

# DeGauss: Dynamic-Static Decomposition with Gaussian Splatting for Distractor-free 3D Reconstruction

## Supplementary Material

### A. Detailed Loss Function Formulation

Loss function design is important to maintain the balance of the dynamic-static decomposition task. For example, directly adding SSIM loss could improve the overall reconstruction quality but often leads to a larger gradient magnitude in the static region with fine details. As a result, this often leads to the over-expressiveness of foreground gaussians that undesirably models the static fine details. As the densification of process of gaussian is controlled by loss gradient magnitude, we propose a loss function that comprises two components  $\mathcal{L}_{\text{main}}$  and  $\mathcal{L}_{\text{uti}}$  to decouple parameter updates and the adaptive densification process. While both  $\mathcal{L}_{\text{main}}$  and  $\mathcal{L}_{\text{uti}}$  contribute to the background and foreground gaussian feature updates, only the gradient of  $\mathcal{L}_{\text{main}}$  is used for the densification process. The main loss component is defined as:

$$\mathcal{L}_{\text{main}} = \mathcal{L}_1 + \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{diversity}} + \mathcal{L}_f + \mathcal{L}_b + \mathcal{L}_{\text{depth}}, \quad (1)$$

where

$$\mathcal{L}_1 = \|\hat{\mathbf{C}} - \mathbf{C}_{gt}\|_1$$

denotes the  $\mathcal{L}_1$  loss between the fully composed rendered image  $\hat{\mathbf{C}}$  and the ground truth image  $\mathbf{C}_{gt}$ . The regularization loss  $\mathcal{L}_{\text{reg}}$  enforces time smoothness and k-plane total variations, following the settings in [2, 4, 5, 22]. Furthermore, to encourage a higher foreground probability  $\mathbf{P}_f$  for the foreground render  $\hat{\mathbf{C}}_f$  at region which exhibits significant structural differences relative to the detached background render  $\bar{\mathbf{C}}_b$ , similar to [15], we employ a diversity loss based on the structural component of the SSIM loss :

$$\mathcal{L}_{\text{diversity}}(\mathbf{C}_f, \bar{\mathbf{C}}_b) = \mathbb{1}_{\{\mathbf{P}_f > \mathbf{P}_\tau\}} \cdot \frac{\sigma_{\mathbf{C}_f \bar{\mathbf{C}}_b} + c_3}{\sigma_{\hat{\mathbf{C}}_f} \sigma_{\bar{\mathbf{C}}_b} + c_3}, \quad (2)$$

where  $\mathbb{1}_{\{\mathbf{P}_f > \mathbf{P}_\tau\}}$  is the indicator function and  $\mathbf{P}_\tau$  is the probability threshold,  $\sigma$  denotes the variance, and  $c_3$  is a constant to stabilize the loss. To refine the regions assigned to the background and foreground, we further introduce updating losses  $\mathcal{L}_f$  and  $\mathcal{L}_b$ , defined as:

$$\begin{aligned} \mathcal{L}_e &= \mathbb{1}_{\{\mathbf{P}_e > \mathbf{P}_\tau\}} \left( \|\hat{\mathbf{C}}_e - \mathbf{C}_{gt}\|_1 \right. \\ &\quad \left. + 0.1 \mathcal{L}_{\text{SSIM}}(\hat{\mathbf{C}}_e, \mathbf{C}_{gt}) \right), \quad e \in \{f, b\}. \end{aligned} \quad (3)$$

