

**Total-Recon:  
Deformable Scene Reconstruction for  
Embodied View Synthesis**

Chonghyuk Song  
CMU-RI-TR-23-13  
May 09



The Robotics Institute  
School of Computer Science  
Carnegie Mellon University  
Pittsburgh, PA

**Thesis Committee:**

Professor Deva K. Ramanan, *Co-chair*  
Professor Jun-Yan Zhu, *Co-chair*  
Professor Shubham Tulsiani  
Jason Zhang, *Carnegie Mellon University*

*Submitted in partial fulfillment of the requirements  
for the degree of Master of Science in Robotics.*

Copyright © 2023 Chonghyuk Song. All rights reserved.



*To my parents and my family for their continued love and support*



## Abstract

We explore the task of embodied view synthesis from monocular videos of deformable scenes. Given a minute-long RGBD video of people interacting with their pets, we render the scene from novel camera trajectories derived from in-scene motion of actors: (1) egocentric cameras that simulate the point of view of a target actor and (2) 3rd-person cameras that follow the actor. Building such a system requires reconstructing the root-body and articulated motion of each actor in the scene, as well as a scene representation that supports free-viewpoint synthesis. Longer videos are more likely to capture the scene from diverse viewpoints (which helps reconstruction) but are also more likely to contain larger motions (which complicates reconstruction). To address these challenges, we present Total-Recon, the first method to photorealistically reconstruct deformable scenes from long monocular RGBD videos. Crucially, to scale to long videos, our method hierarchically decomposes the scene motion into the motion of each object, which itself is decomposed into global root-body motion and local articulations. To quantify such “in-the-wild” reconstruction and view synthesis, we collect ground-truth data from a specialized stereo RGBD capture rig for 11 challenging videos, significantly outperforming prior art.



## Acknowledgments

They say that time moves more quickly as one ages, but I am still in awe of how quickly my time as a master’s student at Carnegie Mellon University (CMU) has gone by. So much has happened since the day I first stepped foot into Pittsburgh. I have undergone immense personal growth as a researcher and as a person persevering through life. I have made many friends from within and outside academia that I am sure I will carry with me for years to come. And I have many people to thank for that.

Let me start with my wonderful advisors: Jun-Yan Zhu and Deva Ramanan. If someone had told me two years ago that I would be working with Professor Jun-Yan Zhu and Professor Deva Ramanan, I would have had a very hard time believing them. Even now, it feels surreal whenever I find myself in a meeting with Jun-Yan and Deva, talking about how to respond to the latest round of CVPR reviews or just gushing over the latest cool paper on arXiv. Jun-Yan has frequently pushed me outside my comfort zone and encouraged me to set high standards for everything I do in research, from finding the killer application for marketing my research to more fine-grained habits such as automating my training scripts. Deva has ingrained in me the importance of crisp and concise communication, whether for verbal interactions during our meetings or writing the perfect caption for a table in a paper. It’s still a work in progress, but at least now I frequently find myself asking, “Is there a way to deliver the same message with fewer words?”. Not only that, despite their insanely busy schedules, Jun-Yan and Deva have always found the time to meet with me whenever I request for facetime - to brainstorm research directions or simply to complain about how overwhelming coursework has been lately. I am extremely grateful for their guidance and patience along this journey. They have truly shaped me into the researcher and person I am today. I am also thankful to Professor Shubham Tulsiani for agreeing to be on my master’s thesis committee and regularly checking in with me on the progress of my research over the past year.

I would also like to thank my brilliant collaborators on my master’s project: Gengshan and Kangle. I cannot begin to recount the occasions when Gengshan answered my stupid questions over Slack or magically debugged a piece of code that I had been stuck on for days. Having Gengshan as a collaborator was like having a third advisor. Kangle also generously offered his help whenever I asked for it. Before Gengshan joined as a

collaborator on my research project and when I was having difficulty deciding on a research direction, Kangle frequently assumed the role of therapist, listening to my problems and offering strategic advice on how to proceed with my research. I cannot overemphasize how important their support has been, not only just to our research but also to my mental health and sanity.

Finally, I would like to thank other members (including past members) of the Generative Intelligence Lab - Sheng-Yu, Ruihan, Nupur, Gaurav, George, Songwei, Or, Muiyang, Fred, Ani, Rohan, Maxwell, Mia, Bingliang - and Deva's Lab - Jonathon, Aljosa, Martin, Jason, Haithem, Zhiqiu, Neehar, Gautam, Tarasha, Andrew (Saba), Andrew (Shen), Anish, Xindi, Jeff - for being fantastic lab mates and friends.

## Funding

I am grateful for the financial support provided by the Sony Corporation and the CMU Argo AI Center for Autonomous Vehicle Research during my master's program.



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Challenges . . . . .	1
1.2	Key Ideas . . . . .	2
1.3	Evaluation . . . . .	3
1.4	Contributions . . . . .	3
<b>2</b>	<b>Related Works</b>	<b>4</b>
2.1	Neural Radiance Fields . . . . .	4
2.2	Deformable NeRFs . . . . .	4
2.3	Reconstruction with RGBD Sensors . . . . .	6
2.4	Concurrent Works . . . . .	6
<b>3</b>	<b>Method</b>	<b>7</b>
3.1	Limitations of Prior Art . . . . .	7
3.2	Component Radiance Fields . . . . .	8
3.3	Composite Rendering of Multiple Objects . . . . .	10
3.3.1	Rendering Flow, Depth, and Silhouettes . . . . .	11
3.3.2	Optimization . . . . .	12
3.4	Embodied View Synthesis and 3D Filters . . . . .	12
<b>4</b>	<b>Experiments</b>	<b>14</b>
4.1	Implementation Details . . . . .	14
4.2	Dataset . . . . .	15
4.3	Reconstruction and Applications . . . . .	15
4.4	Baselines and Evaluation . . . . .	16
4.4.1	Comparisons . . . . .	19
4.5	Ablation Studies . . . . .	19
4.5.1	Depth Supervision . . . . .	20
4.5.2	Motion Modeling . . . . .	21
<b>5</b>	<b>Discussion and Limitations</b>	<b>23</b>
5.1	Limitations. . . . .	23
<b>A</b>	<b>Appendix Overview</b>	<b>25</b>

<b>B</b>	<b>Implementation Details</b>	<b>26</b>
B.1	Data Preprocessing. . . . .	26
B.2	Optimization. . . . .	26
B.3	Pretraining. . . . .	27
B.4	Joint Finetuning. . . . .	28
<b>C</b>	<b>Dataset Details</b>	<b>29</b>
<b>D</b>	<b>Details of Baseline Comparisons</b>	<b>31</b>
D.1	Baseline Experiment Details. . . . .	31
D.2	Additional Quantitative Metrics. . . . .	32
D.3	Additional Qualitative Results. . . . .	32
<b>E</b>	<b>Reconstruction and Applications</b>	<b>36</b>
E.1	Geometry and Embodied View Synthesis. . . . .	36
E.2	Object Removal. . . . .	36
<b>F</b>	<b>Additional Ablation Studies</b>	<b>39</b>
F.1	Ablation Study on Depth Supervision . . . . .	39
F.2	Ablation Study on Motion Modeling . . . . .	41
F.3	Ablation Study on Joint-Finetuning . . . . .	46
<b>G</b>	<b>Societal Impact</b>	<b>48</b>
	<b>Bibliography</b>	<b>49</b>

# List of Figures

1.1	<b>Embodied View Synthesis.</b> Given a long video of deformable objects captured by a handheld RGBD sensor, Total-Recon renders the scene from novel camera trajectories derived from <i>in-scene</i> motion of actors: (1) egocentric cameras that simulate the <i>point-of-view of a target actor</i> (such as the pet) and (2) 3rd-person (or pet) cameras that follow the actor from behind. Our method also enables (3) 3D video filters that attach virtual 3D assets to the actor. Total-Recon achieves this by reconstructing the geometry, appearance, root-body- and articulated motion of each deformable object in the scene as well as the background. <a href="#">[Videos]</a> . . . . .	2
3.1	<b>Method Overview.</b> Total-Recon represents the entire scene as a composition of $M$ object-centric neural fields, one for the rigid background and for each of the $M - 1$ deformable objects. To render a scene, (1) <i>each object field <math>j</math></i> is transformed into the camera space with a rigid transformation $(\mathbf{G}_j^t)^{-1}$ that encodes root-body motion and, for each deformable object, an additional deformation field $\mathbf{J}_j^{t,\rightarrow}$ that encodes articulated motion. Next, all (2) <i>posed object fields</i> are combined into a (3) <i>composite field</i> , which is then volume-rendered into (4) <i>color, depth, optical flow, and object silhouettes</i> . Each rendered output defines a reconstruction loss that derives supervision from a monocular RGBD video captured by a moving iPad Pro. . . . .	8
4.1	<b>6-DOF Trajectories for Embodied View Synthesis.</b> To synthesize embodied views from egocentric or actor-following cameras, Total-Recon reconstructs the entire background, every individual actor in the scene, as well as global 6-DOF trajectories of its root-body and its articulated body parts (e.g. head). <a href="#">[Videos]</a> . . . . .	16

4.2	<b>Embodied View Synthesis and 3D Filters.</b> For select sequences of our RGBD dataset, we visualize the scene geometry and appearance reconstructed by our method (3D reconstruction) and the resulting downstream applications. The yellow and blue camera meshes in the mesh renderings represent the egocentric and 3rd-person-follow cameras, respectively. To showcase the 3D video filter we attach a sky-blue unicorn horn to the forehead of the target object, which is then automatically propagated across all frames. <a href="#">[Videos]</a> . . . . .	17
4.3	<b>Baseline Comparisons.</b> We compare Total-Recon to depth-supervised variants of HyperNeRF [31] and D <sup>2</sup> NeRF [52] on the task of stereo-view synthesis (the left camera is used for training and the right is used for testing). While the baselines are able to reconstruct only the background at best, Total-Recon can reconstruct <i>both</i> the background and the moving deformable object(s), demonstrating holistic scene reconstruction. <a href="#">[Videos]</a> . . . . .	18
4.4	<b>Ablation Study.</b> We visualize results from the ablations in Table 4.2. Removing depth supervision causes the cat to sink into the ground due to inconsistent object scales. Removing the deformation field results in coarse object reconstructions that fail to capture articulated body parts such as moving limbs. We do not visualize Total-Recon without root-body poses as this ablation does not converge. <a href="#">[Videos]</a> . . . . .	20
4.5	<b>Ablation Study on Depth Supervision.</b> While removing depth supervision does not significantly deteriorate training-view RGB renderings, it significantly hurts <i>novel-view</i> 3D reconstructions, as characterized by the following: (a) floating foreground objects (as evidenced by their shadows) and (b) the human incorrectly occluding the dog. These failure modes indicate that without depth supervision, Total-Recon overfits the training view, and the reconstructed objects fail to converge to the same scale. <a href="#">[Videos]</a> . . . . .	21
C.1	<b>Stereo Validation Rig Used Only for Evaluation.</b> To enable quantitative evaluation, we built a stereo rig comprised of two iPad Pros rigidly attached to a camera mount, and captured 11 pairs of RGBD sequences. <b>We train each method <i>only</i> on the sequences captured from the left camera</b> and evaluate the images rendered from the viewpoint of the right camera. . . . .	30

D.1	<b>Qualitative Comparisons on Novel View Synthesis (Additional Sequences).</b> We compare Total-Recon to depth-supervised variants of HyperNeRF [31] and D <sup>2</sup> NeRF [52] on the task of stereo-view synthesis (the left camera is used for training and the images are rendered to the right camera). While the baselines are only able to reconstruct the background at best, Total-Recon is able to reconstruct <i>both</i> the background and the moving deformable object(s), demonstrating holistic scene reconstruction. <a href="#">[Videos]</a> . . . . .	34
D.1	<b>[Continued] Qualitative Comparisons on Novel View Synthesis (Additional Sequences).</b> We compare Total-Recon to depth-supervised variants of HyperNeRF [31] and D <sup>2</sup> NeRF [52] on the task of stereo-view synthesis (the left camera is used for training and the images are rendered to the right camera). While the baselines are only able to reconstruct the background at best, Total-Recon is able to reconstruct <i>both</i> the background and the moving deformable object(s), demonstrating holistic scene reconstruction. <a href="#">[Videos]</a> . . . . .	35
E.1	<b>Embodied View Synthesis and 3D Filters (Additional Sequences).</b> We visualize the 3D reconstructions (rendered from the novel view) and the applications enabled by Total-Recon for the remaining 5 sequences of our RGBD dataset that were not shown in the main paper. The yellow and blue camera meshes in the mesh renderings represent the egocentric and 3rd-person-follow cameras, respectively. To showcase the 3D filter, we attach a sky-blue unicorn horn to the forehead of the foreground object, which is automatically propagated across all frames. <a href="#">[Videos]</a> . . . . .	37
E.2	<b>Object Removal.</b> Our compositional scene representation enables object removal. We remove the HUMAN and then the PET object from our composite rendering process (Chapter 3.3) and display the resulting renderings. <a href="#">[Videos]</a> . . . . .	38

**F.1 Ablation Study on Depth Supervision.** While removing depth supervision from Total-Recon (COMPLETE MODEL) doesn't significantly hamper the training-view RGB renderings, it induces the following failure modes in the *novel-view* 3D reconstructions. (a) *Floating objects*: for the HUMAN 1 & DOG 1, DOG 1, HUMAN 1, and CAT 2 sequences, the foreground objects float above the ground, as evidenced by their shadows. (b) *Objects that sink into the background*: for the HUMAN 2 & CAT 1 sequence, the reconstructed cat is halfway sunk into the ground. (c) *Incorrect occlusions*: for the HUMAN 1 & DOG 1 sequence, the human is incorrectly occluding the dog. (d) *Lower reconstruction quality*: for the HUMAN 2 & CAT 1 sequence, the cat displays lower reconstruction quality, and, for all sequences except HUMAN 1 & DOG 1 and CAT 2, the background object exhibits lower reconstruction quality. [\[Videos\]](#) . . . . . 40

**F.2 Ablation Study on Motion Modeling.** We visualize the novel-view renderings of the ablations in Table F.2. Ablating camera-pose optimization (w/o cam. opt.) does not qualitatively change the scene reconstruction. Changing the deformation field from Total-Recon's neural blend skinning (NBS) function to an SE(3)-field (w/ SE(3)-field) results in minor artifacts in the foreground reconstruction. Removing the deformation field entirely (w/o deform. field) produces coarse object reconstructions that fail to model moving body parts such as limbs. Removing object root-body poses (w/o root-body (SE3)) breaks foreground reconstruction entirely, as evidenced by the ghosting artifacts indicative of failed foreground reconstruction. These experiments justify our method's hierarchical motion representation, where object motion is decomposed into global root-body motion and local articulations. [\[Videos\]](#) . . . . . 43

F.2	<p><b>[Continued] Ablation Study on Motion Modeling.</b> We visualize the novel-view renderings of the ablations in Table F.2. Ablating camera-pose optimization (w/o cam. opt.) does not qualitatively change the scene reconstruction. Changing the deformation field from Total-Recon’s neural blend skinning (NBS) function to an SE(3)-field (w/ SE(3)-field) results in minor artifacts in the foreground reconstruction. Removing the deformation field entirely (w/o deform. field) produces coarse object reconstructions that fail to model moving body parts such as limbs. Removing object root-body poses (w/o root-body (SE3)) breaks foreground reconstruction entirely, as evidenced by the ghosting artifacts indicative of failed foreground reconstruction. These experiments justify our method’s hierarchical motion representation, where object motion is decomposed into global root-body motion and local articulations. <a href="#">[Videos]</a> . . . . .</p>	44
F.2	<p><b>[Continued] Ablation Study on Motion Modeling.</b> We visualize the novel-view renderings of the ablations in Table F.2. Ablating camera-pose optimization (w/o cam. opt.) does not qualitatively change the scene reconstruction. Changing the deformation field from Total-Recon’s neural blend skinning (NBS) function to an SE(3)-field (w/ SE(3)-field) results in minor artifacts in the foreground reconstruction. Removing the deformation field entirely (w/o deform. field) produces coarse object reconstructions that fail to model moving body parts such as limbs. Removing object root-body poses (w/o root-body (SE3)) breaks foreground reconstruction entirely, as evidenced by the ghosting artifacts indicative of failed foreground reconstruction. These experiments justify our method’s hierarchical motion representation, where object motion is decomposed into global root-body motion and local articulations. <a href="#">[Videos]</a> . . . . .</p>	45
F.3	<p><b>Ablation Study on Joint-Finetuning.</b> Joint-finetuning enables our method to learn the correct human-pet interactions, particularly for frames without any detected object segmentation masks. . . . .</p>	47

