

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}_{uti} to decouple parameter updates and the adaptive densification process. While both $\mathcal{L}_{\text{main}}$ and \mathcal{L}_{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}_{reg} enforces time smoothness and k-plane total variations, following the settings in [3, 5, 6, 31]. 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 $\hat{\mathbf{C}}_b$, similar to [20], 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}_τ is the probability threshold, σ 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:

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

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 [9, 30] that encourages smooth depth predictions for static background Gaussians, particularly in regions with small color variance:

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

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[23] 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 [14].

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

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

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:

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

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

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

where $\bar{\mathbf{C}}_b$ denotes the novel view rendered from the background branch (detached from gradient propagation), and α is a coefficient that increases linearly with training iterations. The first term ensures an accurate prediction of the brightness control mask, while the second term acts as a regularizer. Finally, the scale loss \mathcal{L}_s penalizes spiky Gaussians, as defined in [32].

The loss coefficients set to balance each loss term is set 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 0.1 for the rest components.

B. Additional Pruning for Dynamic Scene modeling

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

Specifically, a Gaussian is discarded if the maximum value of the product of its opacity σ and its foreground mask elements m'_f —computed across all input views and timestamps—falls below a predefined threshold τ . This procedure effectively eliminates Gaussians that contribute negligibly to the overall dynamic representation for dynamic scene modeling tasks.

C. Detailed Dataset preparation

Aria Glass Recordings [2, 16, 18] feature egocentric video captured at 20-30 FPS, encompassing intensive human-object, human-scene, and human-human interactions, along with challenges such as rapid camera motion and motion blur. We used NerfStudio [28] to preprocess fisheye camera frames from Project Aria, with camera mask and distortion parameters, and camera vignetting mask provided [8, 11]. The original resolution of an Aria frame is 1415×1415 after fisheye undistortion. To balance rendering quality and speed and avoid excessive training time for the nerf baseline neuraldiff, we downsample the frames to 707×707 . The first 50 frames of each sequence are omitted to allow the camera stream to stabilize.

Epic-field Dataset [29] builds upon the EPIC-Kitchen dataset [4], which comprises long egocentric video recordings of human activities in a kitchen recorded at 50 FPS. We use the point clouds and camera poses provided in [29]. To keep a consistent frame rate with aria recordings, we take testing segments of 10,000 consecutive frames and down-sample by 2, which leads to 5000 frames in the end.

NerF On-the-Go Dataset [20] we prepare the Nerf On-the-go dataset following the setting of SpotlessS-plats [22]. The dataset was originally captured with high-resolution images and downsampled 4 times for patio set and 8 times for others, following [13, 20, 22]. We follow the camera undistortion setting of [22]. **Neu3D Dataset** [14] Following the setup of [31], the resolution is downsampled to 1352×1014 . We compute the camera poses and generate a dense point cloud using COLMAP [23, 24] based on the first frame of each video.

HyperNerf dataset As noted in [10], the camera poses are considerably inaccurate, which diminishes the reliability of quantitative comparisons. Therefore, we run colmap[23] to recompute camera poses and focus primarily on qualitative visualizations for this dataset.

D. Implementation Details

Initialization During initialization, for the background branch, Gaussians are derived from point clouds generated using COLMAP [24, 24] or from sparse perception point clouds provided by the ARIA project [11]. The scene boundary is determined based on the range of the background points, with an additional padding equal to 0.3 times the diagonal length of the camera trajectory. Foreground Gaussians are initialized from randomly generated points within this 3D scene boundary.

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

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

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

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

Parameters set up We generally follow the parameter set up in [31]. With the basic resolution of Hexplane set to 256 for egocentric recordings and 64 for other scenes, upsampled by 2 and 4. The learning rate of set to Hexplane is set to 6×10^{-4} and decays to 2×10^{-5} during training. The deformation learning rate is set to 1.6×10^{-4} and decays to 1.6×10^{-5} .

