







Supplementary Material for *EgoLifter*: Open-world 3D Segmentation for Egocentric Perception

Qiao Gu^{1,2*}, Zhaoyang Lv², Duncan Frost², Simon Green²,
Julian Straub², and Chris Sweeney²

¹ University of Toronto, Toronto, ON M5S 1A1, Canada
q.gu@mail.utoronto.ca

² Meta Reality Labs, Redmond, WA 98052, USA
{zhaoyang, frost, simongreen, jstraub, sweeneychris}@meta.com

A1 Video Qualitative Results

Please refer to videos on the project page ³, which contains:

- Videos qualitative results of the multiple applications of *EgoLifter* (corresponding to Fig. 1).
- Video qualitative results on the ADT dataset, comparing *EgoLifter* and its variants (corresponding to Fig. 3).
- Video qualitative results on the ADT dataset, comparing with Gaussian Grouping [12] (corresponding to Fig. 4).
- Video qualitative results on the AEA and Ego-Exo4D datasets. (corresponding to Fig. 3).
- Demonstration video of the interactive visualization and segmentation system.

A2 Experiment Details

A2.1 Image Formation Model for Project Aria

Aria Glasses [4] use a fisheye camera, and thus recorded images have a fisheye distortion and vignette effect, but 3DGS uses a linear camera model and does not have a vignette effect. Therefore we account for these effects in training 3D Gaussian models using the image formation model $f(\cdot)$ in Eq. 1, such that not the raw rendered image but a processed one is used for loss computation. Specifically, we apply an image processing pipeline as shown in Fig. A.1. In the pipeline, the raw recorded images are first rectified to a pinhole camera model using `projectaria_tools`⁴, and then multiplied with a valid-pixel mask that removes the pixels that are too far from the image center. The rendered image

* Work done during internship at Reality Labs, Meta.

³ <https://egolifter.github.io/>

⁴ Link

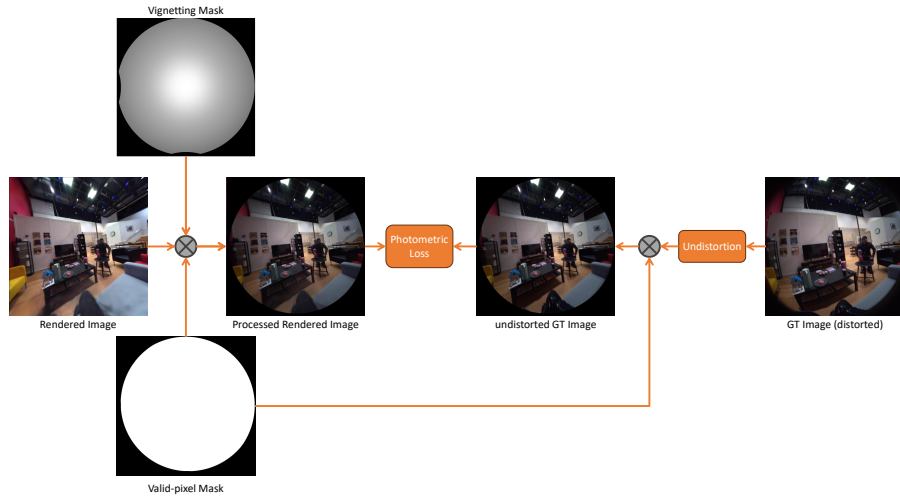


Fig. A.1: Image processing pipeline during training. The \otimes symbol indicates element-wise multiplication.

from 3DGS is multiplied with a vignette mask and also the valid-pixel masks. Then the photometric losses are computed between the processed rendered image and the processed GT image during training. This pipeline models the camera model used in Aria glasses and leads to better 3D reconstruction. Empirically we found that without this pipeline, 3DGS will create a lot of floaters to account for the vignette effect in the reconstruction and significantly harm the results.

A2.2 Additional Training Details

Due to the GPU memory constraint, we sampled at most $|\mathcal{U}| = 4096$ pixels within the valid-pixel mask for computing the contrastive loss in Eq. 2. Note that for *EgoLifter* where the transient prediction is used, the samples are additionally constrained to be pixels with transient probability less than $\delta = 0.5$.

For the segmentation masks generated by SAM, some masks may have overlapped with each other. In our experiments, we discarded the information about overlapping and simply overlaid all masks on the image space to get a one-hot segmentation for each pixel. While making use of these overlapping results leads to interesting applications like hierarchical 3D segmentation as shown in [6, 13], this is beyond the scope of *EgoLifter* and we left this for future exploration. The images used for training are of resolution of 1408×1408 and segmentation masks from SAM are in the resolution of 512×512 . Therefore, during training, two forward passes are performed. In the first pass, only the RGB image is rendered at the resolution of 1408×1408 and in the second, only the feature map is rendered at 512×512 . The losses are computed separately from each pass and

Table A.1: 2D instance segmentation results (measured in mIoU) and novel view synthesis results (measured in PSNR) on **seen** subsets in the ADT dataset.