# List of Tables

2.1	<b>Comparison to Related Work.</b> Unlike prior work, Total-Recon exhibits all of the properties required for embodied view synthesis of scenes containing humans and pets: the ability to (1) reconstruct <i>entire scenes</i> , (2) model <i>deformable objects</i> , (3) extend <i>beyond humans</i> , (4) recover <i>global trajectories</i> of objects’ root-bodies and articulated body parts, and (5) process <i>minute-long videos</i> of dynamic scenes. . . . .	5
3.1	Rendered 2D features $\hat{\mathbf{f}}$ and their corresponding 3D features $\mathbf{f}_{ij}$ . $\Pi^{t'}$ denotes the camera intrinsics at time $t'$ . . . . .	11
4.1	<b>Baseline Comparisons.</b> We train Total-Recon, HyperNeRF [31], D <sup>2</sup> NeRF [52], and their depth-supervised variants on the <i>left</i> video captured with our stereo rig, and evaluate the novel view synthesis results on the <i>held-out</i> right video. Total-Recon significantly outperforms all of the baselines for all 11 sequences, both in terms of LPIPS and RMS depth error, $\epsilon_{\text{depth}}$ (units: meters). . . . .	18
4.2	<b>Ablation Study.</b> Removing depth supervision ( $\mathcal{L}_{\text{depth}}$ ) significantly deteriorates the metrics. Removing the deformation field $\mathbf{J}_j$ moderately hurts the metrics, while removing object root-body motion $\mathbf{G}_j$ prevents the optimization from converging (since $\mathbf{J}_j$ alone has to explain global object motion; see Figure 3.1). This justifies hierarchical modeling of motion, where object motion is decomposed into global root-body motion and local articulations. We visualize qualitative results in Figure 4.4 and include additional ablations in Appendix F. . . . .	19
D.1	<b>Baseline Comparisons (Visual Metrics).</b> We compare our method to HyperNeRF [31], D <sup>2</sup> NeRF [52], and their depth-supervised variants in terms of their novel-view visual metrics. Our method significantly outperforms all baselines in terms of LPIPS and PSNR for all sequences of our stereo RGBD dataset and the mean SSIM averaged over all sequences. . . . .	33

D.2	<b>Baseline Comparisons (Depth Metrics).</b> We compare our method to HyperNeRF [31], D <sup>2</sup> NeRF [52], and their depth-supervised variants in terms of their novel-view depth metrics. Our method significantly outperforms all baselines in terms of the RMS depth error $\epsilon_{\text{depth}}$ (units: meters) and average accuracy at 0.1m for all sequences of our stereo RGBD dataset. . . . .	33
F.1	<b>Ablation Study on Depth Supervision.</b> Depth supervision improves our model in terms of the visual (LPIPS) and depth (RMS depth error $\epsilon_{\text{depth}}$ ) metrics, an observation that is consistent with the qualitative results displayed in Figure F.1. . . . .	39
F.2	<b>Ablation Study on Motion Modeling.</b> Ablating camera-pose optimization (row 2), changing the deformation field (row 3), or removing it (row 4), results in moderate deterioration of the visual and depth metrics. Removing object root-body poses (row 5) prevents our method from converging for most sequences (N/A), as the deformation field alone has to explain global object motion (See Figure 3.1). We perform another ablation that replaces Total-Recon’s neural blend skinning (NBS) deformation field with the more flexible SE(3)-field [30], which does converge but still performs worse than other ablations. These experiments justify our method’s hierarchical motion representation, where object motion is decomposed into global root-body motion and local articulations. <sup>†</sup> We freeze the camera poses when ablating root-body poses to ensure that the object fields, which are now all defined in the world space, have the same camera poses throughout their separate pretraining processes. . . . .	41
F.3	<b>Ablation Study on Joint-Finetuning.</b> Joint fine-tuning generally improves LPIPS but doesn’t always reduce depth error (units: meters). We posit that using a depth signal during pretraining of the individual objects is sufficient for learning a metric model, such that the qualitative improvements induced by joint-finetuning (Figure F.3) are not always reflected in the metrics. . . . .	46

# Chapter 1

## Introduction

We explore *embodied view synthesis*, a new class of novel-view synthesis tasks that renders deformable scenes from novel 6-DOF trajectories reconstructed from *in-scene* motion of actors: egocentric cameras [6, 39] that simulate the point-of-view of moving actors and 3rd-person-follow cameras [6, 47] that track a moving actor from behind (Figure 1.1). We focus on everyday scenes of people interacting with their pets, producing renderings from the point-of-view of the person *and pet* (Figure 1.1). While such camera trajectories could be manually constructed (e.g., by artists via keyframing), building an *automated* system is an interesting problem of its own: spatial cognition theory [49] suggests that the ability to visualize behavior from another actor’s perspective is necessary for action learning and imitation; in the context of gaming and virtual reality [6, 39], egocentric cameras offer high levels of user immersion, while 3rd-person-follow cameras provide a large field of view that is useful for exploring a user’s environment.

### 1.1 Challenges

Building a system for embodied view synthesis is challenging for many reasons. First, to reconstruct everyday-but-interesting content, it needs to process long, monocular captures of multiple interacting actors. However, such videos are likely to contain large scene motions, which we demonstrate are difficult to reconstruct with current approaches. Second, it needs to produce a deformable 3D scene representation that

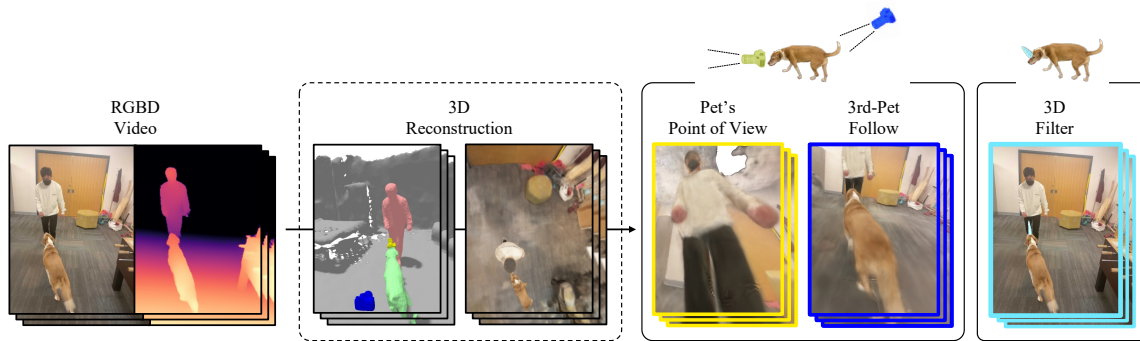


Figure 1.1: **Embodied View Synthesis.** Given a long video of deformable objects captured by a handheld RGBD sensor, Total-Recon renders the scene from novel camera trajectories derived from *in-scene* motion of actors: (1) egocentric cameras that simulate the *point-of-view of a target actor* (such as the pet) and (2) 3rd-person (or pet) cameras that follow the actor from behind. Our method also enables (3) 3D video filters that attach virtual 3D assets to the actor. Total-Recon achieves this by reconstructing the geometry, appearance, root-body- and articulated motion of each deformable object in the scene as well as the background. [\[Videos\]](#)

supports free-viewpoint synthesis, which also would benefit from long videos that are likely to capture the scene from diverse viewpoints. Recent approaches have extended Neural Radiance Fields (NeRFs) [23] to deformable scenes, but such work is often limited to rigid-only object motion [14, 28], short videos with limited scene motion [8, 17, 30, 31, 35, 48, 50, 52, 53], or reconstructing single objects as opposed the entire scene [4, 57, 58, 59]. Third, it needs to compute global 6-DOF trajectories of root-bodies and articulated body parts (e.g., head) of multiple actors.

## 1.2 Key Ideas

To address these challenges, we introduce Total-Recon, the first monocular NeRF that enables embodied view synthesis for deformable scenes with large motions. Given a monocular RGBD video, Total-Recon reconstructs the scene as a composition of object-centric representations, which encode the 3D appearance, geometry, and motion of each deformable object and the background. Crucially, Total-Recon hierarchically decomposes the motion of the scene into the motion of individual objects, which itself is decomposed into global root-body movement and the local

deformation of articulated body parts. We demonstrate that this crucial design choice allows reconstruction to scale to longer videos, enabling free-viewpoint synthesis. By reconstructing such motions in a globally-consistent coordinate frame, Total-Recon can generate renderings from egocentric and 3rd-person-follow cameras, as well as static but extreme viewpoints like bird’s-eye-views.

### 1.3 Evaluation

Due to the difficulty of collecting ground-truth data for embodied view synthesis on in-the-wild videos, we evaluate our method on the proxy task of stereo-view synthesis [30], which compares rendered views to those captured from a stereo pair. To this end, we build a stereo RGBD sensor capture rig for ground-truthing and collect a dataset of 11 long video sequences in various indoor environments, including people interacting with their pets. Total-Recon outperforms the state-of-the-art monocular deformable NeRF methods [31, 52], even when modified to use depth sensor measurements.

### 1.4 Contributions

In summary, our contributions are: (1) Total-Recon, a hierarchical 3D representation that models deformable scenes as a composition of object-centric representations, each of which decomposes object motion into its global root-body motion and its local articulations; (2) a system based on Total-Recon for automated embodied view synthesis from casual, minute-long RGBD videos of highly dynamic scenes; (3) a dataset of stereo RGBD videos containing various deformable objects, such as humans and pets, in a host of different background environments. Code, videos, and data can be found on our [project page](#).

# Chapter 2

## Related Works

### 2.1 Neural Radiance Fields

Prior works on Neural Radiance Fields (NeRF) optimize a continuous scene function for novel view synthesis given a set of multi-view images, usually under the assumption of a rigid scene and densely sampled views [10, 18, 21, 22, 23, 51]. DS-NeRF [5] and Dense Depth Priors [38] extend NeRFs to the sparse-view setting by introducing depth as additional supervision. Total-Recon also operates in the sparse-view regime and uses depth supervision to reduce the ambiguities inherent to monocular, multibody, non-rigid reconstruction [29, 54]. Another line of work [14, 28] represents rigidly moving scenes as a composition of multiple object-level NeRFs. Total-Recon also leverages such an object-centric scene representation, but models scenes containing *non-rigidly* moving objects, such as humans and pets.

### 2.2 Deformable NeRFs

Recent approaches extend NeRF to monocular deformable scene reconstruction either by learning an additional function that deforms observed points in the camera space to a time-independent canonical space [30, 31, 35, 48, 52] or explicitly modeling density changes over time [8, 17, 50, 53]. Such methods are typically limited to short videos containing little scene and camera motion. They also perform novel-view

Table 2.1: **Comparison to Related Work.** Unlike prior work, Total-Recon exhibits all of the properties required for embodied view synthesis of scenes containing humans and pets: the ability to (1) reconstruct *entire scenes*, (2) model *deformable objects*, (3) extend *beyond humans*, (4) recover *global trajectories* of objects’ root-bodies and articulated body parts, and (5) process *minute-long videos* of dynamic scenes.

Method	Entire Scenes	Deform. Objects	Beyond Humans	Global Traj.	Long Videos
BANMo [59]	✗	✓	✓	✗	✓
PNF [14]	✓	✗	✓	✓	✗
NeuMan [12]	✓	✓	✗	✓	✗
SLAHMR [61]	✗	✓	✗	✓	✗
HyperNeRF [31]	✓	✓	✓	✗	✗
D <sup>2</sup> NeRF [52]	✓	✓	✓	✗	✗
Ours	✓	✓	✓	✓	✓

synthesis only over small baselines. Total-Recon belongs to the former category of prior monocular deformable NeRFs, but unlike them, our method hierarchically decomposes scene motion into the motion of each object, which itself is further decomposed into global root-body motion and local articulations. The proposed motion decomposition is what enables embodied view synthesis: it allows Total-Recon to scale to *minute-long* videos and reconstruct a deformable 3D scene representation that supports free-viewpoint synthesis; it also makes it easy to extract an object’s root-body motion, the key motion primitive required for 3rd-person-follow view synthesis.

Several works have taken different approaches to making non-rigid reconstruction more tractable. One group of work leverages human-specific priors [12, 15, 19, 27, 32, 33, 34, 46] such as human body models (e.g., SMPL), 3D skeletons, or 2D poses to achieve high reconstruction quality. We achieve similar levels of fidelity *without* relying on such shape priors, allowing Total-Recon to generalize to pets and, by extension, reconstruct human-pet interaction videos. Another body of work [11, 16, 41] achieves high-fidelity scene reconstructions by relying on synchronized multi-view video captured from a specialized camera rig ranging from 8 to 18 static cameras. In contrast, Total-Recon only requires a single video captured from a moving RGBD camera equipped with inertial sensors, which has now become widely accessible in consumer products with the advent of Apple’s iPhone and iPad Pro.

## 2.3 Reconstruction with RGBD Sensors

Depth sensors represent the third class of attempts to make non-rigid reconstruction more tractable, reducing the need for a pre-defined shape template. Kinect-fusion [24] creates a real-time system for indoor scene localization and mapping. Dynamic Fusion [25] builds a template-free dense SLAM system for dynamic objects. Later works improve RGBD reconstruction to be able to deal with topology changes [43, 44] and use correspondence matching for registration over large motions [2, 3]. Recent works have incorporated neural implicit representations to reconstruct the surface geometry and 3D motion fields for deformable objects [4, 37] or large-scale rigid scenes [1, 36] in isolation. Other works have reconstructed humans alongside small-scale objects and furniture [3, 7], but not the entire background. We aim to go even further by reconstructing the entire scene, which includes the background and multiple deformable targets such as humans and pets; not only do we reconstruct the geometry, but we also recover a radiance field that allows for photorealistic scene rendering from embodied viewpoints and other novel 6-DOF trajectories. We summarize and compare prior work to Total-Recon in terms of the system requirements for embodied view synthesis in Table 2.1.

## 2.4 Concurrent Works

Concurrent work exhibits a subset of the design choices necessary for embodied view synthesis. SLAHMR [61] reconstructs the geometry and in-scene motion of human actors but does not reconstruct scene appearance. Nerflets [63] reconstructs a compositional dynamic scene representation that models each object as a group of posed local NeRFs but is limited to scenes with rigid moving objects. RoDynRF [20] and NeRF-DS [55] reconstructs scenes containing a variety of dynamic objects (including deformable or specular objects), but both methods are limited to short videos and do not learn the object-centric representations required for embodied view synthesis.

# Chapter 3

## Method

### 3.1 Limitations of Prior Art

The state-of-the-art monocular deformable NeRFs [31, 52] decompose a deformable scene into a rigid, canonical template model and a deformation field  $\mathbf{J}^{t,\leftarrow}$  that maps the world space  $\mathbf{G}_0^t \mathbf{X}^t$  to the canonical space  $\mathbf{X}^*$ , where  $\mathbf{G}_0^t$  is the *known* camera pose at time  $t$ , and  $\mathbf{X}^t$  is a camera space point at time  $t$ :

$$\mathbf{X}^* = \mathcal{W}^{t,\leftarrow}(\mathbf{X}^t) = \mathbf{J}^{t,\leftarrow}(\mathbf{G}_0^t \mathbf{X}^t). \quad (3.1)$$

In theory, this formulation is sufficient to represent all continuous motion; it performs well on short videos containing near-rigid scenes, as the deformation field only has to learn minute deviations from the template model.

However, this motion model is difficult to scale to minute-long videos, which are more likely to contain deformable objects undergoing large translations (e.g., a person walking into another room) and pose changes (e.g., a person sitting down). As a result, the deformation field must learn large deviations from the canonical model, significantly complicating optimization. Furthermore, this motion model conflates an object’s global rigid-body movement with its local articulations, preventing it from leveraging transformation invariance. In other words, *rigidly* transforming an object, from time  $t = t_1$  to  $t = t_2$ , requires the deformation field to encode the object state at each time instance with a different set of values  $\mathbf{J}^{t_1,\leftarrow} \neq \mathbf{J}^{t_2,\leftarrow}$ , even though the

### 3. Method

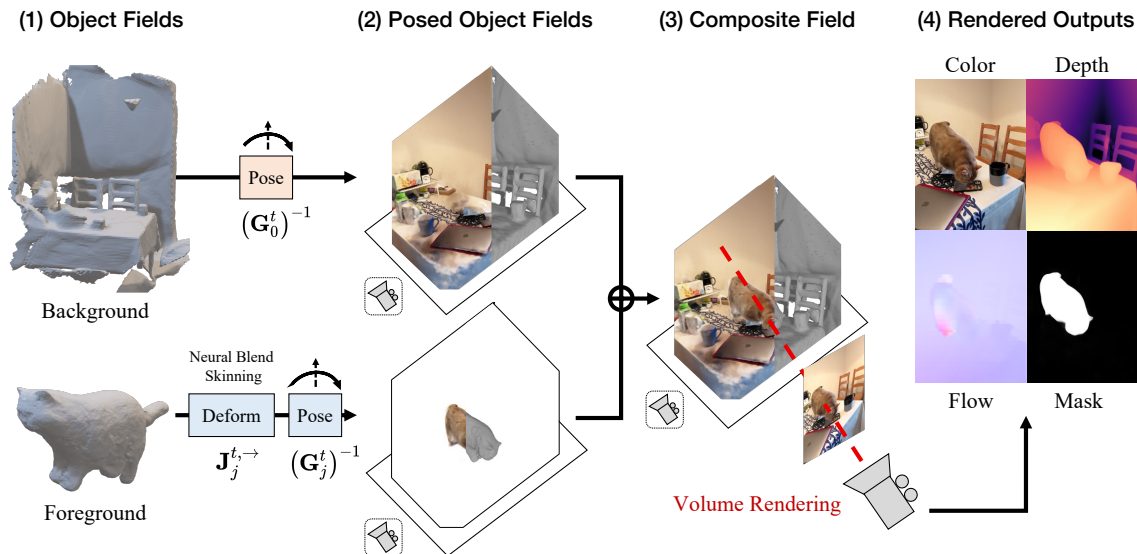


Figure 3.1: **Method Overview.** Total-Recon represents the entire scene as a composition of  $M$  object-centric neural fields, one for the rigid background and for each of the  $M - 1$  deformable objects. To render a scene, (1) *each object field  $j$*  is transformed into the camera space with a rigid transformation  $(\mathbf{G}_j^t)^{-1}$  that encodes root-body motion and, for each deformable object, an additional deformation field  $\mathbf{J}_j^{t, \rightarrow}$  that encodes articulated motion. Next, all (2) *posed object fields* are combined into a (3) *composite field*, which is then volume-rendered into (4) *color, depth, optical flow, and object silhouettes*. Each rendered output defines a reconstruction loss that derives supervision from a monocular RGBD video captured by a moving iPad Pro.

object’s body pose remains the same. This represents a missed opportunity for easier optimization.