This loss term is scaled down 4 times compared to the  $\mathcal{L}_1$  loss between foreground render and background render to suppress their contribution to gaussian densification

process. Additionally, to softly regularize the spatial relationship between foreground-background gaussians and encourage a distractor-free background reconstruction, we introduce depth-related loss  $\mathcal{L}_{\text{depth}}$ , defined as:

$$\mathcal{L}_{\text{depth}} = \mathcal{L}_{\text{smooth}} + \mathcal{L}_{\text{sep}}, \quad (4)$$

where  $\mathcal{L}_{\text{smooth}}$  is an edge-aware total variation loss [7, 21] that encourages smooth depth predictions for static background Gaussians, particularly in regions with small color variance:

$$\begin{aligned} \mathcal{L}_{\text{smooth}} = \frac{1}{N} \sum_{i,j} & \left( |D_{b_{ij}} - D_{b_{ij+1}}| \cdot e^{-\|\mathbf{C}_{gt_{ij}} - \mathbf{C}_{gt_{ij+1}}\|_1} \right. \\ & \left. + |D_{b_{ij}} - D_{b_{ij+1}}| \cdot e^{-\|\mathbf{C}_{gt_{ij}} - \mathbf{C}_{gt_{ij+1}}\|_1} \right). \end{aligned} \quad (5)$$

Here,  $N$  denotes the total number of pixels, and  $D_{b_{ij}}$  represents the depth value at pixel  $(i, j)$  of the rendered background depth image, normalized by the scene bounding box to account for the scale ambiguity of colmap[17] reconstruction. Moreover, the depth separation loss is defined as:

$$\mathcal{L}_{\text{sep}} = \mathbb{1}_{\{\mathbf{P}_f > \mathbf{P}_\tau\}} \left( \sum_{i,j} \max(D_{f_{ij}} - D_{b_{ij}}, 0) \right) \quad (6)$$

$$+ \mathbb{1}_{\{\mathbf{P}_b > \mathbf{P}_\tau\}} \left( \sum_{i,j} \max(D_{f_{ij}} - D_{b_{ij}}, 0) \right). \quad (7)$$

The first loss term encourages the rendered foreground to be positioned closer to the camera, thereby preserving occlusion relationships with the static background. In addition, the second loss term pushes the utility gaussians with low foreground render contributions to be further away from the camera to prevent their presence during novel view rendering. This term efficiently regularizes floaters for datasets with sparse fixed camera input as Neu3D dataset [11].

$\mathcal{L}_{\text{uti}}$  is introduced to stabilize training, promote fine reconstruction, and enhance separation without contributing to the densification process:

$$\mathcal{L}_{\text{uti}} = \mathcal{L}_{\text{SSIM}}(\hat{\mathbf{C}}, \mathbf{C}_{gt}) + \mathcal{L}_{\text{entropy}} + \mathcal{L}_{\text{brightness}} + \mathcal{L}_s. \quad (8)$$

The SSIM loss  $\mathcal{L}_{\text{SSIM}}$ , computed between the composed render  $\hat{\mathbf{C}}$  and the ground truth image  $\mathbf{C}_{gt}$ , improves reconstruction quality of fine detailed region; Additionally, the entropy loss is defined as a binary cross-entropy loss that encourages the foreground probability  $\mathbf{P}_f$  to converge toward either 0 or 1:

$$\mathcal{L}_{\text{entropy}} = - \sum_N \mathbf{P}_f \cdot \log(\mathbf{P}_f). \quad (9)$$

076 Furthermore, to promote the update of brightness control  
 077 mask  $\hat{\mathbf{B}}$  in the early stage, we define the brightness loss as:

$$\mathcal{L}_{\text{brightness}} = \alpha \cdot \|\hat{\mathbf{B}} * \bar{\mathbf{C}}_b - \mathbf{C}_{gt}\|_1 + (1 - \alpha) \cdot \|\hat{\mathbf{B}} - \mathbf{I}\|_1, \quad (10)$$

079 where  $\bar{\mathbf{C}}_b$  denotes the novel view rendered from the back-  
 080 ground branch (detached from gradient propagation), and  
 081  $\alpha$  is a coefficient that increases linearly with training iterations.  
 082 The first term ensures an accurate prediction of the  
 083 brightness control mask, while the second term acts as a reg-  
 084 ularizer. Finally, the scale loss  $\mathcal{L}_s$  penalizes spiky Gaussians,  
 085 as defined in [23].