Evaluation Object set	mIoU (In-view)			mIoU (Cross-view)			PSNR		
	Static	Dynamic	All	Static	Dynamic	All	Static	Dynamic	All
SAM [7]	62.74	52.48	61.00	-	-	-	-	-	-
Gaussian Grouping [12]	40.86	42.24	41.09	32.26	26.23	31.24	27.97	19.13	25.53
<i>EgoLifter</i> -Static	64.34	57.71	63.21	62.20	35.39	57.64	27.65	19.60	25.64
<i>EgoLifter</i> -Deform	63.33	57.11	62.27	62.24	34.91	57.59	28.60	19.89	26.24
<i>EgoLifter</i> (Ours)	65.08	52.12	62.88	63.65	33.70	58.56	26.86	16.02	23.34

summed up for gradient computation. Note that the view-space gradients from both passes are also summed for deciding whether to split 3D Gaussians.

For optimization on the 3D Gaussian models, we adopt the same setting as used in the original implementation [5], in terms of parameters used in the optimizer and scheduler and density control process. The learning rate for the additional per-Gaussian feature vector \mathbf{f}_i is 0.0025, the same as that for updating color \mathbf{c}_i . All models are trained for 30,000 iterations on each scene in the ADT dataset, and for 100,000 iterations on scenes in the AEA and Ego-Exo4D datasets, as these two datasets contain more frames in each scene. In the latter case, the learning rate scheduler and density control schedule are also proportionally extended.

A2.3 Timing

Using one NVIDIA A100 (40GB), training *EgoLifter* on one ADT sequence takes around 130 minutes (training vanilla 3DGS takes around 100 minutes). For a trained *EgoLifter* model, rendering both the RGB image and the instance feature map of 1408×1408 resolution runs at around 103 fps. If only RGB images are rendered, the speed goes to 158 fps. Note that we use a different implementation than the original 3DGS, where we made several changes like not caching images on GPU to enable training on large datasets, e.g. AEA and Ego-Exo4D.

A2.4 ADT Dataset Benchmark

Sequence selection Based on the 218 sequences in the full ADT datasets [9], we filter out the sequences that have too narrow baselines for 3D reconstruction (sequences with name starting with `Lite_release_recognition`) or do not have segmentation annotation on human bodies. From the rest of the sequences, we select 16 sequences for evaluation, where 6 of them contain recordings of Aria glasses from two human users in the scene (sequences with `multiskeleton` in the name), and the rest 10 only have recordings from one user, although there may be multiple two persons in the scene (sequences with `multiuser` in the name). The names of the selected sequences are listed as follows:

`Apartment_release_multiskeleton_party_seq121`

Apartment_release_multiskeleton_party_seq122
 Apartment_release_multiskeleton_party_seq123
 Apartment_release_multiskeleton_party_seq125
 Apartment_release_multiskeleton_party_seq126
 Apartment_release_multiskeleton_party_seq127
 Apartment_release_multiuser_cook_seq114
 Apartment_release_multiuser_meal_seq140
 Apartment_release_multiuser_cook_seq143
 Apartment_release_multiuser_party_seq140
 Apartment_release_multiuser_clean_seq116
 Apartment_release_multiuser_meal_seq132
 Apartment_release_work_skeleton_seq131
 Apartment_release_work_skeleton_seq140
 Apartment_release_meal_skeleton_seq136
 Apartment_release_decoration_skeleton_seq137

A2.5 Comparison with NeRF



Fig. A.2: Qualitative results on ADT datasets. From left to right: GT image; Render by Nerfacto; Render by *EgoLifter*.

Method	Nerfacto	<i>EgoLifter</i>
PSNR (all)	17.22	20.28

Table A.2: Comparison to Nerfacto on the ADT dataset.

Method	INGP-Big	M-NeRF360	3DGS	EgoLifter
PSNR	25.59	27.69	27.21	27.26

Table A.3: Quantitative comparison on the MipNeRF 360 dataset.



Fig. A.3: Qualitative results on MipNeRF 360. From left to right: GT image; *EgoLifter* RGB render; *EgoLifter* feature map (PCA).