Another critical limitation of HyperNeRF [31] and D<sup>2</sup>NeRF [52] is that they cannot track separate deformable objects and therefore cannot perform 3rd-person-follow view synthesis for scenes with *multiple* actors.

## 3.2 Component Radiance Fields

To address the limitations of existing monocular deformable NeRFs, we propose Total-Recon, a novel 3D representation that models a deformable scene as a composition of  $M$  object-centric neural fields, one for the rigid background and for each of the  $M - 1$  deformable objects (Figure 3.1). Crucially, Total-Recon hierarchically decomposes

the scene motion into the motion of each object, which itself is decomposed into global root-body motion and local articulations. This key design choice scales our method to minute-long videos containing highly dynamic and deformable objects.

**Background Radiance Field.** We begin by modeling the background environment as a Neural Radiance Field (NeRF) [23]. For a 3D point  $\mathbf{X}^* \in \mathbb{R}^3$  and a viewing direction  $\mathbf{v}^*$  in the canonical world space, NeRF defines a color  $\mathbf{c}$  and density  $\sigma$  represented by an MLP. We follow contemporary variants [21] that include a time-specific embedding code  $\omega_e^t$  to model illumination changes over time and model density with as a function of a neural signed distance function (SDF)  $\mathbf{MLP}_\sigma(\cdot) = \alpha\Gamma_\beta(\mathbf{MLP}_{\text{SDF}}(\cdot))$  [60] to encourage the reconstruction of a valid surface:

$$\sigma = \mathbf{MLP}_\sigma(\mathbf{X}^*), \quad \mathbf{c}^t = \mathbf{MLP}_\mathbf{c}(\mathbf{X}^*, \mathbf{v}^*, \omega_e^t). \quad (3.2)$$

The pixel color can then be computed with differentiable volume rendering equations (Chapter 3.3).

Most NeRF methods, including HyperNeRF [31] and D<sup>2</sup>NeRF [52], assume images with known cameras. While our capture devices are equipped with inertial sensors, we find their self-reported camera poses have room for improvement. As such, we also model camera pose as an *optimizable* rigid-body transformation  $\mathbf{G}_0^t \in SE(3)$  that maps points in a time-specific camera space  $\mathbf{X}^t \in \mathbb{R}^3$  to the world space (where we assume homogenous notation):

$$\mathbf{X}^* = \mathbf{G}_0^t \mathbf{X}^t. \quad (3.3)$$

**Deformable Field (for Object  $j$ ).** We model the deformable radiance field of object  $j \in \{1, \dots, M-1\}$  with BANMo [59], which consists of a canonical rest shape and time-*dependent* deformation field. The canonical rest shape is represented by the same formulation described by Equation 3.2, but now defined in a local *object-centric canonical space* rather than the world space. BANMo represents object motion with a warping function  $\mathcal{W}_j^{t,\leftarrow} : \mathbf{X}^t \rightarrow \mathbf{X}_j^*$  that maps the camera space points  $\mathbf{X}^t$  to canonical space points  $\mathbf{X}_j^*$  with a rigid-body transformation  $\mathbf{G}_j^t \in SE(3)$  and a deformation field  $\mathbf{J}_j^{t,\leftarrow}$  modeled by linear blend skinning [9]:

$$\mathbf{X}_j^* = \mathcal{W}_j^{t,\leftarrow}(\mathbf{X}^t) = \mathbf{J}_j^{t,\leftarrow}(\mathbf{G}_j^t \mathbf{X}^t). \quad (3.4)$$

Note that our choice of deformation field differs from the  $SE(3)$ -field used in HyperNeRF and D<sup>2</sup>NeRF, which has been shown to produce irregular deformation in the presence of complex scene motion [59]. Intuitively, rigid-body transformation  $\mathbf{G}_j^t$  captures the global root-body pose of object  $j$  relative to the camera at time  $t$ , while deformation field  $\mathbf{J}_j^{t,\leftarrow}$  aligns more fine-grained articulations relative to its local canonical space (Figure 3.1). Explicitly disentangling these two sources of object motion (as opposed to conflating them) enables easier optimization of the deformation field, because local articulations are significantly easier to learn than those modeled relative to the world space (Equation 3.1). Furthermore, this motion decomposition makes the deformation field invariant to rigid-body transformations of the object. A motion model similar to ours was proposed by ST-NeRF [11], but their model encodes an object’s global root-body motion with a 3D axis-aligned bounding box that does not explicitly represent object orientation, a prerequisite for embodied view synthesis from 3rd-person-follow cameras.

As did BANMo, Total-Recon also models a forward warp  $\mathbf{X}_j^t = \mathcal{W}_j^{t,\rightarrow}(\mathbf{X}^*) = (\mathbf{G}_j^t)^{-1} \mathbf{J}_j^{t,\rightarrow}(\mathbf{X}^*)$  that maps the canonical space to the camera space, which is used to establish the surface correspondences required for egocentric view synthesis and 3D video filters.

### 3.3 Composite Rendering of Multiple Objects

Given a set of  $M$  object representations (the background is treated as an object as well), we use the composite rendering scheme from prior work [26, 45] to combine the outputs of all object representations and volume-render the entire scene. To volumetrically render the image at frame  $t$ , we sample multiple points along each camera ray  $\mathbf{v}^t$ . Denoting the  $i^{\text{th}}$  sample as  $\mathbf{X}_i^t$ , we write the density and color observed at sample  $i$  due to object  $j$  as:

$$\sigma_{ij} = \text{MLP}_{\sigma,j}(\mathbf{X}_{ij}^*), \quad \mathbf{c}_{ij} = \text{MLP}_{\mathbf{c},j}(\mathbf{X}_{ij}^*, \mathbf{v}_j^*, \omega_e^t),$$

Table 3.1: Rendered 2D features  $\hat{\mathbf{f}}$  and their corresponding 3D features  $\mathbf{f}_{ij}$ .  $\Pi^{t'}$  denotes the camera intrinsics at time  $t'$ .

Rendered 2D features $\hat{\mathbf{f}}$ at pixel $\mathbf{x}^t$	Corresponding 3D features $\mathbf{f}_{ij}(\mathbf{X}_i^t)$
color $\hat{\mathbf{c}}(\mathbf{x}^t)$	$\mathbf{c}_i^t(\mathcal{W}_j^{t', \leftarrow}(\mathbf{X}_i^t))$
flow $\hat{\mathcal{F}}(\mathbf{x}^t, t \rightarrow t')$	$\Pi^{t'}(\mathcal{W}_j^{t', \rightarrow}(\mathcal{W}_j^{t', \leftarrow}(\mathbf{X}_i^t))) - \mathbf{x}^t$
depth $\hat{\mathbf{d}}(\mathbf{x}^t)$	$[0, 0, 1] \cdot \mathbf{X}_i^t$

where  $\mathbf{X}_{ij}^* = \mathcal{W}_j^{t', \leftarrow}(\mathbf{X}_i^t)$  and  $\mathbf{v}_j^* = \mathcal{W}_j^{t', \leftarrow}(\mathbf{v}^t)$  are sample  $i$  and camera ray  $\mathbf{v}^t$  backward-warped into object  $j$ 's canonical space, respectively. The composite density  $\sigma_i$  at sample  $i$  along the ray is then computed as the sum of each object's density  $\sigma_{ij}$ ; the composite color  $\mathbf{c}_i$  is computed as the weighted sum of each object's color  $\mathbf{c}_{ij}$ , where the weights are the normalized object densities  $\sigma_{ij}/\sigma_i$ :

$$\sigma_i = \sum_{j=0}^{M-1} \sigma_{ij}, \quad \mathbf{c}_i = \frac{1}{\sigma_i} \sum_{j=0}^{M-1} \sigma_{ij} \mathbf{c}_{ij}. \quad (3.5)$$

We can then use the standard volume rendering equations to generate an RGB image of the scene, where  $N$  is the number of sampled points along camera ray  $\mathbf{v}^t$ ,  $\tau_i$  is the transmittance,  $\alpha_i$  is the alpha value for sample point  $i$  and  $\delta_i$  is the distance between sample point  $i$  and the  $(i+1)$ :

$$\hat{\mathbf{c}} = \sum_{i=1}^N \tau_i \alpha_i \mathbf{c}_i, \quad \tau_i = \prod_{k=1}^{i-1} (1 - \alpha_k), \quad \alpha_i = 1 - e^{-\sigma_i \delta_i}. \quad (3.6)$$

### 3.3.1 Rendering Flow, Depth, and Silhouettes

Our composite rendering scheme can be used to render different quantities by replacing the object color  $\mathbf{c}_{ij}$  in Equation 3.5 with the appropriately defined 3D *feature*  $\mathbf{f}_{ij}$  (Table 3.1) and rendering the resulting composite feature  $\mathbf{f}_i$ . For occlusion-aware object silhouettes, we render a categorical distribution over  $M$  objects following [45]:

$$\hat{\mathbf{o}}_j = \sum_{i=1}^N \tau_i \alpha_{ij}, \quad \text{where} \quad \tau_i = \prod_{k=1}^{i-1} (1 - \alpha_k), \quad (3.7)$$

$$\alpha_i = 1 - e^{-\sigma_i \delta_i}, \quad \alpha_{ij} = 1 - e^{-\sigma_{ij} \delta_i}. \quad (3.8)$$

### 3.3.2 Optimization

Given a monocular RGBD video, we optimize all parameters in our composite scene representation, which for each of the  $M$  objects includes the appearance and shape MLPs ( $\mathbf{MLP}_{c,j}$ ,  $\mathbf{MLP}_{\sigma,j}$ ), rigid-body transformations  $\mathbf{G}_j^t$ , and forward, backward deformation fields  $\mathbf{J}_j^{\leftarrow}$ ,  $\mathbf{J}_j^{\rightarrow}$ . The model is learned by optimizing three reconstruction losses: a color loss  $\mathcal{L}_{\text{rgb}}$ , a flow loss  $\mathcal{L}_{\text{flow}}$ , and crucially, a depth loss  $\mathcal{L}_{\text{depth}}$ , where the ground truth color  $\mathbf{c}$  and depth  $\mathbf{d}$  are provided by the RGBD video, and the ‘‘ground truth’’ flow  $\mathcal{F}$  is computed by an off-the-shelf flow network [56]. The model also optimizes a 3D-cycle consistency loss  $\mathcal{L}_{\text{cyc},j}$  [59] for each deformable object to encourage their forward and backward warps to be consistent:

$$\mathcal{L}_{\text{rgb}} = \sum_{\mathbf{x}^t} \|\mathbf{c}(\mathbf{x}^t) - \hat{\mathbf{c}}(\mathbf{x}^t)\|^2, \quad (3.9)$$

$$\mathcal{L}_{\text{flow}} = \sum_{\mathbf{x}^t} \|\mathcal{F}(\mathbf{x}^t) - \hat{\mathcal{F}}(\mathbf{x}^t)\|^2, \quad (3.10)$$

$$\mathcal{L}_{\text{depth}} = \sum_{\mathbf{x}^t} \|\mathbf{d}(\mathbf{x}^t) - \hat{\mathbf{d}}(\mathbf{x}^t)\|^2, \quad (3.11)$$

$$\mathcal{L}_{\text{cyc},j} = \sum_i \tau_i \alpha_{ij} \left\| \mathcal{W}_j^{t', \rightarrow} (\mathcal{W}_j^{t, \leftarrow}(\mathbf{X}_i^t)) - \mathbf{X}_i^t \right\|^2, \quad (3.12)$$

where  $\mathbf{x}^t \in \mathbb{R}^2$  is a pixel location at time  $t$ .

## 3.4 Embodied View Synthesis and 3D Filters

To enable embodied view synthesis and 3D video filters from Total-Recon’s scene reconstruction, we design a simple interface that allows a user to select a point on a target object’s surface in its reconstructed canonical mesh, and use its forward warping function  $\mathcal{W}_j^{t, \rightarrow} : \mathbf{X}^* \rightarrow \mathbf{X}^t$  followed by the rigid-body transformation  $\mathbf{G}_0^t$  to place

the egocentric camera (or virtual 3D asset) in the world space. The surface normal to the object’s mesh at the user-defined point provides a reference frame to align the egocentric camera’s viewing direction and place the 3D asset. To implement a 3rd-person-follow camera, we add an user-defined offset to the object’s local reference frame, which is defined by its root-body pose.

# Chapter 4

## Experiments

### 4.1 Implementation Details

We initialize the rigid-body transformations of each deformable object  $\mathbf{G}_j^t$  using a pre-trained PoseNet [59]; we initialize the rigid-body transformation of the background  $\mathbf{G}_0^t$  with the camera poses provided by the iPad Pro. In practice, what we actually optimize are the *inverse* of the rigid-body transformations *i.e.*, the root-body poses of each object relative to the camera. To train our composite scene representation, we first pre-train each object field separately. When pretraining the deformable objects, we optimize a silhouette loss  $\mathcal{L}_{\text{mask}} = \sum_{\mathbf{x}^t} \|\mathbf{o}_j(\mathbf{x}^t) - \hat{\mathbf{o}}_j(\mathbf{x}^t)\|^2$ , where the “ground truth” object silhouette  $\mathbf{o}_j$  is computed by an off-the-shelf instance segmentation engine [13]. For pretraining the background, we optimize rgb, flow, and depth losses (Equations 3.9, 3.10, 3.11, 3.12) on pixels outside the ground truth object silhouettes. Importantly, we don’t supervise the object fields on frames that are not provided an object silhouette since it cannot be determined whether the absence of detection is a true or false negative. After pretraining, we composite the pretrained object fields and jointly finetune them using only the color, depth, flow, and object-specific 3D-cycle consistency losses. Since the silhouette loss is no longer used, the scene representation is supervised on *all* frames of the training sequence during joint-finetuning. We provide the complete description of implementation details in Appendix B.

## 4.2 Dataset

We evaluate Total-Recon on novel-view synthesis for deformable scenes. To enable quantitative evaluation, we built a stereo rig comprised of two iPad-Pros rigidly attached to a camera mount, a setup similar to that of Nerfies [30]. Using the stereo rig, we captured 11 RGBD sequences containing 3 different cats, 1 dog, and 2 human subjects in 4 different indoor environments. The RGBD videos were captured using the Record3D iOS App [42], which also automatically registers the frames captured by each camera. These video sequences, which were subsampled at 10 fps, range from 392 to 901 frames, amounting to, on average minute-long videos that are significantly longer and contain more dynamic motion than the datasets introduced by [30, 31, 52]. The left and right cameras were registered by solving a Perspective-n-Point (PnP) problem using manually annotated correspondences, and their videos were synchronized based on audio. We provide a complete description of our dataset in Appendix C.

## 4.3 Reconstruction and Applications

By hierarchically decomposing scene motion into the motion of each object, which itself is decomposed into root-body motion and local articulations, Total-Recon *automatically* computes novel 6-DoF trajectories such as those traversed by egocentric cameras and 3rd-person follow cameras (Figure 4.1). In turn, these trajectories enable automated embodied view synthesis and 3D occlusion-aware video filters (Figure 4.2). These tasks are also enabled by Total-Recon’s ability to recover an accurate deformable 3D scene representation, which is currently out of reach for the best of related methods (Figure 4.3). As shown in the bird’s eye view, each reconstructed object is properly situated with respect to the background and other objects, a direct consequence of our use of depth supervision. Furthermore, although the iPad Pro is only able to measure depth up to 4m, Total-Recon can render depth for *beyond* this sensor limit by pooling the depth information from other frames into a single metric scene reconstruction. We provide results on additional sequences in Appendix E.

