MLP with 2 hidden layers (128 channels for inward-facing scenes, and 64 channels for forward-facing scenes). The 3D canonical space is optimized using a standard Adam optimizer [1] for 20k with a batch size of 8192 rays for inward-facing scenes and 4096 rays for forward-facing scenes. The learning rate of $\mathbf{V}_{\text{density}}$, $\mathbf{V}_{\text{color}}$, and the designed MLP are set to 10^{-1}, 10^{-1}, and 10^{-3}, respectively.

4D voxel deformation field optimization. The dense 4D voxel deformation field is modeled in $N_t \times C \times N_x \times N_y \times N_z$ resolution, which corresponds to $50 \times 3 \times 160 \times 160 \times 160$. In our proposed coarse-to-fine optimization, we progressively upscale the $(x\text{-}y\text{-}z)$ resolution of the 4D voxel deformation field $\mathbf{V}_{\text{motion}}$ as $(10 \times 10 \times 10) \rightarrow (20 \times 20 \times 20) \rightarrow (40 \times 40 \times 40) \rightarrow (80 \times 80 \times 80)$ $\rightarrow (160 \times 160 \times 160)$. Such an optimization strategy can estimate the fine-grained voxels motion from a cascaded learning sequence. The base learning rate of the 4D voxel deformation field is 10^{-3}, which is progressively decayed to 10^{-4} during coarse-to-fine optimization. For loss weights, we set $\omega_{\text{Render}} = 1$, $\omega_{\text{Cycle}} = 100$, $\omega_{\text{Flow}} = 0.005$, and $\omega_{\text{TV}} = 1$ across all scenes. The 4D voxel deformation field is optimized using Adam optimizer [1] for 25k iterations with a batch size of 8192 rays.

---

*Work is partially done during internship at ARC Lab, Tencent PCG.

†Corresponding Author.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

B Additional Results

B.1 Per-scene Evaluation on Inward-facing Synthetic Deformable Scenes.

For quantitative comparison, Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) [9], and Learned Perceptual Image Patch Similarity (LPIPS) [10] with VGG [8] are employed as evaluation metrics. PSNR and SSIM are simple and shallow functions, while LPIPS measures the perceptual similarity of deep visual representations and is more representative of visual quality. We report the per-scene comparisons on five inward-facing synthetic dynamic scenes - Lego, Floating robot, Daisy, Glove, Kuka - in Tab. 1 and Tab. 2. DeVRF achieves the best performance in terms of LPIPS in five scenes, and almost the second- or third-best in terms of PSNR and SSIM among all approaches. For the floating robot, daisy, and kuka, DeVRF achieves the best performance in terms of both the PSNR and LPIPS. Most importantly, our per-scene optimization only takes less than $10\mathrm{mins}$ with less than $5.0\mathrm{GB}$ GPU memory on a single NVIDIA GeForce RTX3090 GPU, which is about two orders of magnitude faster than other approaches.

Table 1: Per-scene quantitative evaluation on inward-facing synthetic scenes (Lego, Floating robot, and Daisy) against baselines and ablations of our method. We color code each cell as **best**, **second best**, and **third best**.