086 The loss coefficients set to balance each loss term is set  
 087 to 4 for main  $\mathcal{L}_1$  loss, 1 for  $\mathcal{L}_f$  and  $\mathcal{L}_b$ , 0.01 for  $\mathcal{L}_{\text{entropy}}$  and  
 088 0.1 for the rest components.

## 089 **B. Additional Pruning for Dynamic Scene mod- 090 eling**

091 In our setting, there are utility gaussians that do not con-  
 092 tribute to dynamic rendering but are utilized for probabilistic  
 093 mask and brightness control mask rasterization. Therefore,  
 094 we could optionally further control the number of utility  
 095 gaussians with foreground visibility-based pruning.

096 Specifically, a Gaussian is discarded if the maximum  
 097 value of the product of its opacity  $\sigma$  and its foreground  
 098 mask elements  $m'_f$ —computed across all input views and  
 099 timestamps—falls below a predefined threshold  $\tau$ . This  
 100 procedure effectively eliminates Gaussians that contribute  
 101 negligibly to the overall dynamic representation for dynamic  
 102 scene modeling tasks.

## 103 **C. Detailed Dataset preparation**

104 **Aria Glass Recordings** [1, 12, 13] feature egocentric video  
 105 captured at 20-30 FPS, encompassing intensive human-  
 106 object, human-scene, and human-human interactions, along  
 107 with challenges such as rapid camera motion and motion  
 108 blur. We used NerfStudio [19] to preprocess fisheye camera  
 109 frames from Project Aria, with camera mask and distortion  
 110 parameters, and camera vignetting mask provided [6, 9]. The  
 111 original resolution of an Aria frame is  $1415 \times 1415$  after  
 112 fisheye undistortion. To balance rendering quality and speed  
 113 and avoid excessive training time for the nerf baseline neu-  
 114 ralldiff, we downsample the frames to  $707 \times 707$ . The first  
 115 50 frames of each sequence are omitted to allow the camera  
 116 stream to stabilize.

117 **Epic-field Dataset** [20] builds upon the EPIC-Kitchen  
 118 dataset [3], which comprises long egocentric video record-  
 119 ings of human activities in a kitchen recorded at 50 FPS. We  
 120 use the point clouds and camera poses provided in [20]. To  
 121 keep a consistent frame rate with aria recordings, we take  
 122 testing segments of 10,000 consecutive frames and down-  
 123 sample by 2, which leads to 5000 frames in the end.

**NerF On-the-Go Dataset** [15] we prepare the Nerf  
 124 On-the-go dataset following the setting of SpotlessS-  
 125 plats [16]. The dataset was originally captured with  
 126 high-resolution images and downsampled 4 times for patio set  
 127 and 8 times for others, following [10, 15, 16]. We follow the  
 128 camera undistortion setting of [16]. **Neu3D Dataset** [11]  
 129 Following the setup of [22], the resolution is downsampled to  
 130  $1352 \times 1014$ . We compute the camera poses and generate  
 131 a dense point cloud using COLMAP [17, 18] based on the  
 132 first frame of each video.

**HyperNerf dataset** As noted in [8], the camera poses are  
 133 considerably inaccurate, which diminishes the reliability of  
 134 quantitative comparisons. Therefore, we run colmap[17] to  
 135 recompute camera poses and focus primarily on qualitative  
 136 visualizations for this dataset.

## 137 **D. Implementation Details**

**Initialization** During initialization, for the background  
 141 branch, Gaussians are derived from point clouds generated  
 142 using COLMAP [18, 18] or from sparse perception point  
 143 clouds provided by the ARIA project [9]. The scene bound-  
 144 ary is determined based on the range of the background  
 145 points, with an additional padding equal to 0.3 times the  
 146 diagonal length of the camera trajectory. Foreground Gaus-  
 147 sians are initialized from randomly generated points within  
 148 this 3D scene boundary.