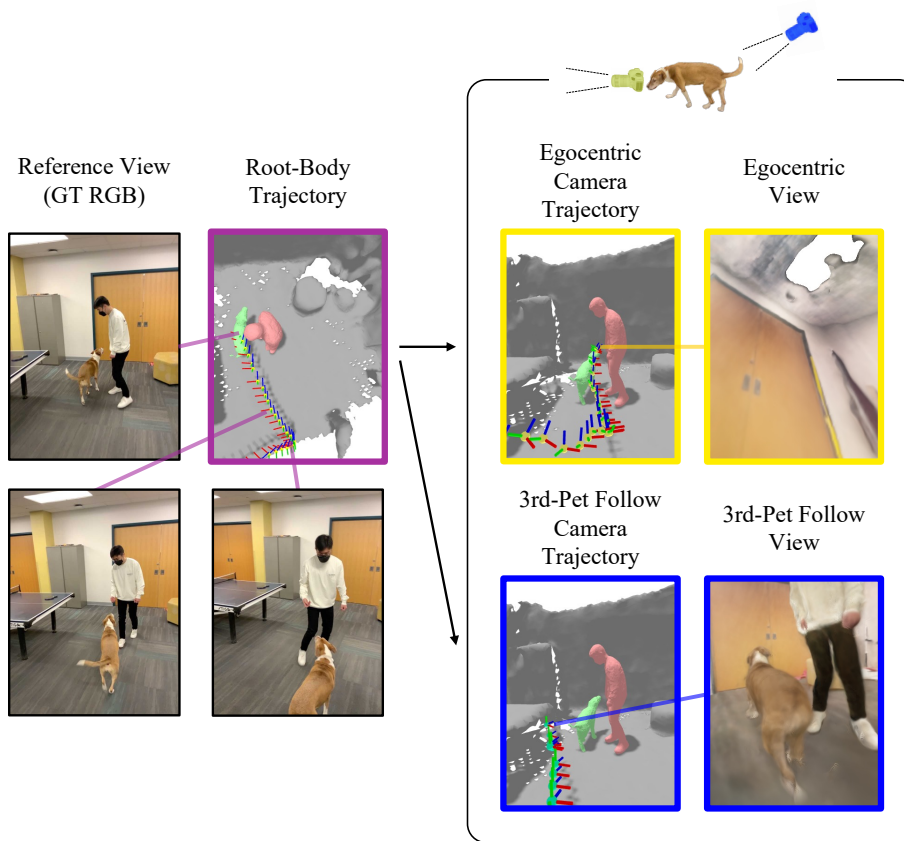


Figure 4.1: **6-DOF Trajectories for Embodied View Synthesis.** To synthesize embodied views from egocentric or actor-following cameras, Total-Recon reconstructs the entire background, every individual actor in the scene, as well as global 6-DOF trajectories of its root-body and its articulated body parts (e.g. head). [\[Videos\]](#)

## 4.4 Baselines and Evaluation

In Table 4.1 and Figure 4.3, we compare Total-Recon to D<sup>2</sup>NeRF [52] and HyperNeRF [31], and their depth-supervised equivalents on the proxy task of stereo-view synthesis, a prerequisite for *embodied* view synthesis: we train each method on the RGBD frames captured from the left camera of our dataset and evaluate the images rendered from the viewpoint of the right camera. The depth-supervised versions of the baselines contain the same depth loss used in Total-Recon. We report LPIPS [62] and RMS depth error in the main paper, but include a more complete set of metrics (PSNR, SSIM, average accuracy at 0.1m) in Appendix D. Because D<sup>2</sup>NeRF and HyperNeRF were not

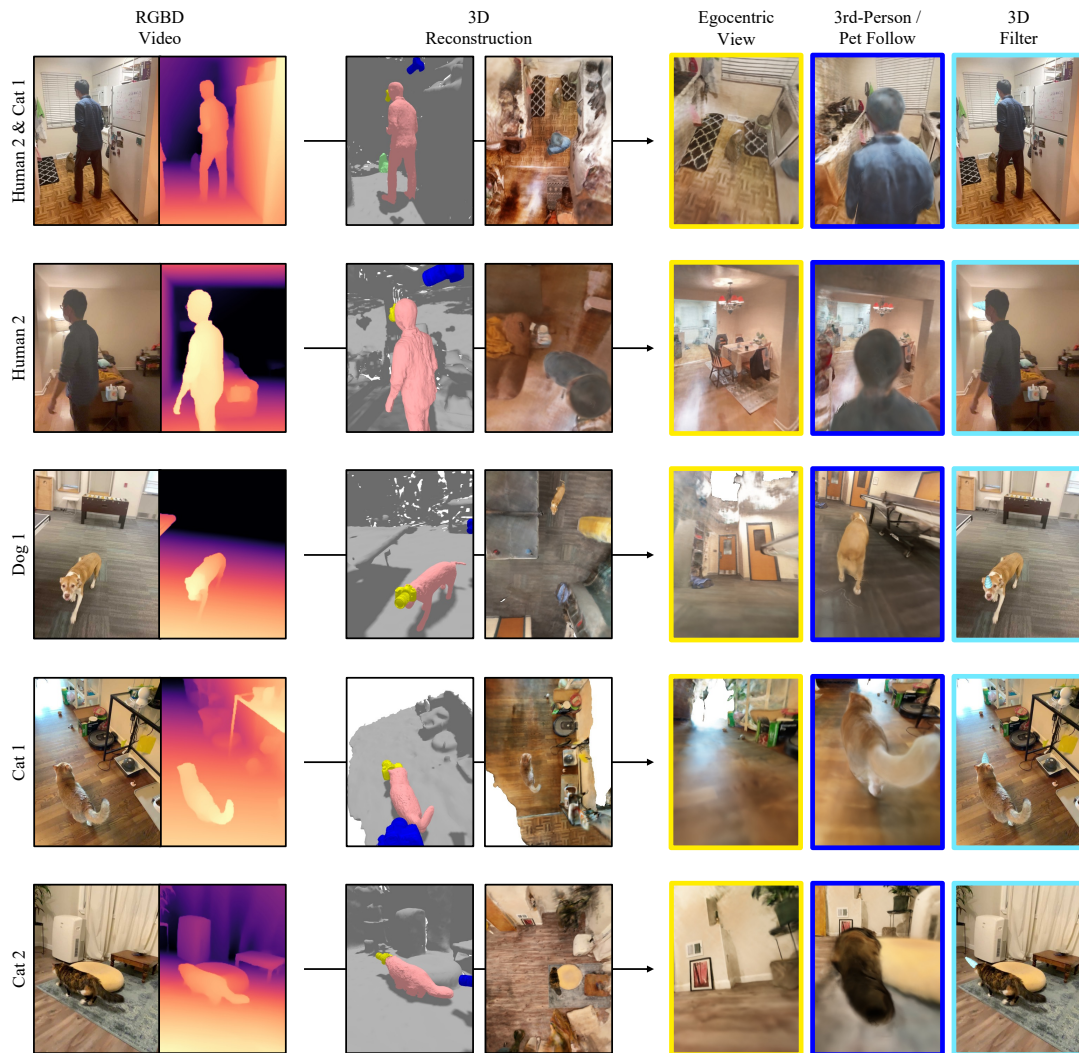


Figure 4.2: **Embodied View Synthesis and 3D Filters.** For select sequences of our RGBD dataset, we visualize the scene geometry and appearance reconstructed by our method (3D reconstruction) and the resulting downstream applications. The yellow and blue camera meshes in the mesh renderings represent the ego-centric and 3rd-person-follow cameras, respectively. To showcase the 3D video filter we attach a sky-blue unicorn horn to the forehead of the target object, which is then automatically propagated across all frames. [\[Videos\]](#)

## 4. Experiments

Table 4.1: **Baseline Comparisons.** We train Total-Recon, HyperNeRF [31], D<sup>2</sup>NeRF [52], and their depth-supervised variants on the *left* video captured with our stereo rig, and evaluate the novel view synthesis results on the *held-out* right video. Total-Recon significantly outperforms all of the baselines for all 11 sequences, both in terms of LPIPS and RMS depth error,  $\epsilon_{\text{depth}}$  (units: meters).

	DOG 1 (v1) (626 images)		DOG 1 (v2) (531 images)		CAT 1 (v1) (641 images)		CAT 1 (v2) (632 images)		CAT 2 (v1) (834 images)		CAT 2 (v2) (901 images)	
	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$
HyperNeRF [31]	.634	.687	.432	.870	.521	.476	.438	.564	.641	.765	.397	.811
D <sup>2</sup> NeRF [52]	.540	.463	.546	.456	.687	.334	.588	.314	.556	.371	.595	.361
HyperNeRF (w/ depth)	.373	.331	.425	.338	.532	.206	.371	.209	.330	.154	.376	.170
D <sup>2</sup> NeRF (w/ depth)	.507	.423	.532	.445	.685	.325	.580	.313	.561	.298	.553	.318
<b>Ours (w/ depth)</b>	<b>.271</b>	<b>.165</b>	<b>.313</b>	<b>.167</b>	<b>.383</b>	<b>.183</b>	<b>.321</b>	<b>.161</b>	<b>.237</b>	<b>.050</b>	<b>.281</b>	<b>.096</b>

	CAT 3 (767 images)		HUMAN 1 (550 images)		HUMAN 2 (483 images)		HUMAN - DOG (392 images)		HUMAN - CAT (431 images)		MEAN	
	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$
HyperNeRF [31]	.592	.800	.632	.821	.585	1.665	.487	.894	.462	.862	.531	.855
D <sup>2</sup> NeRF [52]	.759	.523	.588	1.063	.630	.890	.576	.847	.628	.880	.611	.739
HyperNeRF (w/ depth)	.514	.285	.501	.591	.445	.611	.450	.565	.456	.613	.428	.374
D <sup>2</sup> NeRF (w/ depth)	.730	.496	.585	.984	.609	.813	.608	.789	.645	.757	.599	.549
<b>Ours (w/ depth)</b>	<b>.261</b>	<b>.066</b>	<b>.213</b>	<b>.142</b>	<b>.264</b>	<b>.142</b>	<b>.256</b>	<b>.204</b>	<b>.233</b>	<b>.107</b>	<b>.277</b>	<b>.136</b>

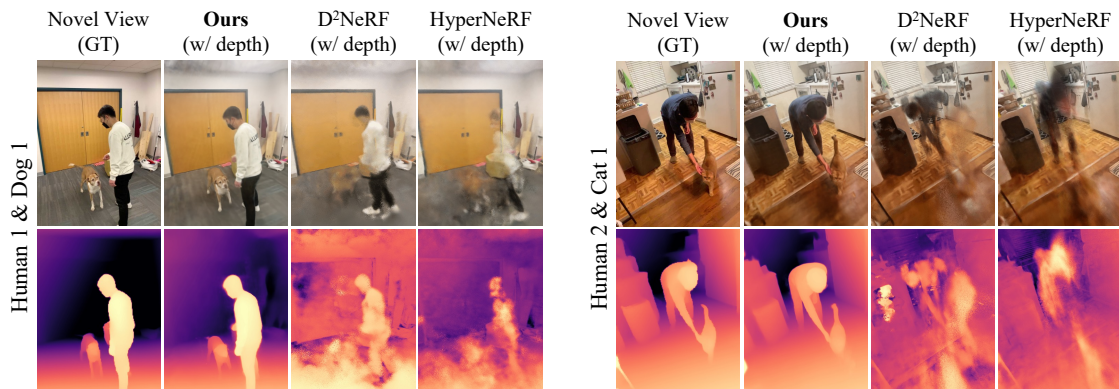


Figure 4.3: **Baseline Comparisons.** We compare Total-Recon to depth-supervised variants of HyperNeRF [31] and D<sup>2</sup>NeRF [52] on the task of stereo-view synthesis (the left camera is used for training and the right is used for testing). While the baselines are able to reconstruct only the background at best, Total-Recon can reconstruct *both* the background and the moving deformable object(s), demonstrating holistic scene reconstruction. [\[Videos\]](#)

designed to recover a *metric* scene representation, we replaced their COLMAP [40] camera poses with those metric ones provided by the iPad Pro for a fair comparison.

Table 4.2: **Ablation Study.** Removing depth supervision ( $\mathcal{L}_{\text{depth}}$ ) significantly deteriorates the metrics. Removing the deformation field  $\mathbf{J}_j$  moderately hurts the metrics, while removing object root-body motion  $\mathbf{G}_j$  prevents the optimization from converging (since  $\mathbf{J}_j$  alone has to explain global object motion; see Figure 3.1). This justifies hierarchical modeling of motion, where object motion is decomposed into global root-body motion and local articulations. We visualize qualitative results in Figure 4.4 and include additional ablations in Appendix F.

Methods	Depth Supervision	Non-Rigid Objects	Root-Body Motion	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$
(1) <b>Ours</b>	✓	✓	✓	<b>.268</b>	<b>.145</b>
(2) w/o loss $\mathcal{L}_{\text{depth}}$	✗	✓	✓	.379	.815
(3) w/o deform. $\mathbf{J}_j$	✓	✗	✓	.300	.211
(4) w/o root-body $\mathbf{G}_j$	✓	✓	✗	N/A	N/A

#### 4.4.1 Comparisons

Total-Recon qualitatively and quantitatively outperforms all of the baselines. As shown in Figure 4.3, Total-Recon successfully reconstructs the entire scene, whereas the baselines are only able to reconstruct the rigid background at best. As shown in Table 4.1, Total-Recon also significantly outperforms all baselines regarding LPIPS and the RMS depth error. We attribute this huge gap to the baselines’ inability to reconstruct the moving deformable objects. We provide more details regarding the baselines and additional visualizations in Appendix D.

## 4.5 Ablation Studies

Table 4.2 (and Figure 4.4 and 4.5) analyzes the importance of Total-Recon’s design choices (see Chapter 3) by ablating three key components: the depth reconstruction loss  $\mathcal{L}_{\text{depth}}$  (row 2), deformation field  $\mathbf{J}_j^t$  (row 3), and root-body pose  $\mathbf{G}_j^t$  (row 4), where  $j$  denotes a deformable actor. For all experiments, we use the same set of training losses used in Total-Recon and initialize camera pose  $\mathbf{G}_0^t$  with those reported by ARKit; for rows (2) - (3), we initialize each deformable actor’s root-body pose  $\mathbf{G}_j^t$  with predictions made by PoseNet [59] and optimize them during training. We report the novel-view metrics averaged over 6 select sequences of our dataset: DOG 1 (v1), CAT 1 (v1), CAT 2 (v1), HUMAN 1, HUMAN 1 & DOG 1, and HUMAN 2 & CAT 1.

## 4. Experiments

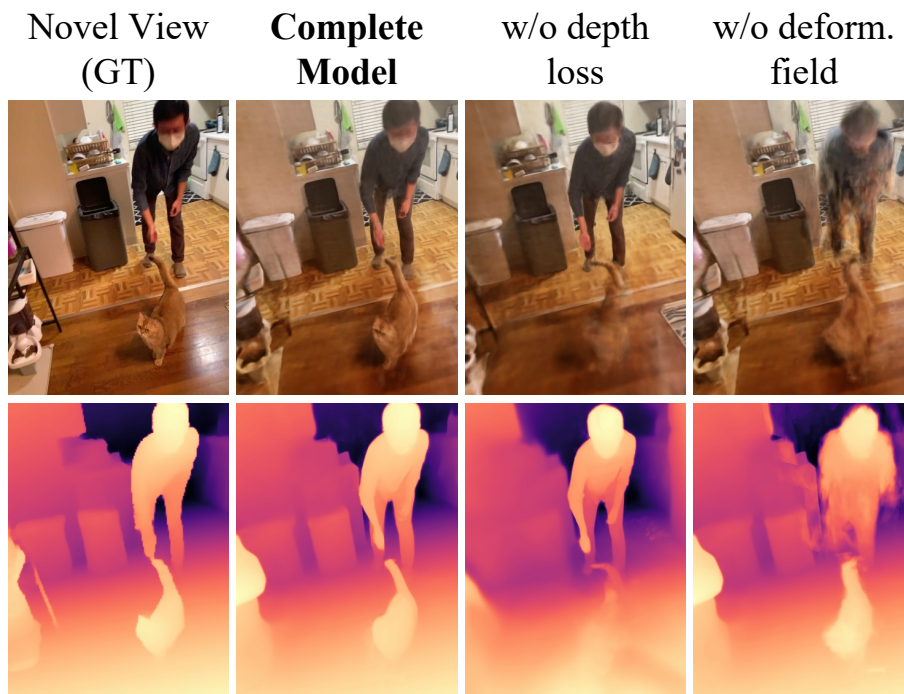


Figure 4.4: **Ablation Study.** We visualize results from the ablations in Table 4.2. Removing depth supervision causes the cat to sink into the ground due to inconsistent object scales. Removing the deformation field results in coarse object reconstructions that fail to capture articulated body parts such as moving limbs. We do not visualize Total-Recon without root-body poses as this ablation does not converge. [Videos]

### 4.5.1 Depth Supervision

Table 4.2 shows that removing depth supervision (row 2) results in a significant increase in the RMS depth error  $\epsilon_{\text{depth}}$ . Figure 4.5 indicates that this reflects the incorrect arrangement of objects stemming from their scale inconsistency - while removing depth supervision does not significantly deteriorate the training-view RGB renderings, it induces critical failure modes as shown in the *novel-view* 3D reconstructions: (a) floating foreground objects, as evidenced by their shadows, and (b) the human incorrectly occluding the dog. In other words, without depth supervision, Total-Recon overfits to the training view and learns a degenerate scene representation where the reconstructed objects fail to converge to the same scale. We show results on additional RGBD sequences in Appendix F.1.

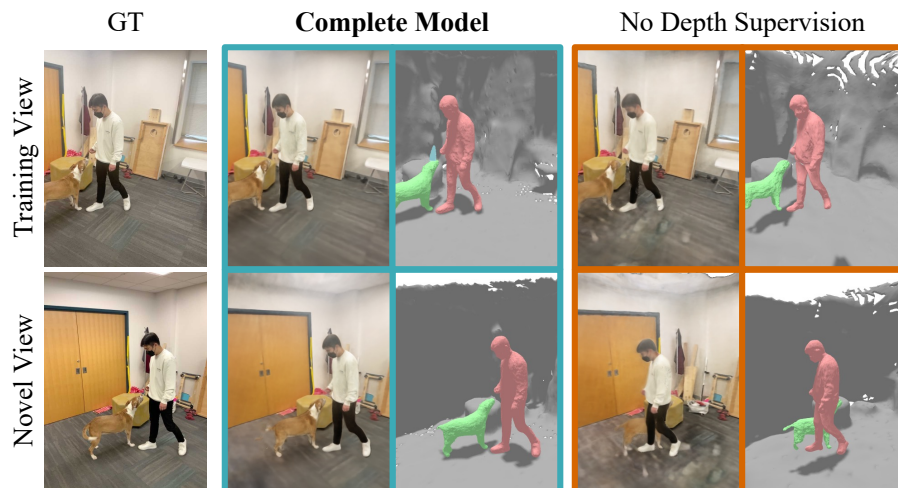


Figure 4.5: **Ablation Study on Depth Supervision.** While removing depth supervision does not significantly deteriorate training-view RGB renderings, it significantly hurts *novel-view* 3D reconstructions, as characterized by the following: (a) floating foreground objects (as evidenced by their shadows) and (b) the human incorrectly occluding the dog. These failure modes indicate that without depth supervision, Total-Recon overfits the training view, and the reconstructed objects fail to converge to the same scale. [Videos]

## 4.5.2 Motion Modeling

Table 4.2 shows that ablating the deformation field (row 3) also worsens the metrics. This is because, without the deformation field, our method has to explain each object’s (non-rigid) motion solely via its rigid, root-body poses. As a result, this ablation can only recover coarse object reconstructions that fail to model moving body parts such as limbs (see Figure 4.4). More notably, Table 4.2 shows that ablating object root-body poses (row 4) prevents our method from converging at all (for most sequences). This is a surprising observation since the deformation field should be sufficient to represent all continuous motion, at least in theory. However, when object root-body poses are removed from our method, each object’s canonical model is defined in the *static, world* coordinate frame (Equation 3.1) as opposed to the moving, object-centric coordinate frame (Equation 3.4). Therefore, the deformation field alone has to explain *global* object motion by learning potentially large deviations from the canonical model, significantly complicating optimization.