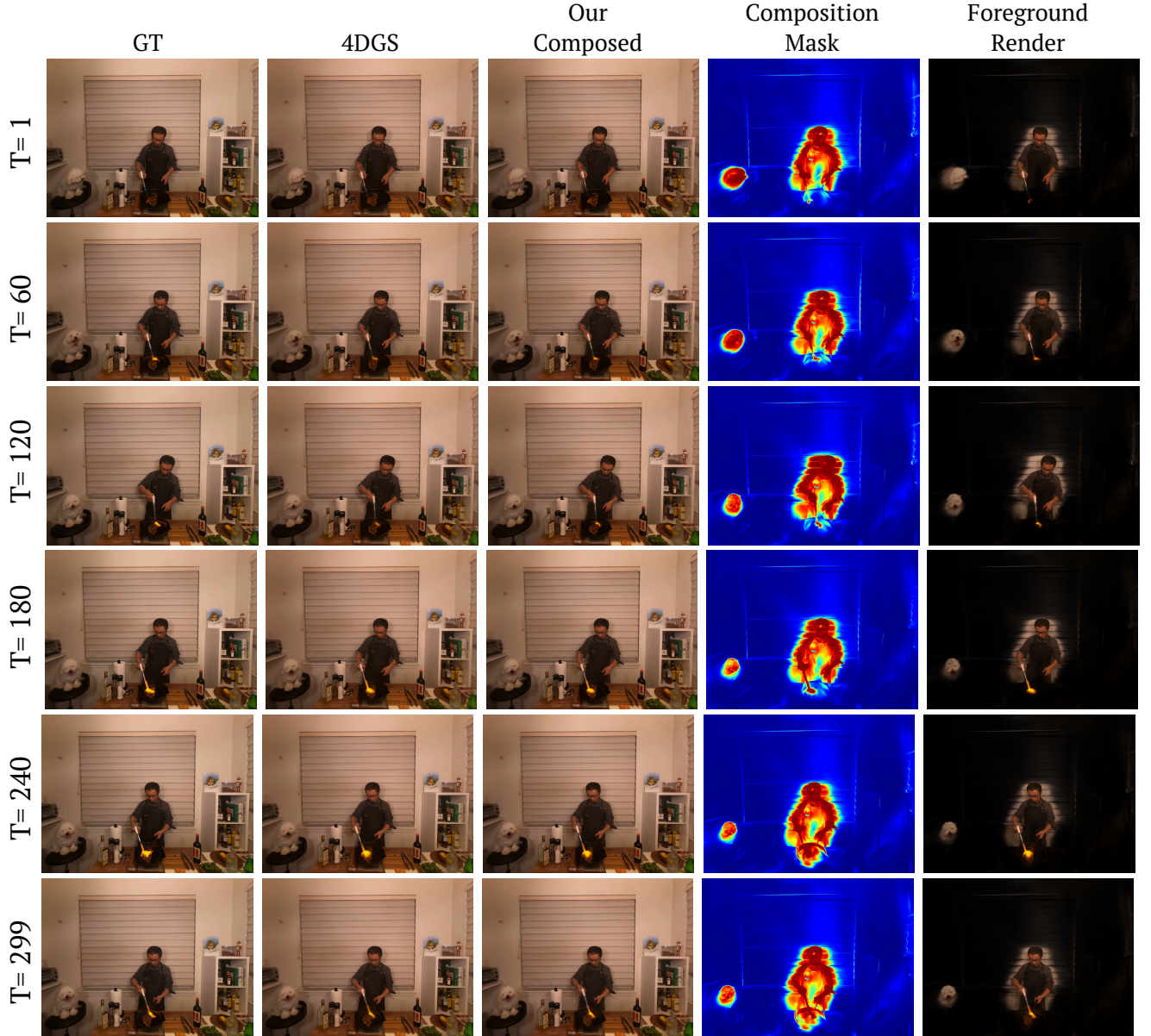


Figure 1. Visualization for Flame Steak Sequence of Neu3D [14] dataset. Our method achieves accurate dynamic-static decomposition with high reconstruction quality.

during training. The deformation learning rate for mask update is set to 1.6×10^{-5} and decays to 1.6×10^{-6} during training. Generally, the batch size is set to 2 as [31]. For low-resolution image collections in [20], we set the batch size to 4 for the dynamic branch and additionally accumulate the update of 4 batches for the background gaussians to account for the low resolution and loose temporal correlations.

