LoD-Loc v2: Aerial Visual Localization over Low Level-of-Detail City Models using Explicit Silhouette Alignment

Supplementary Material

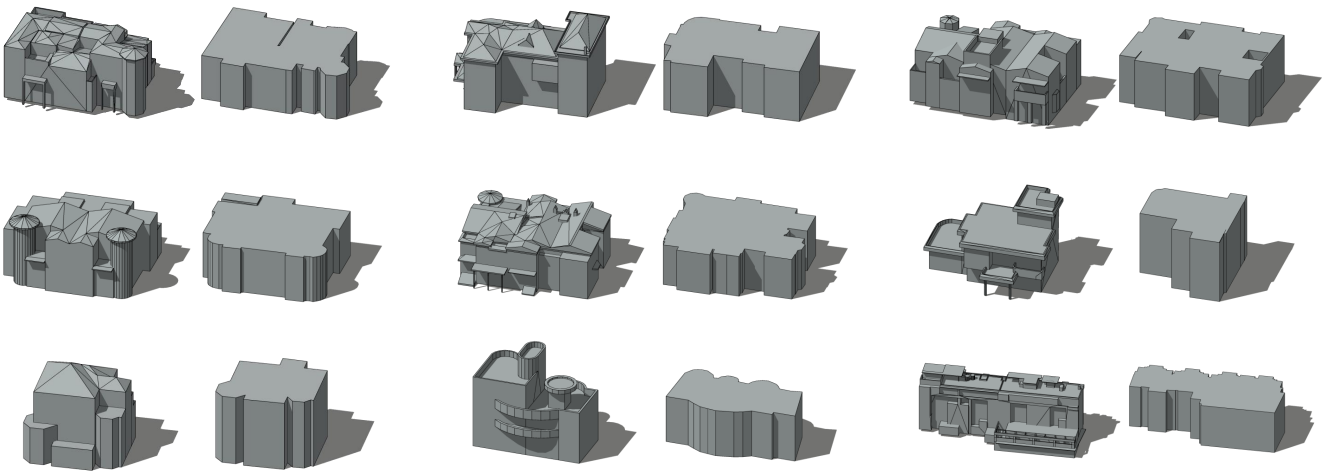


Figure 1. **Visualization of LoD3 and LoD1 Building Models.** This figure illustrates a side-by-side comparison of LoD3 and LoD1 building models from the UAVD4L-LoD dataset [28] and the UAVD4L-LoDv2 dataset, respectively. In each pair, the model on the left represents LoD3, showcasing intricate details such as complex roof structures, facades, and other architectural elements. The model on the right represents LoD1, with a simplified representation consisting of basic geometric shapes and flat surfaces.

1. Demo Video

Supplementary material includes demonstration videos that showcase the localization performance of our proposed method on two distinct aerial sequences—one from an urban area and the other from a rural area. We visualize the results by overlaying the projected LoD model onto the query images using the estimated poses, clearly illustrating the outcomes. Our results show that our approach effectively localizes UAVs and outperforms the LoD-Loc [28] algorithm.

Metric	LoD3	LoD1
Geometry	Detailed structures	Simple block
Roof	Precise roof	Simple flat
Triangles	145,124	45,734
Vertices	98,021	31,182
File Size	9.6 MB	4.41 MB
Manual Work	Need	No need

Table 1. **Comparison between LoD3 and LoD1 Models.**

2. Details of Dataset Collection

We collect and release two novel datasets: the UAVD4L-LoDv2 dataset and the Swiss-EPFLv2 dataset. 1). The UAVD4L-LoDv2 dataset comprises a LoD1 map and UAV-captured query images with their corresponding pose annotations; the data was captured in the area of Changsha, China, covering an area of 2.5 km^2. 2). The Swiss-EPFLv2 dataset includes a LoD1 map along with UAV-captured query images and their pose labels; the data was collected near the École Polytechnique Fédérale de Lausanne in Switzerland, covering an area of 8.2 km^2.

LoD1 model construction. Unlike LoD3 models requiring additional manual involvement to complete structural details, LoD1 models can be automatically constructed by using the DP modeler [3] based on the mesh model. Given that the UAVD4L dataset [23] provides these mesh models, we automatically generate the corresponding LoD1 models for the UAVD4L-LoDv2 dataset. For the Swiss-EPFLv2 dataset, no corresponding mesh models are available. Therefore, we manually label the building footprints and automatically generate the LoD1 model using the digital surface model provided by swisstopo [6].