#### 4. Experiments

These diagnostics justify Total-Recon’s hierarchical motion representation, which decomposes object motion into global root-body motion and local articulations, especially given that row 3 performs better than row 4, even though row 4 models non-rigid object motion and row 3 does not. In turn, they suggest that conflating these two sources of scene motion is what prevents the baseline methods from reconstructing the highly dynamic objects that appear in our dataset. We provide a more detailed analysis in Appendix [F.2](#) with additional experiments and RGBD sequences.

# Chapter 5

## Discussion and Limitations

We have presented a new system for automated embodied view synthesis from monocular RGBD videos, focusing on videos of people interacting with their pets. Our main technical contribution is Total-Recon, a 3D representation for deformable scenes that hierarchically decomposes scene motion into the motion of each object, which in turn, is decomposed into rigid, root-body motion and local articulations; this key design choice enables easier optimization over long videos containing large motions, which are difficult to reconstruct but necessary for supporting free-viewpoint synthesis. By explicitly reconstructing the geometry, appearance, root-body- and articulated motion of each object, Total-Recon enables seeing through the eyes of people and pets and generating game-like traversals of deformable scenes from behind a target object.

### 5.1 Limitations.

In Total-Recon, scene decomposition is primarily supervised by object silhouettes computed by an off-the-shelf segmentation model [13], which may be inaccurate, especially in partial occlusion scenarios and hence may damage the resulting reconstructions and embodied view renderings. We believe that incorporating the latest advances in video instance segmentation will enable Total-Recon to be applied to more challenging scenarios. Second, Total-Recon initializes the root-body pose of each deformable object using a PoseNet [59] trained for humans and quadruped animals,

## *5. Discussion and Limitations*

which does not generalize to other object categories (e.g., birds, fish). We reserve the reconstruction of generic scenes for future work. Our model needs to be optimized on a per-sequence basis for roughly 15 hours with 4 NVIDIA RTX A5000 GPUs and is therefore not suitable for real-time applications. Incorporating recent advances in fast neural field training methods would also be an interesting avenue for future work.

# Appendix A

## Appendix Overview

The appendix is comprised of the following: additional details of Total-Recon’s implementation (Appendix B), our dataset (Appendix C), and the baselines (Appendix D), additional metrics and results for the baseline comparisons (Tables D.1 and D.2, Figure D.1), reconstructions and embodied view synthesis results on additional sequences and object removal results (Appendix E), additional ablation studies (Appendix F), and the societal impact of our method (Appendix G).

# Appendix B

## Implementation Details

### B.1 Data Preprocessing.

Before training our composite scene representation, we follow BANMo [59] by resizing the raw RGB images and the ground truth depth maps from  $960 \times 720$  and  $256 \times 192$  resolution, respectively, to a resolution of  $512 \times 512$ , which is the resolution used during training. We also scale the ground-truth depth measurements and the translation component of the ARKit camera poses (used to initialize the background’s root-body poses  $\mathbf{G}_0^t$ ) by a scaling factor of 0.2, which we empirically found to improve the pretraining of the deformable objects. After training, we scale both the ground-truth and rendered depth back to the original metric space and compute the evaluation metrics at  $480 \times 360$  resolution.

### B.2 Optimization.

As mentioned in Chapter 4.1, we optimize our composite scene representation by first pretraining each object field separately and then jointly finetuning them. For both the pretraining and joint-finetuning stages, we use the same batch size, sampled rays per batch, and sampled points per ray as BANMo [59]. Pretraining a deformable object takes 8.5 hours with 4 NVIDIA RTX A5000 GPUs, and pretraining the background takes 4.5 hours. Jointly finetuning one deformable object and the background takes

an additional 1.5 hours with 4 NVIDIA RTX A5000 GPUs, and jointly finetuning two deformable objects and the background takes an additional 2.5 hours with 4 NVIDIA RTX A6000 GPUs.

### B.3 Pretraining.

For pretraining deformable objects, we follow the training procedure of and use the same hyper-parameters as BANMo [59], which we augment with a depth reconstruction loss weighted by a default value of  $\lambda_{\text{depth}} = 5$  (for the ‘‘Human 1’’ sequence, we use a loss weight of  $\lambda_{\text{depth}} = 1.5$  for pretraining the deformable object). Following BANMo’s latest implementation, we pre-train each deformable object in three training stages, each for 24k, 6k, and 24k iterations.

For pretraining the background model, we optimize color, flow, and depth reconstruction losses  $\mathcal{L}_{\text{rgb}}$ ,  $\mathcal{L}_{\text{flow}}$ ,  $\mathcal{L}_{\text{depth}}$  on pixels outside the ground-truth object silhouettes, each with a default weight of  $\lambda_{\text{rgb}} = 0.1$ ,  $\lambda_{\text{flow}} = 1$ ,  $\lambda_{\text{depth}} = 1$ , respectively. We also optimize an eikonal loss  $\mathcal{L}_{\text{SDF}}$  [60] with a default weight of  $\lambda_{\text{SDF}} = 0.001$  to encourage the reconstruction of a valid signed distance function (SDF):

$$\mathcal{L}_{\text{SDF}} = \sum_{\mathbf{x}^t} \sum_{\mathbf{X}_i^t} (|\|\nabla_{\mathbf{X}_i^*} \text{MLP}_{\text{SDF}}(\mathbf{X}_i^*)\|_2 - 1)|^2, \quad (\text{B.1})$$

where  $\mathbf{x}^t \in \mathbb{R}^2$  denotes the pixel location at time  $t$ ,  $\mathbf{X}_i^* \in \mathbb{R}^3$  is the 3D point in the canonical world space corresponding to  $\mathbf{X}_i^t \in \mathbb{R}^3$ , the  $i^{\text{th}}$  sample in the camera space. To compute this eikonal loss, we sample 17 uniformly spaced grid of points  $\mathbf{X}_i^t$  along each camera ray  $\mathbf{v}^t$ , from a truncated region that is 0.2m long and centered at the surface point computed by backprojecting the ground-truth depth.

We pre-train the background model in two stages: in the first stage, we optimize the color, flow, depth, and eikonal losses with their respective default loss weights for 24k iterations. In the second stage, we optimize the same set of losses for another 24k iterations while fixing the background model’s root-body poses  $\mathbf{G}_0^t$ , increasing the weight of the color loss from  $\lambda_{\text{rgb}} = 0.1$  to  $\lambda_{\text{rgb}} = 1$ , and performing active sampling of pixels  $\mathbf{x}^t$  to improve the background model’s appearance, as was done in [59].

## **B.4 Joint Finetuning.**

For the joint-finetuning of all of the object models, we optimize the color, flow, depth, and per-object 3D-cycle consistency losses for another 6k iterations, each with a default weight of  $\lambda_{\text{rgb}} = 1$ ,  $\lambda_{\text{flow}} = 1$ ,  $\lambda_{\text{depth}} = 5$ , and  $\lambda_{\text{cyc}, j} = 1$ , respectively. Importantly, we freeze the background’s appearance and shape models by default and only allow its root-body poses  $\mathbf{G}_0^t$ , the foregrounds’ root-body poses  $\mathbf{G}_j^t$ , and the foregrounds’ appearance and shape models to be optimized (for the “Human 1” sequence, we use a loss weight of  $\lambda_{\text{depth}} = 1.5$ , and for the “Cat 1” sequence, we allow the background’s appearance and shape models to be optimized during joint-finetuning). We also perform active sampling of pixels  $\mathbf{x}^t$  over all deformable foreground objects. Intuitively, the joint-finetuning stage improves the appearance of the foreground objects and helps the model learn correct object-to-object interactions.

# Appendix C

## Dataset Details

In this chapter, we describe the stereo validation rig we built for the purpose of evaluation and explain in greater detail how we used the validation rig for evaluating novel-view synthesis.

As shown in Figure C.1, our stereo validation rig is comprised of two iPad Pro’s rigidly attached to a camera mount. Importantly, **we train each method *only* on the sequences captured from the left camera** and evaluate the images rendered from the viewpoint of the right camera *i.e.*, the “novel-view”. To compute the pose of the “novel-view” camera, we compute the rigid transform between the left and right cameras and use this transform to map the optimized training-view cameras of our method to the novel-view cameras. We perform the left-right camera registration for each sequence of our dataset, by solving a Perspective-n-Point (PnP) problem using manually annotated 2D-2D correspondences.

The PnP problem aims to estimate the pose of a *calibrated camera* given  $n$  3D-2D correspondences *i.e.*, a set of  $n$  3D points defined in some *world frame* and their corresponding 2D image projections. We formulate the problem of estimating the left-to-right camera transform as a PnP problem where the left camera of our validation rig corresponds to the *world frame*, and the right camera of our validation rig corresponds to the *calibrated camera*.

To obtain the required 3D-2D correspondences, we first manually annotate at least 20, 2D-2D correspondences for each sequence. Next, we obtain the 3D points defined in the left-camera frame, by backprojecting the ground truth depth observed from

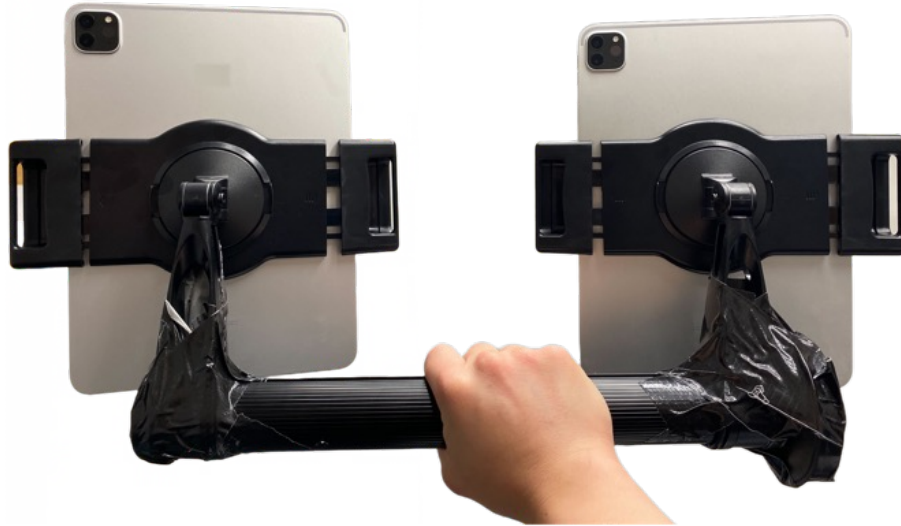


Figure C.1: **Stereo Validation Rig Used Only for Evaluation.** To enable quantitative evaluation, we built a stereo rig comprised of two iPad Pros rigidly attached to a camera mount, and captured 11 pairs of RGBD sequences. **We train each method *only* on the sequences captured from the left camera** and evaluate the images rendered from the viewpoint of the right camera.

the left camera using its intrinsics provided by ARKit. Finally, we feed 1) the 3D points in the left-camera frame, 2) the 2D annotations, and 3) the camera intrinsics of the right camera to a generic PnP solver to compute the desired left-to-right camera transform.

We have captured 11 pairs of RGBD sequences, which contains 3 different cats, 1 dog, and 2 human subjects in 4 different indoor environments. We plan to release our full dataset, which contains for each sequence 1) the RGBD images captured from both cameras of our validation rig, 2) the camera pose trajectories of both cameras, 3) the camera intrinsics for both cameras, and 4) the left-to-right camera transform.

# Appendix D

## Details of Baseline Comparisons

### D.1 Baseline Experiment Details.

We provide additional details of the experiment settings of the baselines. For the sake of fair comparison, we set up augmented versions of the baselines D<sup>2</sup>NeRF [52] and HyperNeRF [31], whereby we replace their COLMAP [40] camera poses with the iPad Pro’s camera poses provided by ARKit - the same camera poses used to initialize the root-body transforms  $\mathbf{G}_0^t$  of our method’s background model. We also compared our method to the depth-supervised variants of HyperNeRF and D<sup>2</sup>NeRF, which contain the same depth loss used in our method with a loss weight of  $\lambda_{\text{depth}} = 0.1$  for experiments on all sequences. We empirically observe that using a higher weight on the depth loss significantly deteriorates the visual quality of the baseline methods’ rendered appearance.

As was done for our method, before training, we scale the ground-truth depth measurements and the translation component of the ARKit camera poses by a scaling factor of 0.2; after training, we scale both the ground-truth and rendered depth back to the original metric space and compute the evaluation metrics at a resolution of  $480 \times 360$ . All of the other baseline hyperparameters were kept the same by default.

## **D.2 Additional Quantitative Metrics.**

In Tables [D.1](#) and [D.2](#), we display the full set of quantitative metrics for our method and all of the baselines. In addition to the LPIPS and rms depth error metrics reported in the main paper, we also report the PSNR, SSIM, and depth accuracy at 0.1m. Our method significantly outperforms all of the baselines, in terms of LPIPS, PSNR, rms depth error, depth accuracy at 0.1m for all sequences, and in terms of the mean SSIM averaged over all sequences.

## **D.3 Additional Qualitative Results.**

In Figure [D.1](#), we show qualitative comparisons on the remaining 10 sequences of our RGBD dataset that were not shown in the main paper, namely sequences HUMAN 2 & CAT1, HUMAN 2, DOG 1 (v1), DOG 1 (v2), CAT 1 (v1), CAT 1 (v2), CAT 2 (v1), CAT 2 (v2), CAT 3, HUMAN 1. Due to space and resolution limits, we only display the visualizations for the depth-supervised variants of the baselines, but we show the full set of baseline results on our [project page](#). Total-Recon outperforms all of the baselines, which at best are only able to reconstruct the rigid background. On the other hand, Total-Recon is able to reconstruct the *entire* scene, including the dynamic deformable objects.

Table D.1: **Baseline Comparisons (Visual Metrics)**. We compare our method to HyperNeRF [31], D<sup>2</sup>NeRF [52], and their depth-supervised variants in terms of their novel-view visual metrics. Our method significantly outperforms all baselines in terms of LPIPS and PSNR for all sequences of our stereo RGBD dataset and the mean SSIM averaged over all sequences.

	DOG 1 (v1) (626 images)			DOG 1 (v2) (531 images)			CAT 1 (v1) (641 images)			CAT 1 (v2) (632 images)			CAT 2 (v1) (834 images)			CAT 2 (v2) (901 images)		
	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑
HyperNeRF [31]	.634	12.84	.673	.432	14.27	.721	.521	14.86	<b>.632</b>	.438	14.87	.597	.641	12.32	.632	.397	15.68	.657
D <sup>2</sup> NeRF [52]	.540	13.37	.694	.546	11.74	.685	.687	10.92	.545	.588	11.88	.548	.556	12.55	.664	.595	12.71	.604
HyperNeRF (w/ depth)	.373	16.86	.730	.425	16.95	.740	.532	14.37	.621	.371	15.65	.617	.330	18.47	.728	.376	16.56	.670
D <sup>2</sup> NeRF (w/ depth)	.507	13.44	.698	.532	11.88	.690	.685	10.81	.534	.580	12.00	.563	.561	12.59	.656	.553	12.76	.629
<b>Ours (w/ depth)</b>	<b>.271</b>	<b>17.60</b>	<b>.745</b>	<b>.313</b>	<b>17.78</b>	<b>.768</b>	<b>.383</b>	<b>17.36</b>	.561	<b>.321</b>	<b>16.50</b>	<b>.657</b>	<b>.237</b>	<b>21.22</b>	<b>.793</b>	<b>.281</b>	<b>18.50</b>	<b>.713</b>

	CAT 3 (767 images)			HUMAN 1 (550 images)			HUMAN 2 (483 images)			HUMAN - DOG (392 images)			HUMAN - CAT (431 images)			MEAN		
	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑
HyperNeRF [31]	.592	13.74	.624	.632	11.94	.603	.585	14.97	.620	.487	15.04	.699	.462	13.52	.512	.531	14.00	.635
D <sup>2</sup> NeRF [52]	.759	11.03	.578	.588	11.88	.638	.630	12.13	.599	.576	12.41	.652	.628	10.41	.453	.611	11.97	.608
HyperNeRF (w/ depth)	.514	14.86	.635	.501	13.25	.664	.445	15.58	.665	.450	15.01	.704	.456	14.40	.535	.428	15.80	.667
D <sup>2</sup> NeRF (w/ depth)	.730	11.08	.582	.585	12.14	.638	.609	12.11	.612	.608	12.30	.633	.645	10.51	.451	.599	12.02	.611
<b>Ours (w/ depth)</b>	<b>.261</b>	<b>19.89</b>	<b>.734</b>	<b>.213</b>	<b>18.39</b>	<b>.778</b>	<b>.264</b>	<b>16.73</b>	<b>.712</b>	<b>.256</b>	<b>16.69</b>	<b>.756</b>	<b>.233</b>	<b>17.64</b>	<b>.628</b>	<b>.277</b>	<b>18.26</b>	<b>.715</b>

Table D.2: **Baseline Comparisons (Depth Metrics)**. We compare our method to HyperNeRF [31], D<sup>2</sup>NeRF [52], and their depth-supervised variants in terms of their novel-view depth metrics. Our method significantly outperforms all baselines in terms of the RMS depth error  $\epsilon_{\text{depth}}$  (units: meters) and average accuracy at 0.1m for all sequences of our stereo RGBD dataset.