	Lego					FLOATING ROBOT					DAISY				
	PSNR↑	SSIM↑	LPIPS↓	GPU (GB)↓	Time↓	PSNR↑	SSIM↑	LPIPS↓	GPU (GB)↓	Time↓	PSNR↑	SSIM↑	LPIPS↓	GPU (GB)↓	Time↓
Neural Volumes [3]	5.958	0.369	0.8314	19.4	22.4hrs	6.403	0.405	0.7127	19.4	22.4hrs	13.47	0.679	0.4429	19.4	22.4hrs
D-NeRF [7]	28.41	0.935	0.0582	10.3	18.8hrs	31.98	0.978	0.0251	10.3	18.4hrs	33.51	0.990	0.0137	9.8	17.9hrs
D-NeRF [7]-2 stage	24.34	0.885	0.1020	9.7	18.5hrs	28.79	0.973	0.0289	9.7	18.5hrs	31.40	0.985	0.0225	9.7	17.8hrs
D-NeRF [7]-dynamic	20.29	0.852	0.1360	10.0	22.0hrs	14.22	0.821	0.2720	10.0	21.6hrs	22.76	0.947	0.0873	9.5	21.7hrs
Nerfies [4]	30.34	0.986	0.0303	22.5	19.3hrs	27.07	0.973	0.0773	22.5	18.4hrs	38.26	0.998	0.0056	22.5	18.5hrs
Nerfies [4]-2 stage	29.27	0.984	0.0449	22.4	15.8hrs	30.05	0.991	0.0286	22.4	15.8hrs	35.81	0.997	0.0100	22.4	15.8hrs
Nerfies [4]-dynamic	21.15	0.911	0.1410	22.5	18.7hrs	19.84	0.859	0.1400	22.5	19.3hrs	23.71	0.864	0.1450	22.4	19.4hrs
HyperNeRF [5]	30.99	0.963	0.0360	22.5	21.3hrs	33.15	0.930	0.0511	22.5	19.8hrs	36.31	0.994	0.0080	22.5	20.8hrs
HyperNeRF [5]-2stage	27.28	0.933	0.0758	22.5	21.3hrs	28.28	0.955	0.0554	22.3	18.9hrs	31.63	0.983	0.0238	22.3	18.5hrs
HyperNeRF [5]-dynamic	14.41	0.774	0.2910	22.4	19.8hrs	14.88	0.835	0.2700	22.4	21.0hrs	22.73	0.946	0.0957	22.4	21.0hrs
NSFF [2]	25.44	0.912	0.0941	23.6	12.7hrs	25.27	0.935	0.0944	20.9	13.2hrs	28.71	0.967	0.0493	20.9	12.8hrs
NSFF [2]-dynamic	15.14	0.762	0.2732	12.5	14.8hrs	16.66	0.878	0.1769	15.7	16.0hrs	24.02	0.937	0.1069	15.7	16.0hrs
Ours (base)	17.83	0.799	0.2060	4.1	8mins	21.74	0.907	0.1040	4.5	8mins	25.91	0.950	0.0637	4.2	7mins
Ours w/ c2f	27.55	0.952	0.0342	4.1	7mins	32.04	0.983	0.0108	4.5	7mins	37.89	0.996	0.0055	4.2	6mins
Ours w/ c2f, tv	28.44	0.958	0.0301	4.1	8mins	32.78	0.985	0.0101	4.5	7mins	38.55	0.950	0.0048	4.2	6mins
Ours w/ c2f, tv, cycle	29.11	0.963	0.0254	4.1	8mins	34.12	0.988	0.0084	4.5	8mins	39.00	0.996	0.0044	4.2	7mins
Ours w/ c2f, tv, cycle, flow	29.25	0.964	0.0250	4.1	9mins	35.20	0.989	0.0074	4.5	9mins	38.39	0.996	0.0046	4.2	7mins

Table 2: Per-scene quantitative evaluation on inward-facing synthetic scenes (Glove and Kuka) against baselines and ablations of our method. We color code each cell as **best**, **second best**, and **third best**.

	GLOVE					KUKA				
	PSNR↑	SSIM↑	LPIPS↓	GPU(GB)↓	Time↓	PSNR↑	SSIM↑	LPIPS↓	GPU(GB)↓	Time↓
Neural Volumes [3]	6.371	0.449	0.6101	19.4	22.4hrs	15.92	0.757	0.1645	19.4	22.4hrs
D-NeRF [7]	34.24	0.927	0.0455	9.8	18.5hrs	31.03	0.975	0.0349	9.8	18.4hrs
D-NeRF [7]-2 stage	30.86	0.922	0.0494	9.7	18.3hrs	26.05	0.959	0.0614	9.6	18.8hrs
D-NeRF [7]-dynamic	15.71	0.801	0.2660	10.3	22.1hrs	14.97	0.777	0.2679	9.5	22.4hrs
Nerfies [4]	36.37	0.993	0.0328	18.9	18.9hrs	33.40	0.996	0.0193	22.5	18.5hrs
Nerfies [4]-2 stage	34.96	0.991	0.0549	22.4	15.9hrs	31.79	0.994	0.0227	22.4	15.8hrs
Nerfies [4]-dynamic	15.67	0.636	0.1740	22.5	18.5hrs	16.86	0.698	0.2372	22.5	19.3hrs
HyperNeRF [5]	35.33	0.956	0.0471	22.5	20.0hrs	32.88	0.983	0.0255	22.5	20.8hrs
HyperNeRF [5]-2stage	31.47	0.935	0.0694	22.3	18.8hrs	27.12	0.959	0.0532	22.3	18.6hrs
HyperNeRF [5]-dynamic	22.04	0.850	0.1870	22.4	20.6hrs	15.97	0.856	0.2429	22.4	20.8hrs
NSFF [2]	27.66	0.902	0.1050	20.7	12.7hrs	28.24	0.962	0.0574	20.9	12.7hrs
NSFF [2]-dynamic	16.51	0.846	0.2091	15.7	16.0hrs	18.59	0.865	0.1986	15.7	14.8hrs
Ours (base)	25.14	0.867	0.1080	4.6	7mins	21.59	0.914	0.1050	4.5	8mins
Ours w/ c2f	30.69	0.959	0.0274	4.6	7mins	31.70	0.985	0.0147	4.5	7mins
Ours w/ c2f, tv	31.38	0.963	0.0234	4.6	8mins	32.48	0.987	0.0176	4.5	8mins
Ours w/ c2f, tv, cycle	33.67	0.970	0.0183	4.6	8mins	33.95	0.989	0.0147	4.5	8mins
Ours w/ c2f, tv, cycle, flow	34.67	0.973	0.0168	4.6	8mins	33.96	0.989	0.0147	4.5	9mins

B.2 Per-scene Video Comparisons on Synthetic and Real-world Deformable Scenes.