**Coarse Training Stage** In the coarse training stage, we  
 150 disable the deformation module in the foreground branch and  
 151 train both the foreground and background models for 1,000  
 152 iterations. For longer sequences (containing thousands of  
 153 frames), the number of coarse training iterations is adjusted  
 154 so that each image is processed exactly once. During this  
 155 stage, the standard color loss  $\mathcal{L}_1$  in Equation 1 is replaced  
 156 by a combination of foreground and background losses:

$$\mathcal{L}_{\text{coarse}} = \|(\mathbf{P}_f * \mathbf{C}_f + \mathbf{P}_b * \hat{\mathbf{B}} * \bar{\mathbf{C}}_b) - 0.9 \mathbf{C}_{gt}\|_1 + \|\mathbf{C}_b - \mathbf{C}_{gt}\|_1. \quad (158)$$

The discount factor of 0.9 applied to the ground truth further  
 159 regularizes the expressiveness of the foreground Gaussians,  
 160 particularly in featureless regions (e.g., walls) that are often  
 161 associated with poor structural reconstruction in COLMAP.

**Fine Training Stage** In the fine training stage, we jointly  
 163 optimize the foreground and background branches. For short  
 164 video clips and image collections, training is performed for  
 165 20,000 iterations; for longer video clips, training extends to  
 166 120,000 iterations.

**Parameters set up** We generally follow the parameter set up  
 168 in [22]. With the basic resolution of Hexplane set to 256 for  
 169 egocentric recordings and 64 for other scenes, upsampled by  
 170 2 and 4. The learning rate of set to Hexplane is set to  $6 \times 10^{-4}$   
 171 and decays to  $2 \times 10^{-5}$  during training. The deformation  
 172 learning rate is set to  $1.6 \times 10^{-4}$  and decays to  $1.6 \times 10^{-5}$

174 during training. The deformation learning rate for mask up-  
 175 date is set to  $1.6 \times 10^{-5}$  and decays to  $1.6 \times 10^{-6}$  during  
 176 training. Generally, the batch size is set to 2 as [22]. For low-  
 177 resolution image collections in [15], we set the batch size to  
 178 4 for the dynamic branch and additionally accumulate the  
 179 update of 4 batches for the background gaussians to account  
 180 for the low resolution and loose temporal correlations.

181 **Baseline Evaluation** For 4DGS [22] experiments, we follow  
 182 the instruction of their official repo and dataset preparation.  
 183 For the HyperNerf [14] dataset, we use the colmap calculated  
 184 camera poses and point cloud for initialization, the same as  
 185 our method. For experiments with 3DGS [24] and [16], we  
 186 use the official repo of [16] and follow their setup. For the  
 187 Nerf on-the-go dataset [15], EPIC-Field dataset, we use stan-  
 188 dard colmap initialization. For Aria sequences, the sensor  
 189 perception point clouds are without color, which leads to  
 190 unstable initialization for [16]. Therefore, we triangulate  
 191 with COLMAP [17] using camera poses provided by [9] to  
 192 obtain colored point cloud to better evaluate this method.

## 193 E. Additional Visualizations

194 We show additional visualizations in Fig. 1, Fig. 2 and Fig. 3.  
 195

## 196 F. Discussion

197 **Dynamic-Static Elements.** While our method effectively  
 198 handles semi-static objects, there is an inherent ambiguity  
 199 when certain subjects—like people or objects—remain static  
 200 most of the time in long video recordings. In this work,  
 201 we focus on a self-supervised approach that ensures robust  
 202 decomposition across diverse scenarios. For specific down-  
 203 stream applications, it may be beneficial to integrate our  
 204 method with additional semantic information for even more  
 205 accurate separation.

206 **Camera Pose Optimization.** Our approach generally as-  
 207 sumes reasonably accurate camera poses to facilitate static-  
 208 dynamic decomposition. Nonetheless, we observe that even  
 209 when camera poses are suboptimal (as in HyperNeRF [14]),  
 210 our method can still separate dynamic and static regions. An  
 211 interesting direction for future research is to leverage our  
 212 predicted masks to optimize camera poses based on regions  
 213 identified as static.