	DOG 1 (v1) (626 images)		DOG 1 (v2) (531 images)		CAT 1 (v1) (641 images)		CAT 1 (v2) (632 images)		CAT 2 (v1) (834 images)		CAT 2 (v2) (901 images)	
	$\epsilon_{\text{depth}}\downarrow$	Acc.@0.1m↑	$\epsilon_{\text{depth}}\downarrow$	Acc.@0.1m↑	$\epsilon_{\text{depth}}\downarrow$	Acc.@0.1m↑	$\epsilon_{\text{depth}}\downarrow$	Acc.@0.1m↑	$\epsilon_{\text{depth}}\downarrow$	Acc.@0.1m↑	$\epsilon_{\text{depth}}\downarrow$	Acc.@0.1m↑
HyperNeRF [31]	.687	.107	.870	.176	.476	.316	.564	.314	.765	.277	.811	.252
D <sup>2</sup> NeRF [52]	.463	.219	.456	.220	.334	.346	.314	.403	.371	.333	.361	.339
HyperNeRF (w/ depth)	.331	.352	.338	.357	.206	.552	.209	.596	.154	.605	.170	.612
D <sup>2</sup> NeRF (w/ depth)	.423	.338	.445	.270	.325	.510	.313	.362	.298	.438	.318	.376
<b>Ours (w/ depth)</b>	<b>.165</b>	<b>.841</b>	<b>.167</b>	<b>.790</b>	<b>.183</b>	<b>.889</b>	<b>.161</b>	<b>.904</b>	<b>.050</b>	<b>.967</b>	<b>.096</b>	<b>.925</b>

	CAT 3 (767 images)		HUMAN 1 (550 images)		HUMAN 2 (483 images)		HUMAN - DOG (392 images)		HUMAN - CAT (431 images)		MEAN	
	$\epsilon_{\text{depth}}\downarrow$	Acc.@0.1m↑	$\epsilon_{\text{depth}}\downarrow$	Acc.@0.1m↑	$\epsilon_{\text{depth}}\downarrow$	Acc.@0.1m↑	$\epsilon_{\text{depth}}\downarrow$	Acc.@0.1m↑	$\epsilon_{\text{depth}}\downarrow$	Acc.@0.1m↑	$\epsilon_{\text{depth}}\downarrow$	Acc.@0.1m↑
HyperNeRF [31]	.800	.213	.821	.053	1.665	.067	.894	.072	.862	.162	.855	.198
D <sup>2</sup> NeRF [52]	.523	.231	1.063	.066	.890	.128	.847	.078	.880	.126	.739	.247
HyperNeRF (w/ depth)	.285	.451	.591	.211	.611	.249	.565	.283	.613	.214	.374	.439
D <sup>2</sup> NeRF (w/ depth)	.496	.243	.984	.086	.813	.131	.789	.154	.757	.176	.549	.302
<b>Ours (w/ depth)</b>	<b>.066</b>	<b>.949</b>	<b>.142</b>	<b>.909</b>	<b>.142</b>	<b>.849</b>	<b>.204</b>	<b>.827</b>	<b>.107</b>	<b>.912</b>	<b>.136</b>	<b>.896</b>

## D. Details of Baseline Comparisons

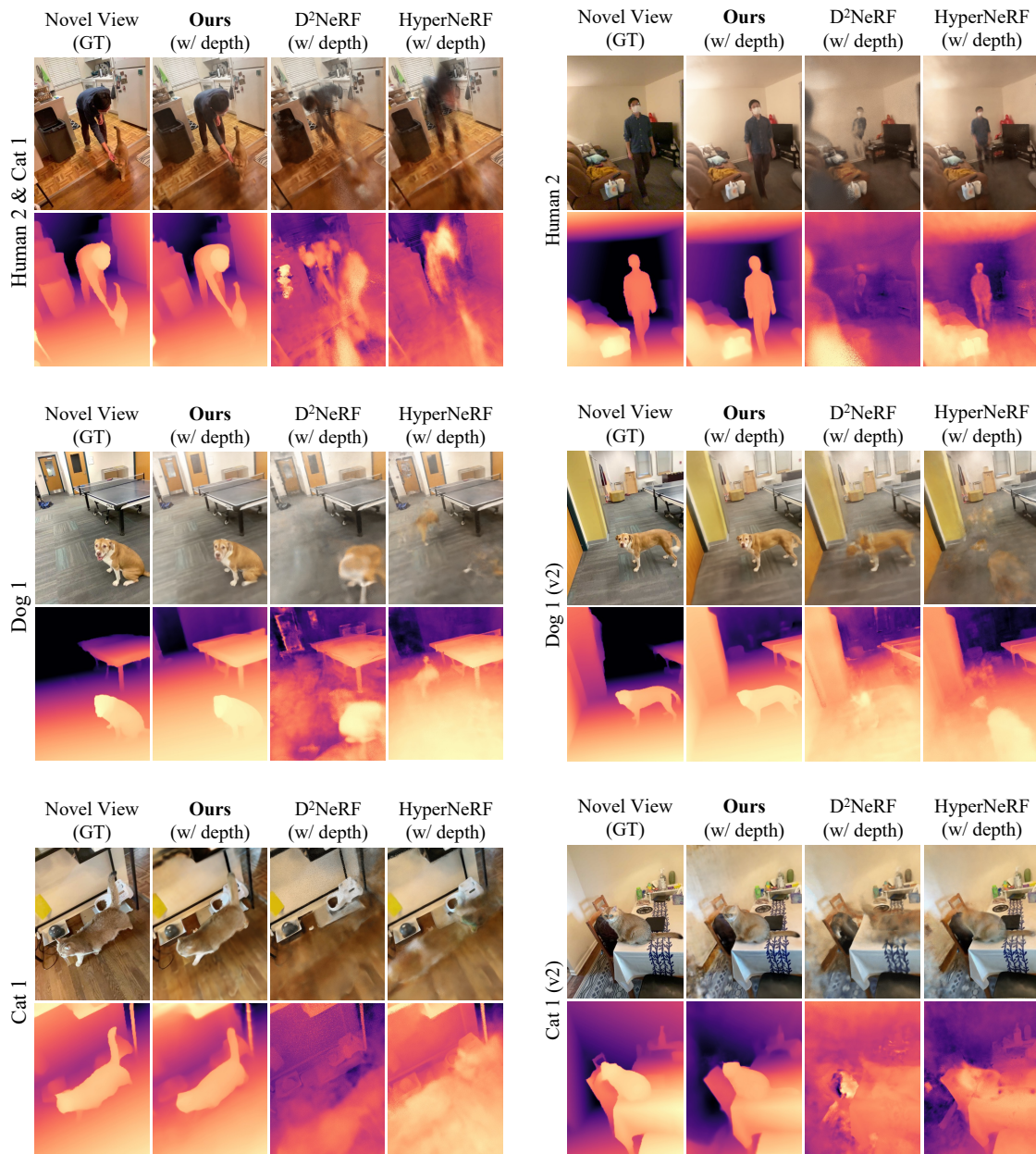


Figure D.1: **Qualitative Comparisons on Novel View Synthesis (Additional Sequences)**. We compare Total-Recon to depth-supervised variants of HyperNeRF [31] and D<sup>2</sup>NeRF [52] on the task of stereo-view synthesis (the left camera is used for training and the images are rendered to the right camera). While the baselines are only able to reconstruct the background at best, Total-Recon is able to reconstruct *both* the background and the moving deformable object(s), demonstrating holistic scene reconstruction. [\[Videos\]](#)

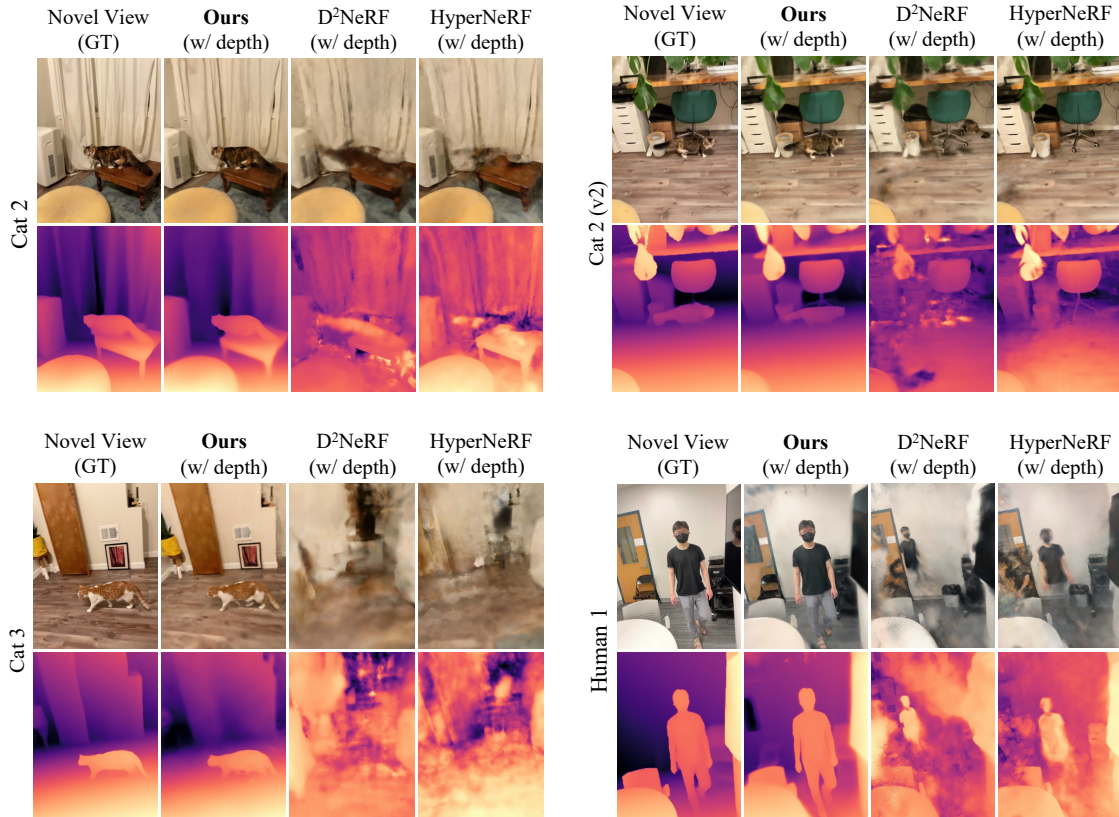


Figure D.1: [Continued] **Qualitative Comparisons on Novel View Synthesis (Additional Sequences)**. We compare Total-Recon to depth-supervised variants of HyperNeRF [31] and D<sup>2</sup>NeRF [52] on the task of stereo-view synthesis (the left camera is used for training and the images are rendered to the right camera). While the baselines are only able to reconstruct the background at best, Total-Recon is able to reconstruct *both* the background and the moving deformable object(s), demonstrating holistic scene reconstruction. [\[Videos\]](#)

# Appendix E

## Reconstruction and Applications

### E.1 Geometry and Embodied View Synthesis.

In Figure [E.1](#), we display the novel-view reconstructions and the corresponding embodied view synthesis results for the remaining 5 sequences of our RGBD dataset that were not shown in the main paper: sequences HUMAN 1, DOG 1 (v2), CAT 1 (v2), CAT 2 (v2), CAT 3.

### E.2 Object Removal.

Our compositional scene representation allows for easy object removal. To remove object  $k$  from our trained scene representation, one simply skips  $j = k$  in the summation that appears in the compositing process described by Equation [3.5](#). We showcase object removal in Figure [E.2](#).

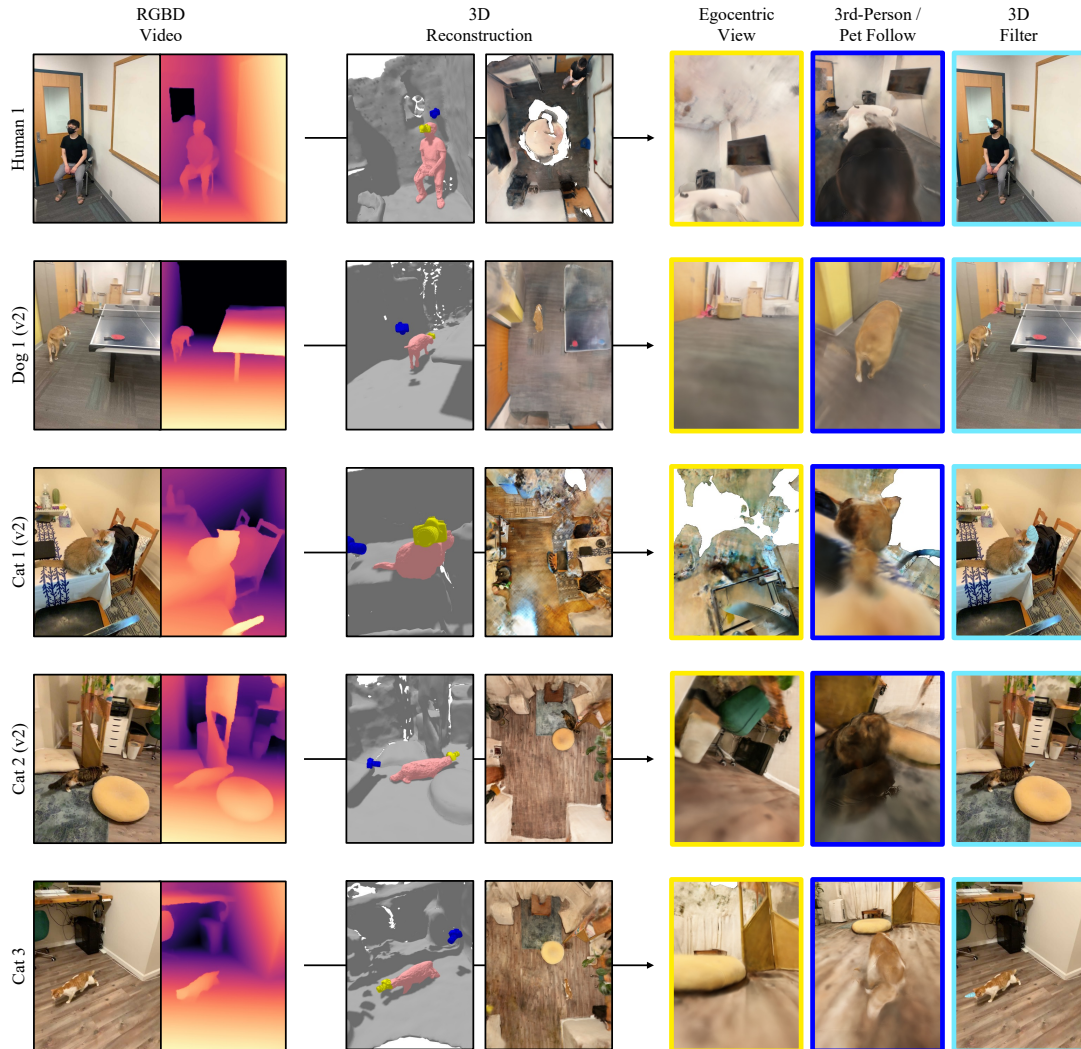


Figure E.1: **Embodied View Synthesis and 3D Filters (Additional Sequences)**. We visualize the 3D reconstructions (rendered from the novel view) and the applications enabled by Total-Recon for the remaining 5 sequences of our RGBD dataset that were not shown in the main paper. The yellow and blue camera meshes in the mesh renderings represent the egocentric and 3rd-person-follow cameras, respectively. To showcase the 3D filter, we attach a sky-blue unicorn horn to the forehead of the foreground object, which is automatically propagated across all frames. [\[Videos\]](#)



Figure E.2: **Object Removal.** Our compositional scene representation enables object removal. We remove the HUMAN and then the PET object from our composite rendering process (Chapter 3.3) and display the resulting renderings. [\[Videos\]](#)

# Appendix F

## Additional Ablation Studies

### F.1 Ablation Study on Depth Supervision

Table F.1: **Ablation Study on Depth Supervision.** Depth supervision improves our model in terms of the visual (LPIPS) and depth (RMS depth error  $\epsilon_{\text{depth}}$ ) metrics, an observation that is consistent with the qualitative results displayed in Figure F.1.