We also provide a supplementary video to better visualize and compare our results to SOTA approaches on all five synthetic and three real-world deformable scenes. As can be seen from the video, our DeVRF achieves on-par high-fidelity dynamic novel view synthesis results on all scenes and synthesizes the cleanest depth maps compared to other approaches.

Table 3: Additional quantitative evaluation on a forward-facing real-world scene against baselines and ablations of our system. We color code each cell as **best**, **second best**, and **third best**.

	Pig Toy				
	PSNR↑	SSIM↑	LPIPS↓	GPU (GB)↓	Time↓
D-NeRF [7]	31.23	0.974	0.0381	12.4	22.1hrs
Nerfies [4]	30.66	0.986	0.0487	22.0	20.5hrs
HyperNeRF [5]	25.92	0.942	0.1037	22.0	22.3hrs
NSFF [2]	28.39	0.957	0.0654	20.5	14.5hrs
Ours (base)	22.84	0.942	0.0739	10.8	8mins
Ours w/ c2f	21.11	0.956	0.0595	10.8	8mins
Ours w/ c2f, tv	21.62	0.957	0.0578	10.8	8mins
Ours w/ c2f, tv, cycle	30.56	0.967	0.0447	10.8	9mins
Ours w/ c2f, tv, cycle, flow	30.85	0.967	0.0447	10.8	10mins

Significant quality enhancements of our DeVRF can be observed in the video examples for floating robot, kuka, flower-360°, plant, and rabbit. Notably, clear differences can be observed in the plant and rabbit scenes, where D-NeRF [7] and NSFF [2] generate intermittent motions. In contrast, the quadruple interpolation of the 4D voxel deformation field in our DeVRF allows us to synthesize smooth motions at novel time steps.

In addition, the quadruple interpolation of the 4D voxel deformation field in DeVRF allows us to conveniently and efficiently synthesize novel views at novel time steps, while existing approaches (i.e., Nerfies [4] and HyperNeRF [5]) cannot synthesize the views at novel time steps that have not been seen during model training [4, 5]. Thus, when rendering video examples, we generate results for Nerfies [4] and HyperNeRF [5] only on the training and testing time steps. This makes their videos' duration shorter than ours.

B.3 Additional Forward-facing Real-world Deformable Scene Evaluation.

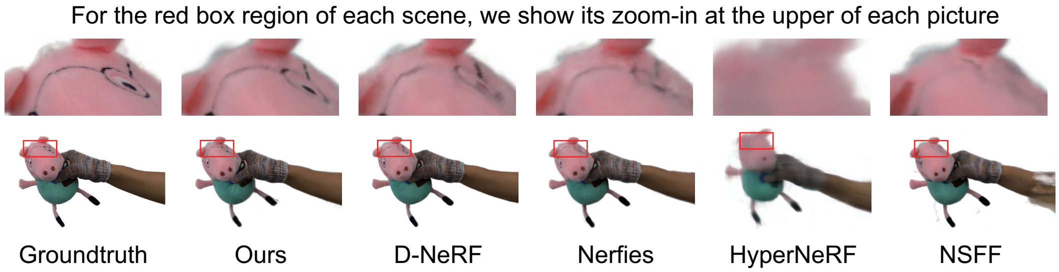
For the red box region of each scene, we show its zoom-in at the upper of each picture
Groundtruth
Ours
D-NeRF
Nerfies
HyperNeRF
NSFF

Figure 1: Qualitative comparisons of baselines and DeVRF on the additional real-world scene.

We additionally collected another forward-facing real-world deformable scene where a human rotated and squeezed a toy pig in 540 × 960 pixels using 4 cameras, and we chose 3 views of them as training data and the other view as test data. As shown in Tab. 3, DeVRF achieves two orders of magnitude speedup compared to other approaches, and the second-best result in terms of PSNR and LPIPS metrics and the third-best result in terms of SSIM metric. Fig. 1 visualizes qualitative comparisons of DeVRF and baselines on this scene.