Subset Splitting For sequences that only have a recording from one pair of Aria glasses, the first 4/5 of the video is considered as seen views and the rest are considered as novel ones. For sequences that have videos from two pairs, the video from one pair is considered as seen views and the other is considered as novel views. During training, every 1 out of 5 consecutive frames in the seen views are used for validation the remaining 4 are used for training. The entire novel subset is hidden from training and solely used for evaluation. For evaluation on 2D instance segmentation, we uniformly sampled at most 200 frames from each subset for fast inference. The objects in each video sequence are also split into dynamic and static subsets, according to whether their GT object positions have changed by over 2cm over the duration of each recording. Humans are always considered dynamic objects.

A3 Additional Results

A3.1 Results on ADT Seen Subset

For completeness, we also report the 2D instance segmentation and photometric results on the **seen** subset of ADT in Tab. A.1. Note that the frames used for evaluation in the seen subset are closer to those for training, and therefore these results mostly reflect how well the models overfit the training viewpoints in each scene, rather than generalize to novel views. As we can see from Tab. A.1, *EgoLifter* outperforms the baselines in segmenting static objects using both in-view

Method	Office0	Office1	Office2	Office3	Office4	Room0	Room1	Room2	Mean
MVSeg [8]	31.4	40.4	30.4	30.5	25.4	31.1	40.7	29.2	32.4
SA3D [3]	84.4	77.0	88.9	84.4	82.6	77.6	79.8	89.2	83.0
OmniSeg3D [13]	83.9	85.3	89.0	87.2	78.3	83.0	79.4	88.9	84.4
<i>EgoLifter</i> (Ours)	82.9	78.4	85.1	84.1	80.0	77.0	85.4	84.3	82.1

Table A.4: Instance Segmentation results (mean IoU) on Replica dataset.

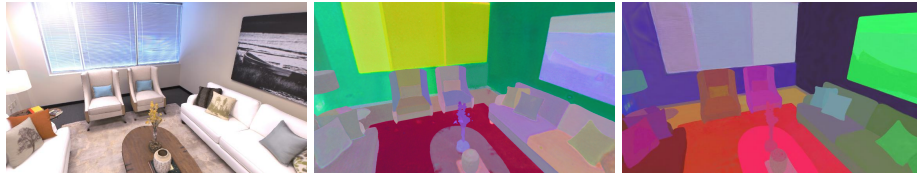


Fig. A.4: Qualitative result on Replica datasets. From left to right: GT image; OmniSeg3D feature map; *EgoLifter* feature map

and cross-view queries. When both static and dynamic objects are considered (the “All” column), *EgoLifter* still achieves the best results in cross-view, which is a harder setting for open-world segmentation. *EgoLifter* also has the second place in the in-view setting.

A3.2 NeRF on ADT Dataset

In Tab. A.2 and Fig. A.2, we compare *EgoLifter* with the (default) nerfacto model in Nerfstudio [11] on the ADT dataset. As we can see from Fig. A.2, although Nerfacto uses per-image appearance embeddings to filter out transient phenomena in reconstruction, it still fails on challenging egocentric datasets like ADT and results in many floaters in the rendering. Quantitatively, *EgoLifter* also outperforms as shown in Fig. A.2.

A3.3 Non-egocentric public benchmarks

Non-egocentric benchmarks (Replica, ScanNet, MipNeRF 360) use careful hand-held scanning motions and lack dynamic phenomena. Therefore, they do not reflect the full capability of *EgoLifter*. We evaluate *EgoLifter* on the MipNeRF 360 dataset [1] in Tab. A.3 and Fig. A.3, where we use *EgoLifter*-static variant as the scenes are all static. Due to the lack of GT segmentation masks, we provide qualitative results on learned instance features in Fig. A.3. As shown in Tab. A.3 and Fig. A.3, *EgoLifter* has a similar PSNR as the original 3DGS and learns clean instance features that distinguish different instances.

We also test *EgoLifter* on Replica [10] and compare to OmniSeg3D [13], a recent feature lifting method based on NeRF representation and contrastive learning [2]. We evaluate the instance segmentation task using the multi-view

mask propagation protocol [3, 8, 13], where the GT mask from one view is used for computing reference instance features and masks on other view are computed based on the feature distance from the reference ones. We follow the evaluation protocol in [13] and use Eq. (11) in [13] for computing the similarity scores. Similar to the experiments on MipNeRF 360, we used *EgoLifter*-static as there is no dynamic content in Replica scenes.

We report the quantitative results (in mIoU) in Tab. A.4 and a qualitative example in Fig. A.4. From Tab. A.4, we can see that *EgoLifter* has similar performance with the state-of-the-art NeRF-based segmentation methods [3, 13] on the non-egocentric Replica dataset. From the qualitative example in Fig. A.4, we can see that *EgoLifter* also results in clean and sharp feature boundaries on Replica as contemporary work OmniSeg3D [13], which distinguish different object instances and even the parts within each object.