	DOG 1 (626 images)		CAT 1 (641 images)		HUMAN 1 (550 images)		CAT 2 (834 images)		HUMAN - DOG (392 images)		HUMAN - CAT (431 images)		MEAN	
	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$	LPIPS↓	$\epsilon_{\text{depth}}↓$
w/o depth	.307	.317	.489	.719	.348	1.276	.297	.333	.376	1.062	.519	1.056	.379	.815
<b>Ours (w/ depth)</b>	<b>.271</b>	<b>.165</b>	<b>.383</b>	<b>.183</b>	<b>.213</b>	<b>.142</b>	<b>.237</b>	<b>.050</b>	<b>.256</b>	<b>.204</b>	<b>.233</b>	<b>.107</b>	<b>.268</b>	<b>.145</b>

In this chapter, we perform an ablation study on depth supervision for additional sequences in our dataset. Figure F.1 shows that while removing depth supervision from Total-Recon does not significantly deteriorate the training-view RGB renderings, it induces the following failure modes in the *novel-view* 3D reconstructions: (a) *Floating objects*: for the HUMAN 1 & DOG 1, DOG 1, HUMAN 1, and CAT 2 sequences, the foreground objects float above the ground, as evidenced by their shadows. (b) *Objects that sink into the background*: for the HUMAN 2 & CAT 1 sequence, the reconstructed cat is halfway sunk into the ground. (c) *Incorrect occlusions*: for the HUMAN 1 & DOG 1 sequence, the human is incorrectly occluding the dog. (d) *Lower reconstruction quality*: for the HUMAN 2 & CAT 1 sequence, the cat displays lower reconstruction quality, and, for all sequences except HUMAN 1 & DOG 1 and CAT 2, the background exhibits lower reconstruction quality.

## F. Additional Ablation Studies



Figure F.1: **Ablation Study on Depth Supervision.** While removing depth supervision from Total-Recon (COMPLETE MODEL) doesn't significantly hamper the training-view RGB renderings, it induces the following failure modes in the *novel-view* 3D reconstructions. (a) *Floating objects*: for the HUMAN 1 & DOG 1, DOG 1, HUMAN 1, and CAT 2 sequences, the foreground objects float above the ground, as evidenced by their shadows. (b) *Objects that sink into the background*: for the HUMAN 2 & CAT 1 sequence, the reconstructed cat is halfway sunk into the ground. (c) *Incorrect occlusions*: for the HUMAN 1 & DOG 1 sequence, the human is incorrectly occluding the dog. (d) *Lower reconstruction quality*: for the HUMAN 2 & CAT 1 sequence, the cat displays lower reconstruction quality, and, for all sequences except HUMAN 1 & DOG 1 and CAT 2, the background object exhibits lower reconstruction quality. [\[Videos\]](#)

Table F.2: **Ablation Study on Motion Modeling.** Ablating camera-pose optimization (row 2), changing the deformation field (row 3), or removing it (row 4), results in moderate deterioration of the visual and depth metrics. Removing object root-body poses (row 5) prevents our method from converging for most sequences (N/A), as the deformation field alone has to explain global object motion (See Figure 3.1). We perform another ablation that replaces Total-Recon’s neural blend skinning (NBS) deformation field with the more flexible SE(3)-field [30], which does converge but still performs worse than other ablations. These experiments justify our method’s hierarchical motion representation, where object motion is decomposed into global root-body motion and local articulations. †We freeze the camera poses when ablating root-body poses to ensure that the object fields, which are now all defined in the world space, have the same camera poses throughout their separate pretraining processes.

Methods	Optimizes Camera	Deformation Field	Non-Rigid Objects	Root-Body Motion	DOG 1 (626 images)		CAT 1 (641 images)		HUMAN 1 (550 images)		CAT 2 (834 images)		HUMAN - DOG (392 images)		HUMAN - CAT (431 images)		MEAN	
					LPIPS↓	$\epsilon_{\text{depth}}\downarrow$	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$
(1) <b>Ours</b>	✓	NBS	✓	✓	.271	.165	.383	.183	.213	.142	.237	.050	.256	.204	.233	.107	.268	.145
(2) w/o cam. opt.	✗	NBS	✓	✓	.318	.213	.420	.287	.207	.120	.266	.058	.268	.315	.283	.120	.297	.199
(3) w/ SE(3)-field	✓	SE(3)-field	✓	✓	.274	.168	.443	.369	.237	.164	.244	.051	.395	.475	.245	.116	.302	.266
(4) w/o deform. field	✓	None	✗	✓	.297	.169	.420	.381	.249	.150	.254	.056	.298	.216	.285	.154	.300	.211
(5) w/o root-body	✗†	NBS	✓	✗	.379	.266	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
(6) w/o root-body (SE3)	✗	SE(3)-field	✓	✗	.373	.281	.453	.395	.378	.550	.313	.101	.326	.340	.328	.167	.363	.331

These observations are corroborated by Table F.1, which shows that depth supervision significantly improves both the visual and depth metrics of our method over all sequences. Another reason for the large difference in metrics is that the novel-view cameras computed for the non-depth-supervised version may not be entirely accurate. This is because our method optimizes the camera poses during training, meaning that in the absence of depth supervision, the training-view camera poses may converge to a different scale to the ground-truth left-to-right camera transform computed in Appendix C, resulting in slightly misaligned novel-view cameras.

## F.2 Ablation Study on Motion Modeling

In this chapter, we perform ablation studies on Total-Recon’s motion model for a more comprehensive set of design choices than those presented in the main paper. Table F.2 and Figure F.2 show that ablating camera-pose optimization (row 2) worsens the metrics, but does not result in qualitative deterioration of the scene reconstruction. This suggests that the ARKit camera poses used to initialize  $\mathbf{G}_0^t$  (Equation 3.3) are already reasonably accurate. Changing the deformation field from Total-Recon’s

## F. Additional Ablation Studies

neural blend skinning (NBS) function to the SE(3)-field used in HyperNeRF [31] (row 3) further worsens the metrics, most likely due to the minor artifacts that appear in the foreground reconstructions. Removing the deformation field entirely (row 4) also worsens the results, as our method now has to explain each object’s (non-rigid) motion solely via its rigid, root-body poses. As a result, this ablation can only recover coarse object reconstructions that fail to model moving body parts such as limbs.

More notably, Table F.2 shows that ablating object root-body poses (row 5) prevents our method from converging for most sequences. This is a surprising observation since the deformation field should be sufficient to represent all continuous motion, at least in theory. However, when object root-body poses are removed from our method, each object’s canonical model is defined in the *static, world* coordinate frame (Equation 3.1) as opposed to the moving, object-centric coordinate frame (Equation 3.4). Therefore, the deformation field alone has to explain *global* object motion by learning potentially large deviations from the canonical model, significantly complicating optimization. We posit that Total-Recon’s neural blend skinning (NBS) function is too constraining of a deformation field to model global object motion, so we perform another ablation that replaces it with the more flexible SE(3)-field (row 6). This ablation does converge but still performs significantly worse than other converging ablations. Figure F.2 shows that this significant deterioration in metrics is due to failed foreground reconstruction, as evidenced by ghosting artifacts.

These diagnostics justify Total-Recon’s hierarchical motion representation, which decomposes object motion into global root-body motion and local articulations, especially given that row 4 performs better than rows 5 and 6, even though rows 5 and 6 model non-rigid object motion and row 4 does not. They also suggest that conflating these two sources of scene motion is what prevents the baseline methods from reconstructing the highly dynamic objects that appear in our dataset; while row 6 in Table F.2 may not be exactly the same as the baseline method D<sup>2</sup>NeRF (w/ depth), they both exhibit the same ghosting artifacts that are indicative of failed foreground reconstruction.

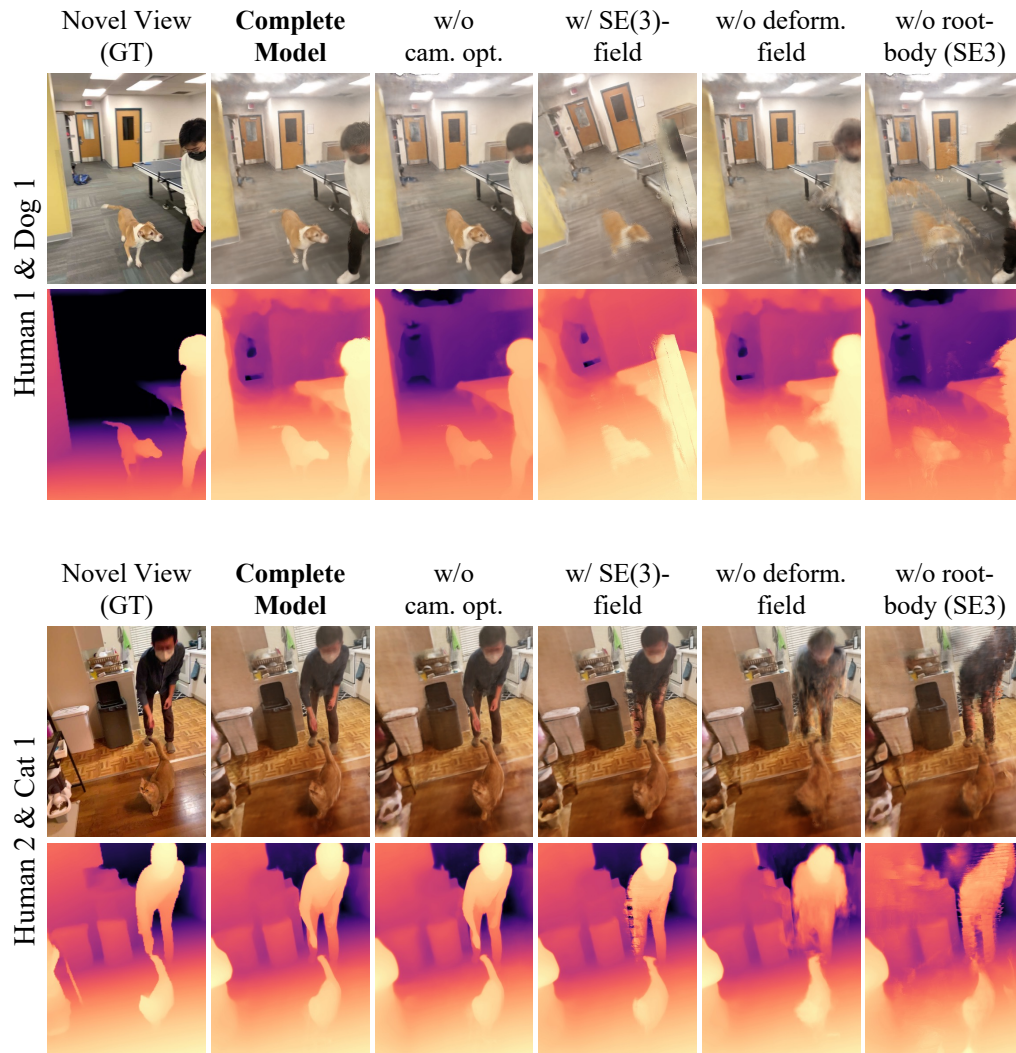


Figure F.2: **Ablation Study on Motion Modeling.** We visualize the novel-view renderings of the ablations in Table F.2. Ablating camera-pose optimization (w/o cam. opt.) does not qualitatively change the scene reconstruction. Changing the deformation field from Total-Recon’s neural blend skinning (NBS) function to an SE(3)-field (w/ SE(3)-field) results in minor artifacts in the foreground reconstruction. Removing the deformation field entirely (w/o deform. field) produces coarse object reconstructions that fail to model moving body parts such as limbs. Removing object root-body poses (w/o root-body (SE3)) breaks foreground reconstruction entirely, as evidenced by the ghosting artifacts indicative of failed foreground reconstruction. These experiments justify our method’s hierarchical motion representation, where object motion is decomposed into global root-body motion and local articulations. [\[Videos\]](#)

## F. Additional Ablation Studies

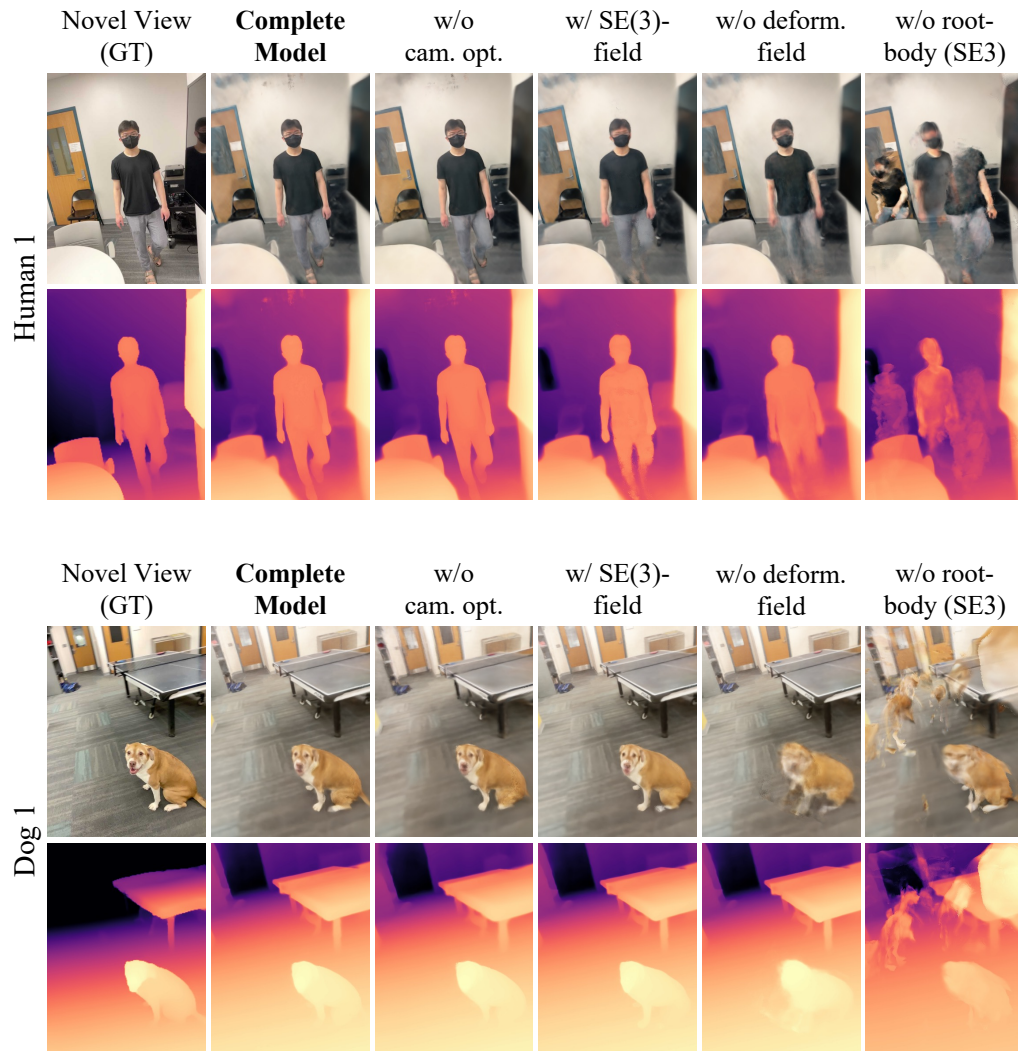


Figure F.2: [Continued] **Ablation Study on Motion Modeling.** We visualize the novel-view renderings of the ablations in Table F.2. Ablating camera-pose optimization (w/o cam. opt.) does not qualitatively change the scene reconstruction. Changing the deformation field from Total-Recon’s neural blend skinning (NBS) function to an SE(3)-field (w/ SE(3)-field) results in minor artifacts in the foreground reconstruction. Removing the deformation field entirely (w/o deform. field) produces coarse object reconstructions that fail to model moving body parts such as limbs. Removing object root-body poses (w/o root-body (SE3)) breaks foreground reconstruction entirely, as evidenced by the ghosting artifacts indicative of failed foreground reconstruction. These experiments justify our method’s hierarchical motion representation, where object motion is decomposed into global root-body motion and local articulations. [\[Videos\]](#)