C Additional Ablations

To further evaluate the influence of the number of dynamic training views, we conduct additional ablations for DeVRF on four real-world dynamic scenes and report the per-scene metrics as well as the average metrics with respect to the number of dynamic training views. As shown in Fig. 2, given the same multi-view static images, the performance of DeVRF largely improves with the increment of dynamic training views and achieves the best performance at three dynamic views. We additionally visualize the qualitative results of DeVRF with different numbers of dynamic training views in Fig. 3. Therefore, in our *static* → *dynamic* learning paradigm, with static multi-view data as a supplement, only a few dynamic views are required to significantly boost the performance of dynamic neural radiance fields reconstruction. This further demonstrates the effectiveness of our low-cost data capture process and the DeVRF model.

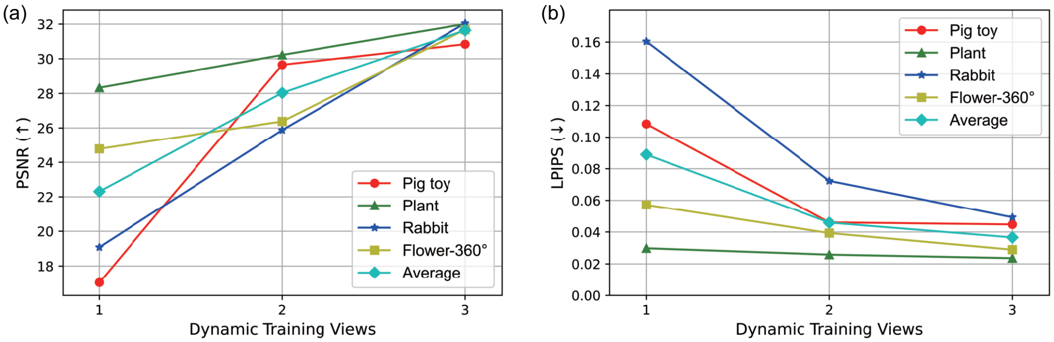

Figure 2: Ablation evaluation on the number of dynamic training views on real-world dataset: (a) PSNR, (b) LPIPS.

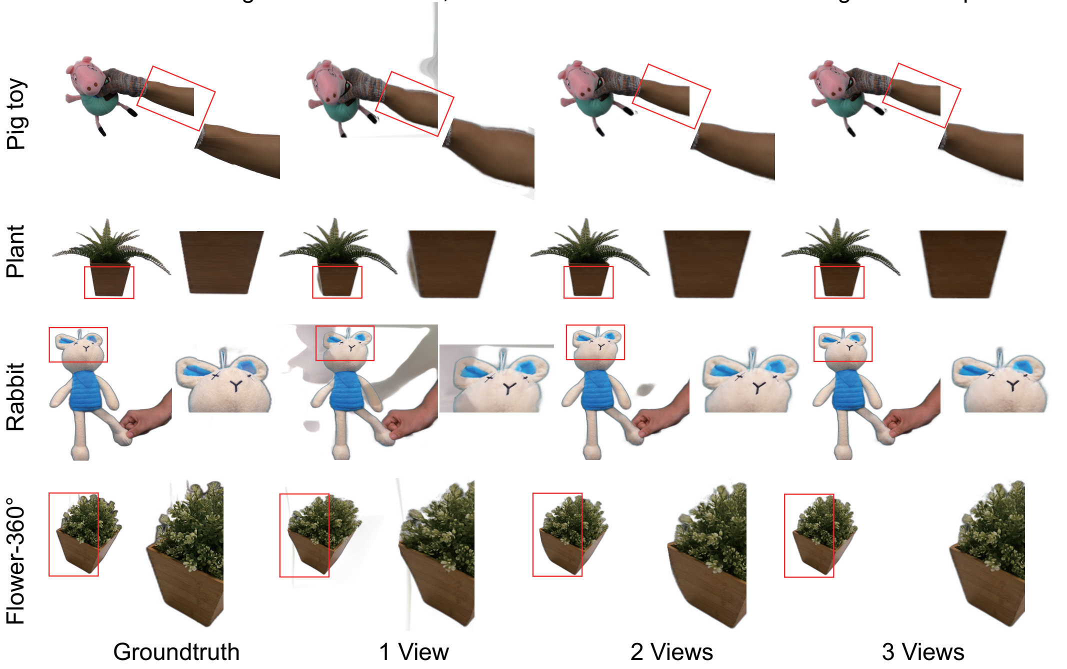
For the red box region of each scene, we show its zoom-in at the bottom-right of each picture

Figure 3: Qualitative results of DeVRF with different numbers of dynamic training views on real-world dataset.

References

[1] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[2] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6498–6508, 2021.

[3] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. Neural volumes: Learning dynamic renderable volumes from images. *arXiv preprint arXiv:1906.07751*, 2019.

[4] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021.

[5] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *arXiv preprint arXiv:2106.13228*, 2021.

[6] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[7] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021.

[8] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[9] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[10] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.