214 **Efficient Dynamic Scene Representation.** In this work,  
 215 we showed that we could achieve high-quality and efficient  
 216 dynamic representation by a decoupled dynamic-static gaus-  
 217 sian representation, which largely reduces the number of  
 218 gaussian in the time-consuming deformation step. However,  
 219 as there are numerous utility gaussian to model probabilistic  
 220 and brightness control mask. Exploring ways to minimize  
 221 this overhead could be a promising avenue for future work.

## References

- [1] Prithviraj Banerjee, Sindi Shkodrani, Pierre Moulon, Shreyas Hampali, Fan Zhang, Jade Fountain, Edward Miller, Selen Basol, Richard Newcombe, Robert Wang, et al. Introducing hot3d: An egocentric dataset for 3d hand and object tracking. *arXiv preprint arXiv:2406.09598*, 2024. **2**
- [2] Ang Cao and Justin Johnson. Hexplane: A fast representation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 130–141, 2023. **1**
- [3] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):4125–4141, 2020. **2**
- [4] Jiemin Fang, Taoran Yi, Xinggang Wang, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Matthias Nießner, and Qi Tian. Fast dynamic radiance fields with time-aware neural voxels. In *SIGGRAPH Asia 2022 Conference Papers*, 2022. **1**
- [5] Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12479–12488, 2023. **1**
- [6] Qiao Gu, Zhaoyang Lv, Duncan Frost, Simon Green, Julian Straub, and Chris Sweeney. Egolifter: Open-world 3d segmentation for egocentric perception. In *European Conference on Computer Vision*, pages 382–400. Springer, 2025. **2**
- [7] Philipp Heise, Sebastian Klose, Brian Jensen, and Alois Knoll. Pm-huber: Patchmatch with huber regularization for stereo matching. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2360–2367, 2013. **1**
- [8] Yi-Hua Huang, Yang-Tian Sun, Ziyi Yang, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes. *arXiv preprint arXiv:2312.14937*, 2023. **2**
- [9] Selcuk Karakas, Pierre Moulon, Wenqi Zhang, Nan Yang, Julian Straub, Lingni Ma, Zhaoyang Lv, Elizabeth Argall, Georges Berenger, Tanner Schmidt, Kiran Somasundaram, Vijay Baiyya, Philippe Bouffetroy, Geof Sawaya, Yang Lou, Eric Huang, Tianwei Shen, David Caruso, Bilal Souti, Chris Sweeney, Jeff Meissner, Edward Miller, and Richard Newcombe. Aria data tools. [https://github.com/facebookresearch/aria\\_data\\_tools](https://github.com/facebookresearch/aria_data_tools), 2022. **2, 3**
- [10] Jonas Kulhanek, Songyou Peng, Zuzana Kukelova, Marc Pollefeys, and Torsten Sattler. Wildgaussians: 3d gaussian splatting in the wild. *NeurIPS*, 2024. **2, 5**
- [11] Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, et al. Neural 3d video synthesis from multi-view video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5521–5531, 2022. **1, 2, 4**
- [12] Zhaoyang Lv, Nicholas Charron, Pierre Moulon, Alexander Gamino, Cheng Peng, Chris Sweeney, Edward Miller, Huix-