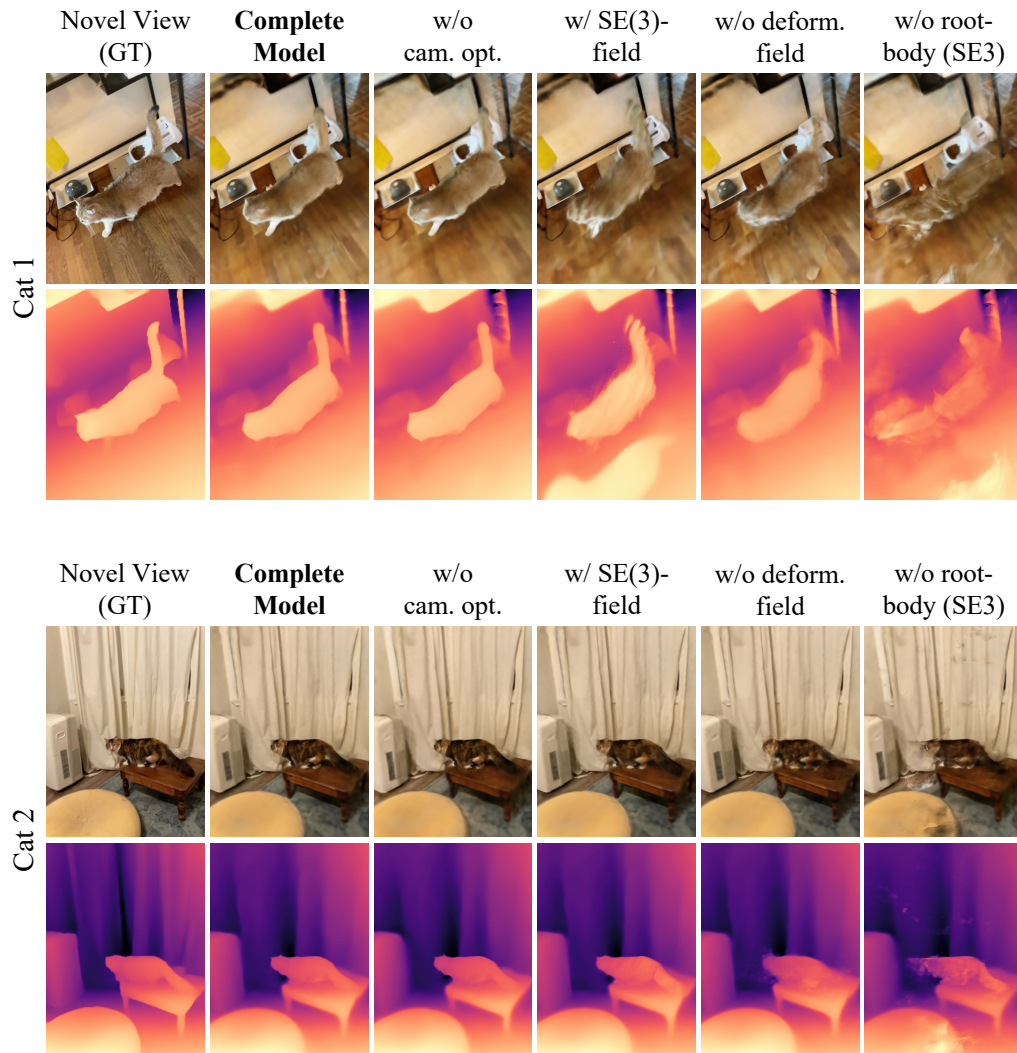


Figure F.2: [Continued] **Ablation Study on Motion Modeling.** We visualize the novel-view renderings of the ablations in Table F.2. Ablating camera-pose optimization (w/o cam. opt.) does not qualitatively change the scene reconstruction. Changing the deformation field from Total-Recon’s neural blend skinning (NBS) function to an SE(3)-field (w/ SE(3)-field) results in minor artifacts in the foreground reconstruction. Removing the deformation field entirely (w/o deform. field) produces coarse object reconstructions that fail to model moving body parts such as limbs. Removing object root-body poses (w/o root-body (SE3)) breaks foreground reconstruction entirely, as evidenced by the ghosting artifacts indicative of failed foreground reconstruction. These experiments justify our method’s hierarchical motion representation, where object motion is decomposed into global root-body motion and local articulations. [\[Videos\]](#)

Table F.3: **Ablation Study on Joint-Finetuning.** Joint fine-tuning generally improves LPIPS but doesn’t always reduce depth error (units: meters). We posit that using a depth signal during pretraining of the individual objects is sufficient for learning a metric model, such that the qualitative improvements induced by joint-finetuning (Figure F.3) are not always reflected in the metrics.

	DOG 1 (626 images)		CAT 1 (641 images)		HUMAN 1 (550 images)		CAT 2 (834 images)		HUMAN - DOG (392 images)		HUMAN - CAT (431 images)		MEAN	
	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$	LPIPS↓	$\epsilon_{\text{depth}}\downarrow$
w/o joint-ft	.273	<b>.162</b>	<b>.379</b>	<b>.126</b>	.231	.154	.244	.053	.259	.216	.235	.113	.273	<b>.138</b>
<b>Ours</b> (w/ joint-ft)	<b>.271</b>	.165	.383	.183	<b>.213</b>	<b>.142</b>	<b>.237</b>	<b>.050</b>	<b>.256</b>	<b>.204</b>	<b>.233</b>	<b>.107</b>	<b>.268</b>	.145

### F.3 Ablation Study on Joint-Finetuning

In Figure F.3, we show that joint-finetuning is indispensable by visualizing its effects on frames without any detected object segmentation masks, which often exist in human-pet interaction videos due to partial occlusions. Since our method does not supervise on frames without segmentation masks during pretraining, the appearance, deformation, and root-body pose of the deformable foreground objects remain uncertain for such frames, as shown in Figure F.3.

For the HUMAN1 & DOG1 sequence, the dog ends up penetrating the human arm; for the HUMAN2 & CAT1 sequence, the cat displays degenerate appearance and lies in front of the human hand rather than behind it. Joint-finetuning resolves these issues as it does not optimize a silhouette loss, enabling our scene representation to be supervised on all frames of the training sequence. By jointly optimizing all objects in the scene, our scene representation learns the correct human-pet interactions by reasoning about occlusions, resulting in improved visual metrics, as shown in Table F.3. Note that joint-finetuning doesn’t always improve the depth error metrics. We posit that the depth supervision during pretraining was sufficient in learning a metric model that the qualitative improvements brought by joint-finetuning are not always reflected in the metrics.



Figure F.3: **Ablation Study on Joint-Finetuning.** Joint-finetuning enables our method to learn the correct human-pet interactions, particularly for frames without any detected object segmentation masks.

# Appendix G

## Societal Impact

We have demonstrated that our method can holistically reconstruct a dynamic scene containing multiple deformable objects, such as humans and pets - all from a single RGBD video captured from a commodity consumer device. We believe that a truly holistic reconstruction of the background geometry, each moving object, its own deformation, and camera pose would enable a number of new applications ranging from augmented reality to asset generation for virtual worlds, especially given the ubiquity of consumer-grade RGBD sensors.

However, the reconstruction capabilities of our method could be a double-edged sword; the very ease with which one could reconstruct a realistic 3D human model from nothing but a casually captured RGBD video poses potential privacy concerns. For instance, one could extract sensitive personal information such as height and other body measurements from a metric human model reconstructed with our method. In terms of appearance synthesis, our method poses similar types of risks as Deepfakes pose to society, especially given that the deformable object model used in our method is animatable (*i.e.*, user-drivable) [59]. An important future direction of research that needs to accompany 3D reconstruction research would therefore be methods of distinguishing photorealistic rendered videos from genuine content.

# Bibliography

- [1] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. [2.3](#)
- [2] Aljaz Bozic, Michael Zollhofer, Christian Theobalt, and Matthias Nießner. Deep-deform: Learning non-rigid rgb-d reconstruction with semi-supervised data. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. [2.3](#)
- [3] Aljaz Bozic, Pablo Palafox, Michael Zollhofer, Justus Thies, Angela Dai, and Matthias Nießner. Neural deformation graphs for globally-consistent non-rigid reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. [2.3](#)
- [4] Hongrui Cai, Wanquan Feng, Xuetao Feng, Yan Wang, and Juyong Zhang. Neural surface reconstruction of dynamic scenes with monocular rgb-d camera. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. [1.1](#), [2.3](#)
- [5] 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)*, 2022. [2.1](#)
- [6] Alena Denisova and Paul Cairns. First person vs. third person perspective in digital games: Do player preferences affect immersion? In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, page 145–148, 2015. [1](#)
- [7] Mingsong Dou, Jonathan Taylor, Henry Fuchs, Andrew Fitzgibbon, and Shahram Izadi. 3d scanning deformable objects with a single rgb-d sensor. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. [2.3](#)
- [8] Chen Gao, Ayush Saraf, Johannes Kopf, and Jia-Bin Huang. Dynamic view synthesis from dynamic monocular video. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. [1.1](#), [2.2](#)
- [9] Alec Jacobson, Zhigang Deng, Ladislav Kavan, and J. P. Lewis. Skinning: Real-time shape deformation. In *ACM SIGGRAPH 2014 Courses*, SIGGRAPH '14.

- Association for Computing Machinery, 2014. [3.2](#)
- [10] Yoonwoo Jeong, Seokjun Ahn, Christopher Choy, Anima Anandkumar, Minsu Cho, and Jaesik Park. Self-calibrating neural radiance fields. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. [2.1](#)
- [11] Zhang Jiakai, Liu Xinhang, Ye Xinyi, Zhao Fuqiang, Zhang Yanshun, Wu Minye, Zhang Yingliang, Xu Lan, and Yu Jingyi. Editable free-viewpoint video using a layered neural representation. In *ACM SIGGRAPH*, 2021. [2.2](#), [3.2](#)
- [12] Wei Jiang, Kwang Moo Yi, Golnoosh Samei, Oncel Tuzel, and Anurag Ranjan. Neuman: Neural human radiance field from a single video. In *European Conference on Computer Vision (ECCV)*, 2022. [2.1](#), [2.2](#)
- [13] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. [4.1](#), [5.1](#)
- [14] Abhijit Kundu, Kyle Genova, Xiaoqi Yin, Alireza Fathi, Caroline Pantofaru, Leonidas J. Guibas, Andrea Tagliasacchi, Frank Dellaert, and Thomas Funkhouser. Panoptic neural fields: A semantic object-aware neural scene representation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. [1.1](#), [2.1](#), [2.1](#)
- [15] Ruilong Li, Julian Tanke, Minh Vo, Michael Zollhofer, Jurgen Gall, Angjoo Kanazawa, and Christoph Lassner. Tava: Template-free animatable volumetric actors. In *European Conference on Computer Vision (ECCV)*, 2022. [2.2](#)
- [16] Tianye Li, Mira Slavcheva, Michael Zollhöfer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, and Zhaoyang Lv. Neural 3d video synthesis from multi-view video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. [2.2](#)
- [17] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. [1.1](#), [2.2](#)
- [18] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. [2.1](#)
- [19] Lingjie Liu, Marc Habermann, Viktor Rudnev, Kripasindhu Sarkar, Jiatao Gu, and Christian Theobalt. Neural actor: Neural free-view synthesis of human actors with pose control. *ACM SIGGRAPH Asia*, 2021. [2.2](#)
- [20] Yu-Lun Liu, Chen Gao, Andreas Meuleman, Hung-Yu Tseng, Ayush Saraf, Changil Kim, Yung-Yu Chuang, Johannes Kopf, and Jia-Bin Huang. Robust

- dynamic radiance fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. [2.4](#)
- [21] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. [2.1](#), [3.2](#)
- [22] Quan Meng, Anpei Chen, Haimin Luo, Minye Wu, Hao Su, Lan Xu, Xuming He, and Jingyi Yu. Gnerf: Gan-based neural radiance field without posed camera. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. [2.1](#)
- [23] 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. [1.1](#), [2.1](#), [3.2](#)
- [24] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE International Symposium on Mixed and Augmented Reality*, pages 127–136. IEEE, 2011. [2.3](#)
- [25] Richard A Newcombe, Dieter Fox, and Steven M Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. [2.3](#)
- [26] 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. [3.3](#)
- [27] Atsuhiko Noguchi, Xiao Sun, Stephen Lin, and Tatsuya Harada. Neural articulated radiance field. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. [2.2](#)
- [28] Julian Ost, Fahim Mannan, Nils Thuerey, Julian Knodt, and Felix Heide. Neural scene graphs for dynamic scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. [1.1](#), [2.1](#)
- [29] Kemal Egemen Ozden, Kurt Cornelis, Luc Van Eycken, and Luc Van Gool. Reconstructing 3d trajectories of independently moving objects using generic constraints. *Comput. Vis. Image Underst.*, 96(3):453–471, 2004. ISSN 1077-3142. [2.1](#)
- [30] 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 *IEEE International Conference on Computer Vision (ICCV)*, 2021. [\(document\)](#), [1.1](#), [1.3](#), [2.2](#), [4.2](#), [F.2](#)

- [31] 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. *ACM Transactions on Graphics (TOG)*, 40(6), 2021. ([document](#)), [1.1](#), [1.3](#), [2.2](#), [2.1](#), [3.1](#), [3.1](#), [3.2](#), [4.2](#), [4.4](#), [4.1](#), [4.3](#), [D.1](#), [D.1](#), [D.2](#), [D.1](#), [D.1](#), [F.2](#)
- [32] Georgios Pavlakos\*, Ethan Weber\*, , Matthew Tancik, and Angjoo Kanazawa. The one where they reconstructed 3d humans and environments in tv shows. In *European Conference on Computer Vision (ECCV)*, 2022. [2.2](#)
- [33] Sida Peng, Junting Dong, Qianqian Wang, Shangzhan Zhang, Qing Shuai, Xiaowei Zhou, and Hujun Bao. Animatable neural radiance fields for modeling dynamic human bodies. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. [2.2](#)
- [34] Sida Peng, Yuanqing Zhang, Yinghao Xu, Qianqian Wang, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. [2.2](#)
- [35] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-NeRF: Neural Radiance Fields for Dynamic Scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. [1.1](#), [2.2](#)
- [36] Konstantinos Rematas, Andrew Liu, Pratul P. Srinivasan, Jonathan T. Barron, Andrea Tagliasacchi, Thomas Funkhouser, and Vittorio Ferrari. Urban radiance fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. [2.3](#)
- [37] Zhongzheng Ren, Xiaoming Zhao, and Alex Schwing. Class-agnostic reconstruction of dynamic objects from videos. *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. [2.3](#)
- [38] 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)*, 2022. [2.1](#)
- [39] Richard Rouse. What’s your perspective? *ACM SIGGRAPH Computer Graphics*, 33(3):9–12, August 1999. [1](#)
- [40] Johannes L. Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. [4.4](#), [D.1](#)
- [41] Qing Shuai, Chen Geng, Qi Fang, Sida Peng, Wenhao Shen, Xiaowei Zhou, and Hujun Bao. Novel view synthesis of human interactions from sparse multi-view videos. In *SIGGRAPH Conference Proceedings*, 2022. [2.2](#)

- [42] Marek Simonik. Record3d. <https://record3d.app>. 4.2
- [43] Miroslava Slavcheva, Maximilian Baust, Daniel Cremers, and Slobodan Ilic. Killingfusion: Non-rigid 3d reconstruction without correspondences. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2.3
- [44] Miroslava Slavcheva, Maximilian Baust, and Slobodan Ilic. Sobolevfusion: 3d reconstruction of scenes undergoing free non-rigid motion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2.3
- [45] Karl Stelzner, Kristian Kersting, and Adam R Kosiorek. Decomposing 3d scenes into objects via unsupervised volume segmentation. *arXiv preprint arXiv:2104.01148*, 2021. 3.3, 3.3.1
- [46] Shih-Yang Su, Frank Yu, Michael Zollhöfer, and Helge Rhodin. A-nerf: Articulated neural radiance fields for learning human shape, appearance, and pose. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 2.2
- [47] Unity Technologies. 3rd person follow. <https://docs.unity3d.com/Packages/com.unity.cinemachine@2.8/manual/Cinemachine3rdPersonFollow.html>, 2022. 1
- [48] Edgar Tretschk, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Christoph Lassner, and Christian Theobalt. Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. 1.1, 2.2
- [49] Barbara Tversky. *Spatial Cognition Embodied and Situated*. Cambridge, 2008. 1
- [50] Chaoyang Wang, Ben Eckart, Simon Lucey, and Orazio Gallo. Neural trajectory fields for dynamic novel view synthesis. *arXiv preprint arXiv:2105.05994*, 2021. 1.1, 2.2
- [51] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf-: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021. 2.1
- [52] Tianhao Wu, Fangcheng Zhong, Andrea Tagliasacchi, Forrester Cole, and Cengiz Oztireli. D<sup>2</sup>nerf: Self-supervised decoupling of dynamic and static objects from a monocular video. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. (document), 1.1, 1.3, 2.2, 2.1, 3.1, 3.1, 3.2, 4.2, 4.4, 4.1, 4.3, D.1, D.1, D.2, D.1, D.1
- [53] Wenqi Xian, Jia-Bin Huang, Johannes Kopf, and Changil Kim. Space-time neural irradiance fields for free-viewpoint video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1.1, 2.2
- [54] Jing Xiao, Jinxiang Chai, and Takeo Kanade. A closed-form solution to non-rigid shape and motion recovery. *International Journal of Computer Vision (IJCV)*,

- 67(2):233–246, April 2006. [2.1](#)
- [55] Zhiwen Yan, Chen Li, and Gim Hee Lee. Nerf-ds: Neural radiance fields for dynamic specular objects. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. [2.4](#)
- [56] Gengshan Yang and Deva Ramanan. Volumetric correspondence networks for optical flow. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. [3.3.2](#)
- [57] Gengshan Yang, Deqing Sun, Varun Jampani, Daniel Vlasic, Forrester Cole, Huiwen Chang, Deva Ramanan, William T Freeman, and Ce Liu. LASR: Learning articulated shape reconstruction from a monocular video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. [1.1](#)
- [58] Gengshan Yang, Deqing Sun, Varun Jampani, Daniel Vlasic, Forrester Cole, Ce Liu, and Deva Ramanan. Viser: Video-specific surface embeddings for articulated 3d shape reconstruction. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. [1.1](#)
- [59] Gengshan Yang, Minh Vo, Natalia Neverova, Deva Ramanan, Andrea Vedaldi, and Hanbyul Joo. Banmo: Building animatable 3d neural models from many casual videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. [1.1](#), [2.1](#), [3.2](#), [3.2](#), [3.3.2](#), [4.1](#), [4.5](#), [5.1](#), [B.1](#), [B.2](#), [B.3](#), [B.3](#), [G](#)
- [60] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. [3.2](#), [B.3](#)
- [61] Vickie Ye, Georgios Pavlakos, Jitendra Malik, and Angjoo Kanazawa. Decoupling human and camera motion from videos in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. [2.1](#), [2.4](#)
- [62] 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. [4.4](#)
- [63] Xiaoshuai Zhang, Abhijit Kundu, Thomas Funkhouser, Leonidas Guibas, Hao Su, and Kyle Genova. Nerflets: Local radiance fields for efficient structure-aware 3d scene representation from 2d supervision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. [2.4](#)