Baseline Evaluation For 4DGS [31] experiments, we follow the instruction of their official repo and dataset preparation. For the HyperNerf [19] dataset, we use the colmap calculated camera poses and point cloud for initialization, the same as

our method. For experiments with 3DGS [33] and [22], we use the official repo of [22] and follow their setup. For the Nerf on-the-go dataset [20], EPIC-Field dataset, we use standard colmap initialization. For Aria sequences, the sensor perception point clouds are without color, which leads to unstable initialization for [22]. Therefore, we triangulate with COLMAP [23] using camera poses provided by [11] to obtain colored point cloud to better evaluate this method.

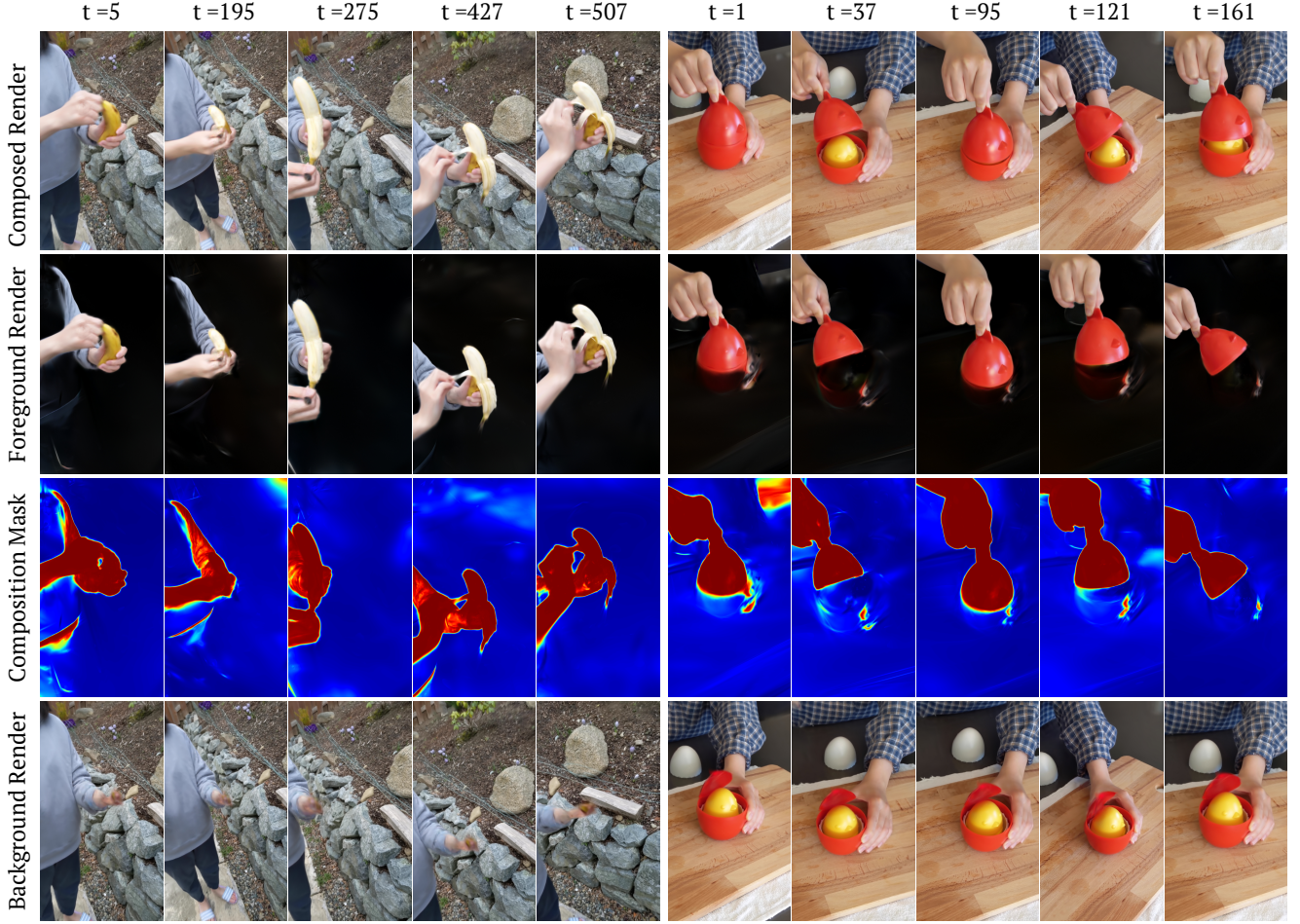


Figure 2. Visualization of dynamic modeling on peel banana and chicken sequence on HyperNerf Vrig dataset [19] dataset. Our methods reconstruct high-quality dynamic scenes with an efficient dynamic-static decoupled representation.

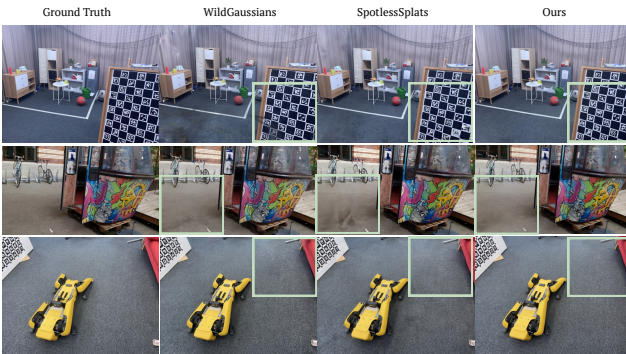


Figure 3. Qualitative comparison of baseline methods[13, 22] on Nerf-On-the-go dataset.

E. Efficiency Analysis on Neu3D [14] dataset

Our dynamic-static hybrid representation enables: **(a) Much higher FPS**: The time critical process of deformation predic-

tion in 4DGS scales with the number of dynamic Gaussians. Table 1 shows we render **3× faster** than 4DGS with superior quality by minimal dynamic element modeling with our dynamic-static decoupling design. **(b) Better Quality** We achieve much higher LPIPS and finer details in static (no stray motion) and