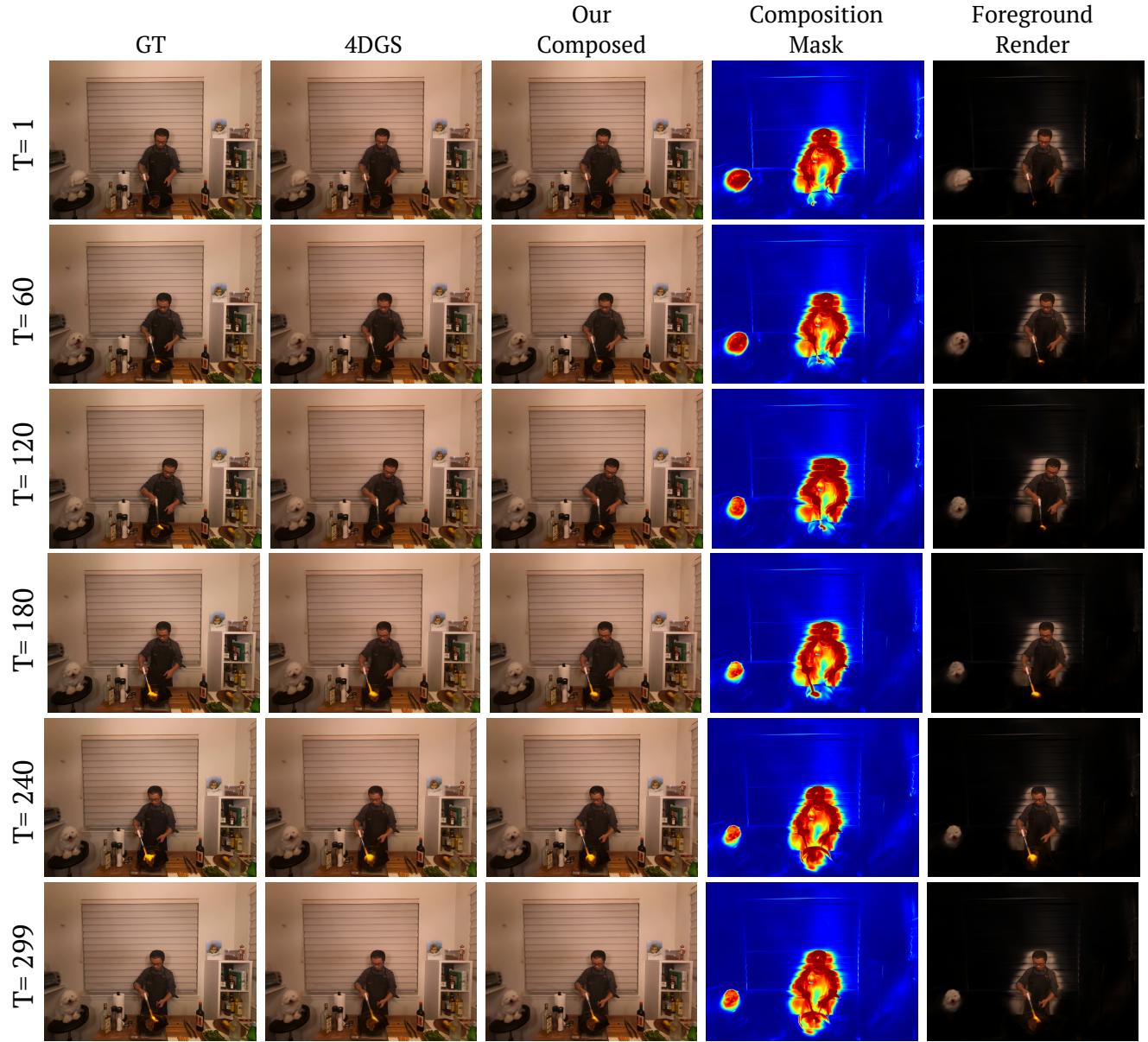


Figure 1. Visualization for Flame Steak Sequence of Neu3D [11] dataset. Our method achieves accurate dynamic-static decomposition with high reconstruction quality.