Differences between hierarchical LoDs. Models at different LoDs exhibit varying amounts of information, typically arranged in descending order—from highest (LoD3) to lowest (LoD1). Fig. 2 illustrates different LoD models using toy examples. The main difference lies in geometric complexity. LoD3 models include detailed external features such as windows, doors, balconies, and complex roofs, and

Name	Capture device	Capture pitch angle	Capture height	Capture route
in-Traj.	DJI M300+H20t	0° or 45°	120m	Zig-zag flight
out-of-Traj.	DJI Mavic3 Pro	30° ∼ 60°	90m ∼ 150m	Manually control

Table 2. Differences between the *in-Traj.* and *out-of-Traj.* sequences.

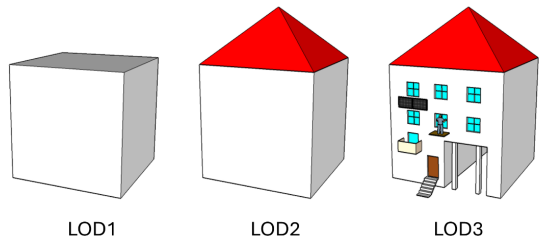


Figure 2. **Toy examples of different LoD models** [24].

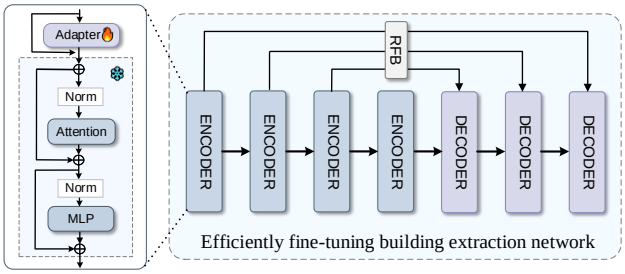


Figure 3. **Architecture of Segmentation Module** [25].

are primarily used for high-precision visualization and architectural design. By contrast, LoD2 models feature roof structures and general building shapes but omit finer details like windows and doors. Meanwhile, LoD1 models represent buildings with simple geometric shapes such as cubes, focusing on large-scale scene analysis, making them suitable for urban planning. Fig. 1 and Tab. 1 present a detailed comparison over the metrics between LoD1 and LoD3 from UAVD4L region.

Query image collection. The UAV query images and annotations in the UAVD4L-LoDv2 dataset were collected from the UAVD4L-LoD [28] dataset, including the *in-Traj.* and *out-of-Traj.* sequences. These sequences cover various types of buildings, such as office buildings, villas, apartment blocks, private residences, rural low-rise structures, and schools, totaling 3,796 images. The query dataset was captured using two UAV models: a DJI M300 [2] UAV equipped with an H20T [1] camera and a DJI Mavic3 Pro [4]. Tab. 2 highlights the differences between the two query image sequences. For details on the ground truth pose annotations, refer to [28]. The Swiss-EPFLv2 dataset contains 1,091 real query images derived from [28], all captured using a DJI Phantom 4 RTK.

3. Architecture of Segmentation Module

We adopt the SAM2-UNet [25] segmentation network for architectural extraction tasks, as shown in Fig. 3. SAM2-UNet utilizes the Hiera [18] backbone from Segment Anything Model 2 (SAM2) [16] as its encoder and incorporates a classic U-shaped [17] decoder, offering a straightforward and robust framework for image segmentation. The network employs Receptive Field Blocks (RFBs) [8, 11] to compress channel dimensions and integrate adapters for parameter-efficient fine-tuning. These adapters, inspired by similar designs [10, 15], consist of linear down-sampling, GeLU activation, and linear up-sampling modules, allowing precise task adaptation while preserving the backbone's frozen parameters. This approach exhibits exceptional generalizability and robustness, effectively addressing diverse segmentation needs in architectural contexts.

4. Details of Baseline

UAVD4L & CAD-Loc. Both UAVD4L [23] and CAD-Loc [14] localization pipelines consist of two stages, retrieval, and matching, to recover the camera pose based on the image databases. Their experimental setups are adapted from [28], with the reference image databases sourced from a subset of the UAVD4L dataset. Specifically, we employed the sensor-guided image retrieval method to narrow down the retrieval scope during the first stage.

$$ {}^{q}\mathcal{I} = \{\mathbf{I}_i^r \mid \|\mathbf{t}_i^r - \mathbf{t}^q\| \leq \delta_t \wedge \arccos(\mathbf{R}_i^r, \mathbf{R}^q) \leq \delta_o\} \quad (1) $$

Here, $\|\cdot\|$ denotes the Frobenius Norm between two translation matrices, while $\arccos(\cdot)$ calculates the rotation angles between two rotation matrices. δ_t and δ_o represent the thresholds for translation distance and orientation difference, respectively. Following the [28], we set $\delta_t = 150$ and $\delta_o = 30$ to filter query and reference images based on spatial and rotational similarity.