A4 Additional Discussion on Limitations

Due to form factor and power constraints, egocentric videos are often captured with more challenges. Due to rapid head motion and lighting condition changes in the egocentric videos, the images contain significant motion blur that causes challenges in recovering sharp reconstructions from them. This explains in part the blurry results shown in some of the reconstruction results by *EgoLifter*. We leave how to improve the reconstruction quality from egocentric videos for future work.

References

1. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mipnerf 360: Unbounded anti-aliased neural radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5470–5479 (2022)
2. Bhalgat, Y., Laina, I., Henriques, J.F., Zisserman, A., Vedaldi, A.: Contrastive lift: 3d object instance segmentation by slow-fast contrastive fusion. In: Advances in Neural Information Processing Systems (2023)
3. Cen, J., Zhou, Z., Fang, J., Shen, W., Xie, L., Jiang, D., Zhang, X., Tian, Q., et al.: Segment anything in 3d with nerfs. Advances in Neural Information Processing Systems **36**, 25971–25990 (2023)
4. Engel, J., Somasundaram, K., Goesele, M., Sun, A., Gamino, A., Turner, A., Tallatof, A., Yuan, A., Souti, B., Meredith, B., Peng, C., Sweeney, C., Wilson, C., Barnes, D., DeTone, D., Caruso, D., Valleroy, D., Ginjupalli, D., Frost, D., Miller, E., Mueggler, E., Oleinik, E., Zhang, F., Somasundaram, G., Solaira, G., Lanaras, H., Howard-Jenkins, H., Tang, H., Kim, H.J., Rivera, J., Luo, J., Dong, J., Straub, J., Bailey, K., Eckenhoff, K., Ma, L., Pesqueira, L., Schwesinger, M., Monge, M., Yang, N., Charron, N., Raina, N., Parkhi, O., Borschowa, P., Moulon, P., Gupta, P., Mur-Artal, R., Pennington, R., Kulkarni, S., Miglani, S., Gondi, S., Solanki, S., Diener, S., Cheng, S., Green, S., Saarinen, S., Patra, S., Mourikis, T., Whelan, T., Singh, T., Balntas, V., Baiyya, V., Dreewes, W., Pan, X., Lou, Y., Zhao, Y.,

- Mansour, Y., Zou, Y., Lv, Z., Wang, Z., Yan, M., Ren, C., Nardi, R.D., Newcombe, R.: Project aria: A new tool for egocentric multi-modal ai research. arXiv preprint arXiv:2308.13561 (2023)
5. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (ToG)* **42**(4), 1–14 (2023)
 6. Kim, C.M., Wu, M., Kerr, J., Goldberg, K., Tancik, M., Kanazawa, A.: Garfield: Group anything with radiance fields. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 21530–21539 (2024)
 7. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 4015–4026 (2023)
 8. Mirzaei, A., Aumentado-Armstrong, T., Derpanis, K.G., Kelly, J., Brubaker, M.A., Gilitschenski, I., Levinshtein, A.: Spin-nerf: Multiview segmentation and perceptual inpainting with neural radiance fields. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 20669–20679 (2023)
 9. Pan, X., Charron, N., Yang, Y., Peters, S., Whelan, T., Kong, C., Parkhi, O., Newcombe, R., Ren, Y.C.: Aria digital twin: A new benchmark dataset for egocentric 3d machine perception. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 20133–20143 (2023)
 10. Straub, J., Whelan, T., Ma, L., Chen, Y., Wijmans, E., Green, S., Engel, J.J., Mura, R., Ren, C., Verma, S., Clarkson, A., Yan, M., Budge, B., Yan, Y., Pan, X., Yon, J., Zou, Y., Leon, K., Carter, N., Briales, J., Gillingham, T., Mueggler, E., Pesqueira, L., Savva, M., Batra, D., Strasdat, H.M., Nardi, R.D., Goesele, M., Lovegrove, S., Newcombe, R.: The Replica dataset: A digital replica of indoor spaces. arXiv preprint arXiv:1906.05797 (2019)
 11. Tancik, M., Weber, E., Ng, E., Li, R., Yi, B., Wang, T., Kristoffersen, A., Austin, J., Salahi, K., Ahuja, A., et al.: Nerfstudio: A modular framework for neural radiance field development. In: *ACM SIGGRAPH 2023 Conference Proceedings*. pp. 1–12 (2023)
 12. Ye, M., Danelljan, M., Yu, F., Ke, L.: Gaussian grouping: Segment and edit anything in 3d scenes. arXiv preprint arXiv:2312.00732 (2023)
 13. Ying, H., Yin, Y., Zhang, J., Wang, F., Yu, T., Huang, R., Fang, L.: Omnise3d: Omniversal 3d segmentation via hierarchical contrastive learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 20612–20622 (2024)