dynamic (better handling disappearing gaussians), as reported in our paper and project page: <https://batfacewayne.github.io/DeGauss.io/>. Even on Coffee Martini, Flame Salmon with very far objects that poses challenges to gaussian splatting methods, our LPIPS and details remain best. **(c) Applications** The decoupled static with 3DGS seamlessly enables diverse applications as editing/styling.

F. Strict Monocular Input

Monocular reconstruction is extremely challenging. And compared to NeRF methods [19], dynamic gaussian methods [15, 31, 33] are highly expressive but much harder to regularize, generalizing poorly to novel views [27] (Fig. 4).

Table 1. Quality and efficiency evaluation on all scenes of Neu3D [14] dataset tested on a RTX4090.†: trained densify grad threshold $\times 2$ to reduce number of gaussians.

Method	PSNR(↑)	SSIM(↑)	LPIPS(↓)	Training Time(↓)	FPS(↑)	Dyna. Gaussian num(↓)
NeRFPlayer [26]	30.29	0.909	0.151	6 hours	0.045	-
HyperReel [1]	30.72	0.931	0.101	9 hours	2.0	-
HexPlane [3]	30.00	0.922	0.113	12 hours	0.2	-
KPlanes [6]	31.63	0.964	0.117	5.0 hours	0.3	-
SWinGS [25]	31.12	0.941	0.095	-	71	-
4DGS [31]	31.12	0.937	0.058	0.85 hours	53	124,197
4DGS† [31]	28.72	0.919	0.078	0.67 hours	68	62298
Ours	31.52	0.942	0.047	2.1 hours	71	56,533
Ours†	31.56	0.942	0.047	2 hours	157	22,479

This actually enables our gaussian-based *decoupled design*, to fast and robustly separate dynamic/static modeling for a wide range of inputs, and our explicit static modeling leads to much better generalizability of novel view synthesis for dynamic scene modeling(Fig. 4).