- 278 uan Tang, Jeff Meissner, Jing Dong, et al. Aria everyday  
279 activities dataset. *arXiv preprint arXiv:2402.13349*, 2024. 2
- 280 [13] Xiaqing Pan, Nicholas Charron, Yongqian Yang, Scott Pe-  
281 ters, Thomas Whelan, Chen Kong, Omkar Parkhi, Richard  
282 Newcombe, and Yuheng Carl Ren. Aria digital twin: A new  
283 benchmark dataset for egocentric 3d machine perception. In  
284 *Proceedings of the IEEE/CVF International Conference on*  
285 *Computer Vision*, pages 20133–20143, 2023. 2
- 286 [14] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T  
287 Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-  
288 Brualla, and Steven M Seitz. Hypernerf: A higher-  
289 dimensional representation for topologically varying neural  
290 radiance fields. *arXiv preprint arXiv:2106.13228*, 2021. 3, 5
- 291 [15] Weining Ren, Zihan Zhu, Boyang Sun, Jiaqi Chen, Marc  
292 Pollefeys, and Songyou Peng. Nerf on-the-go: Exploiting  
293 uncertainty for distractor-free nerfs in the wild. In *Pro-  
294 ceedings of the IEEE/CVF Conference on Computer Vision and*  
295 *Pattern Recognition*, pages 8931–8940, 2024. 1, 2, 3
- 296 [16] Sara Sabour, Lily Goli, George Kopanas, Mark Matthews,  
297 Dmitry Lagun, Leonidas Guibas, Alec Jacobson, David J  
298 Fleet, and Andrea Tagliasacchi. Spotlessplats: Ignor-  
299 ing distractors in 3d gaussian splatting. *arXiv preprint*  
300 *arXiv:2406.20055*, 2024. 2, 3, 5
- 301 [17] Johannes Lutz Schönberger and Jan-Michael Frahm.