Figure 4. Comparison with baseline methods on novel view synthesis with causal strict monocular input of dycheck-iphone dataset [7].

G. Additional Experiment on Bonn RGBD dataset [17]

To further demonstrate generalizability our method, we evaluate on the *Crowd* scene of a SLAM dataset-Bonn RGBD [17], preprocessed with SFM and MVS pipeline of [23, 24]. We qualitatively show the distractor-free static scene reconstruction and dynamic-static decoupling results in Fig. 5.



Figure 5. Evaluation on 928 frames long highly dynamic *Crowd* scene of Bonn RGBD dataset [17](with only RGB as input).

Table 2. Quantitative results on RobustNerf [21] dataset. Our method shows best overall performance and significantly better LPIPS score over all baseline methods.

Method	Android			Crab2			Statue			Yoda		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
3DGS [12]	23.32	0.794	0.159	31.76	0.925	0.172	20.83	0.830	0.148	28.92	0.905	0.192
WildGaussians [13]	24.67	0.828	0.151	30.52	0.909	0.213	22.54	0.863	0.129	30.55	0.905	0.202
SpotLessSplat [22]	24.20	0.810	0.159	33.90	0.933	0.169	21.97	0.821	0.163	34.24	0.938	0.156
Ours	24.54	0.813	0.083	34.48	0.952	0.076	23.08	0.861	0.097	33.48	0.947	0.082

H. Additional Experiment on RobustNerf [21] dataset

We additionally report the performance of our method on RobustNerf [21] in Tab. 2 and Fig. 6.

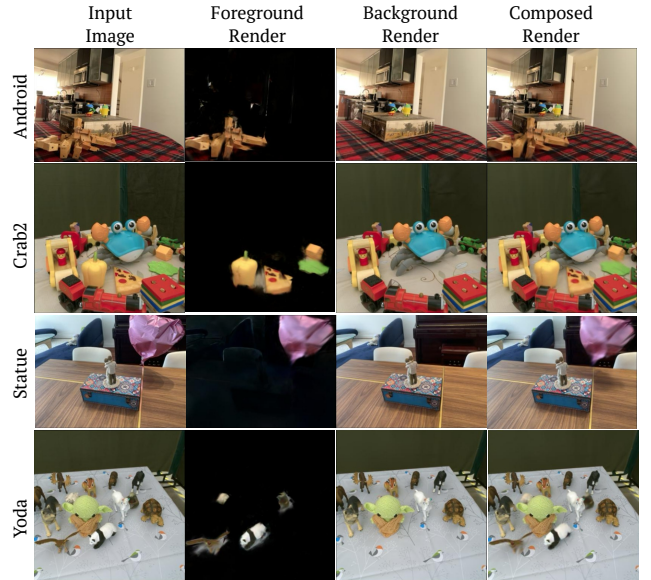


Figure 6. Qualitative result on RobustNerf dataset [21].

I. Additional Visualizations

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

J. Discussion

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

Camera Pose Optimization. Our approach generally assumes reasonably accurate camera poses to facilitate static-dynamic decomposition. Nonetheless, we observe that even when camera poses are suboptimal (as in HyperNeRF [19]), our method can still separate dynamic and static regions. An interesting direction for future research is to leverage our predicted masks to optimize camera poses based on regions identified as static.