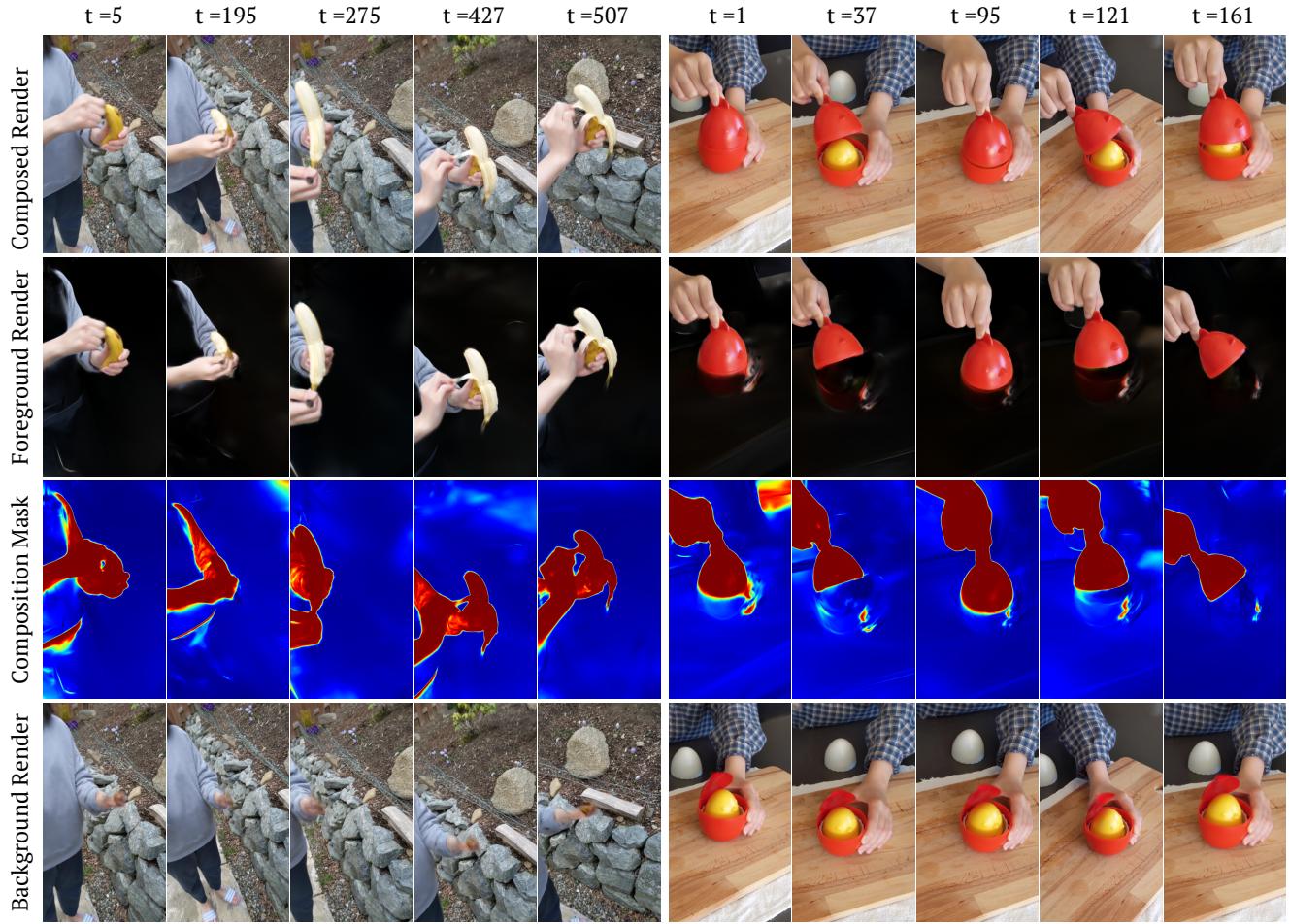


Figure 2. Visualization of dynamic modeling on peel banana and chicken sequence on HyperNerf Vrig dataset [14] dataset. Our methods reconstruct high-quality dynamic scenes with an efficient dynamic-static decoupled representation.

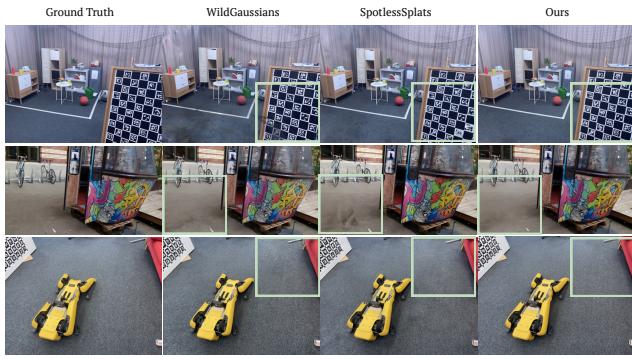


Figure 3. Qualitative comparison of baseline methods[10, 16] on Nerf-On-the-go dataset.

- 302      Structure-from-motion revisited. In *Conference on Computer*  
303      *Vision and Pattern Recognition (CVPR)*, 2016. 1, 2, 3
- 304      [18] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys,  
305      and Jan-Michael Frahm. Pixelwise view selection for un-

structured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016. 2

- 306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326
- [19] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Justin Kerr, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, David McAllister, and Angjoo Kanazawa. Nerfstudio: A modular framework for neural radiance field development. In *ACM SIGGRAPH 2023 Conference Proceedings*, 2023. 2
- [20] Vadim Tschernezki, Ahmad Darkhalil, Zhifan Zhu, David Fouhey, Iro Laina, Diane Larlus, Dima Damen, and Andrea Vedaldi. Epic fields: Marrying 3d geometry and video understanding. *Advances in Neural Information Processing Systems*, 36, 2024. 2
- [21] Matias Turkulainen, Xuqian Ren, Iaroslav Melekhov, Otto Seiskari, Esa Rahtu, and Juho Kannala. Dn-splat: Depth and normal priors for gaussian splatting and meshing. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025. 1
- [22] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering.

327 In *Proceedings of the IEEE/CVF Conference on Computer  
328 Vision and Pattern Recognition*, pages 20310–20320, 2024.  
329 1, 2, 3

330 [23] Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng,  
331 Yin Yang, and Chenfanfu Jiang. Physgaussian: Physics-  
332 integrated 3d gaussians for generative dynamics. *arXiv  
333 preprint arXiv:2311.12198*, 2023. 2

334 [24] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing  
335 Zhang, and Xiaogang Jin. Deformable 3d gaussians for  
336 high-fidelity monocular dynamic scene reconstruction. *arXiv  
337 preprint arXiv:2309.13101*, 2023. 3