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

References

- [1] Benjamin Attal, Jia-Bin Huang, Christian Richardt, Michael Zollhoefer, Johannes Kopf, Matthew O’Toole, and Changil Kim. HyperReel: High-fidelity 6-DoF video with ray-conditioned sampling. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 5
- [2] 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
- [3] 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, 5
- [4] 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
- [5] 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
- [6] 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, 5
- [7] Hang Gao, Ruilong Li, Shubham Tulsiani, Bryan Russell, and Angjoo Kanazawa. Monocular dynamic view synthesis: A reality check. *Advances in Neural Information Processing Systems*, 35:33768–33780, 2022. 5
- [8] 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
- [9] 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
- [10] 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
- [11] 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 Bouttefroy, 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, 2022. 2, 3
- [12] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), 2023. 5
- [13] Jonas Kulhanek, Songyou Peng, Zuzana Kukelova, Marc Pollefeys, and Torsten Sattler. Wildgaussians: 3d gaussian splatting in the wild. *NeurIPS*, 2024. 2, 4, 5
- [14] 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, 3, 4, 5
- [15] Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *3DV*, 2024. 4
- [16] Zhaoyang Lv, Nicholas Charron, Pierre Moulon, Alexander Gamino, Cheng Peng, Chris Sweeney, Edward Miller, Huixuan Tang, Jeff Meissner, Jing Dong, et al. Aria everyday activities dataset. *arXiv preprint arXiv:2402.13349*, 2024. 2
- [17] E. Palazzolo, J. Behley, P. Lottes, P. Giguère, and C. Stachniss. ReFusion: 3D Reconstruction in Dynamic Environments for RGB-D Cameras Exploiting Residuals. 2019. 5
- [18] Xiaqing Pan, Nicholas Charron, Yongqian Yang, Scott Peters, Thomas Whelan, Chen Kong, Omkar Parkhi, Richard Newcombe, and Yuheng Carl Ren. Aria digital twin: A new benchmark dataset for egocentric 3d machine perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20133–20143, 2023. 2
- [19] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *arXiv preprint arXiv:2106.13228*, 2021. 3, 4, 6
- [20] Weining Ren, Zihan Zhu, Boyang Sun, Jiaqi Chen, Marc Pollefeys, and Songyou Peng. Nerf on-the-go: Exploiting uncertainty for distractor-free nerfs in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8931–8940, 2024. 1, 2, 3

- [21] Sara Sabour, Suhani Vora, Daniel Duckworth, Ivan Krasin, David J Fleet, and Andrea Tagliasacchi. Robustnerf: Ignoring distractors with robust losses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20626–20636, 2023. [5](#)
- [22] Sara Sabour, Lily Goli, George Kopanas, Mark Matthews, Dmitry Lagun, Leonidas Guibas, Alec Jacobson, David J Fleet, and Andrea Tagliasacchi. Spotlessplats: Ignoring distractors in 3d gaussian splatting. *arXiv preprint arXiv:2406.20055*, 2024. [2](#), [3](#), [4](#), [5](#)
- [23] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. [1](#), [2](#), [3](#), [5](#)
- [24] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016. [2](#), [5](#)
- [25] Richard Shaw, Michal Nazarczuk, Jifei Song, Arthur Moreau, Sibi Catley-Chandar, Helisa Dhamo, and Eduardo Perez-Pellitero. Swings: Sliding windows for dynamic 3d gaussian splatting. *arXiv preprint arXiv:2312.13308*, 2023. [5](#)
- [26] Liangchen Song, Anpei Chen, Zhong Li, Zhang Chen, Lele Chen, Junsong Yuan, Yi Xu, and Andreas Geiger. Nerfplayer: A streamable dynamic scene representation with decomposed neural radiance fields. *IEEE Transactions on Visualization and Computer Graphics*, 29(5):2732–2742, 2023. [5](#)
- [27] Colton Stearns, Adam Harley, Mikaela Uy, Florian Dubost, Federico Tombari, Gordon Wetzstein, and Leonidas Guibas. Dynamic gaussian marbles for novel view synthesis of casual monocular videos. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024. [4](#)
- [28] 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](#)
- [29] 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](#)
- [30] Matias Turkulainen, Xuqian Ren, Iaroslav Melekhov, Otto Seiskari, Esa Rahtu, and Juho Kannala. Dn-splatter: 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](#)
- [31] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20310–20320, 2024. [1](#), [2](#), [3](#), [4](#), [5](#)
- [32] Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. Physgaussian: Physics-integrated 3d gaussians for generative dynamics. *arXiv preprint arXiv:2311.12198*, 2023. [2](#)
- [33] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. *arXiv preprint arXiv:2309.13101*, 2023. [3](#), [4](#)