MC-Loc. We adopt the default settings of MC-Loc [22], using both the DINOv2 [13] or RoMa [7] feature extractor as baselines. The input image resolution and the resolution of rendered candidates, are set to $(256, 340)$. The number of iterations for the three stages was set to 40, 30, and 60, respectively, with each stage generating 52, 40, and 32 candidate poses. The first two stages used dual-beam optimization, while the final stage employed single-beam optimization. For more details, refer to [22].

	Backbone	*in-Traj.*	*out-of-Traj.*
Implicit Features	DINO v2 [13]	2.50 / 7.50 / 27.6	2.80 / 9.10 / 30.0
early trial	Depth Anything [27]	0.50 / 2.90 / 18.8	1.0 / 3.9 / 23.1
Explicit Silhouettes *ours*	full	**93.7 / 98.4 / 99.5**	**97.9 / 99.8 / 100**

Table 3. Comparison between early-trial implicit-feature baselines and our explicit-silhouette method.

Iter.	*in-Traj.*	*out-of-Traj.*
0	3.0 / 7.7 / 27.9	11.7 / 29.9 / 51.1
5	72.5 / 88.0 / 96.1	81.7 / 95.0 / 99.1
15	92.5 / 97.9 / 99.4	96.9 / 99.8 / 100
20	93.7 / 98.4 / 99.5	97.9 / 99.8 / 100

Table 4. **Ablation study on different iterations.**

Δ	Variant	*in-Traj.*	*out-of-Traj.*
50m	no select	19.6/ 26.3 / 28.5	15.9 / 26.0 / 30.2
	no refine	2.56 / 7.23 / 25.9	2.10 / 5.61 / 21.0
	full	**91.3 / 95.1 / 96.7**	**95.9 / 98.3 / 98.5**
100m	no select	1.10 / 2.10 / 2.70	1.90 / 2.90 / 3.60
	no refine	3.12 / 9.10 / 29.4	0.96 / 4.06 / 16.4
	full	**91.3 / 95.1 / 96.2**	**95.0 / 97.8 / 98.0**
200m	no select	0.30 / 0.30 / 0.40	0.60 / 0.70 / 0.90
	no refine	2.99 / 7.73 / 27.9	11.7 / 29.9 / 51.1
	full	**90.3/ 94.1 / 95.2**	**94.7 / 97.2 / 97.4**

Table 5. **Ablation study on different variants.**

LoD-Loc. We adopt the default experimental settings of LoD-Loc [28], with uniform sampling counts of 10, 10, 30, and 8 along the x, y, z, and yaw axes, respectively, and a lambda value set to 0.8. The key difference in this work is that the experiments are conducted based on LoD1 models, with a 3D sampling interval of 1 meter.

5. Details of Experiments

5.1. Visualization of Training Data

The training dataset consists of two parts: synthetic RGB images and their corresponding building mask annotations.

For the synthetic images, part of the training query dataset is sourced from a subset of the synthetic database in UAVD4L [23], with non-building images excluded. Additionally, following the data generation method of the UAVD4L, we use the OSG [5] rendering technique to supplement the dataset with synthetic RGB images at pitch angles of 30 and 60 degrees.

For the building mask annotations, thanks to the LoD model, we can automatically generate the building segmentation training label dataset without the involvement of manual annotations. Building mask annotations are generated by projecting the 3D surfaces of the LoD3 model onto a 2D plane based on the pose of each synthetic image. A total of 18,395 pairs of synthetic data were collected. Fig. 4 visualizes a part of the dataset.

5.2. Early Trial.

It is a natural consideration to extend LoD-Loc [28] that utilizing the reference features are not extracted from a wireframe but instead encoded by a CNN from a rendering of the LoD models. This would be more reminiscent of PixLoc [19] or MC-Loc [22], which leverages the trained features for the task of pose voting. In our early trials, we explored a similar approach by finetuning backbones (e.g., DINOv2 [13], DepthAnything [27]) to extract consistent features between query images and renderings of the LoD model. This process was supervised by contrastive learning. However, we found that this approach underperformed compared to the proposed explicit silhouette alignment with the same training data, as illustrated in the table below. We believe the performance discrepancy may be due to the difficulty in achieving convergence caused by the modality differences.

5.3. Additional Ablation Studies

We provide additional ablation studies in this section, including the localization accuracy at different iteration stages, the performance of various variants under large prior errors, the impact of different orientation priors, and a comparison of computational costs.

Iterations. Tab. 4 demonstrates the effectiveness of iterative refinement in improving performance. Both sequences benefit significantly from additional iterations, with performance quickly saturating after 15 iterations.

Different variants. As shown in Tab. 5, we compare the localization accuracy of different variants under GPS-limited conditions to demonstrate the effectiveness of our modules. Under normal GPS conditions, the presence or absence of the Pose Selection stage has minimal impact on the final localization results of the full model. This is because the prior information provided by normal GPS conditions still lies within the convergence domain of the particle filter optimization algorithm. However, as the prior error increases,

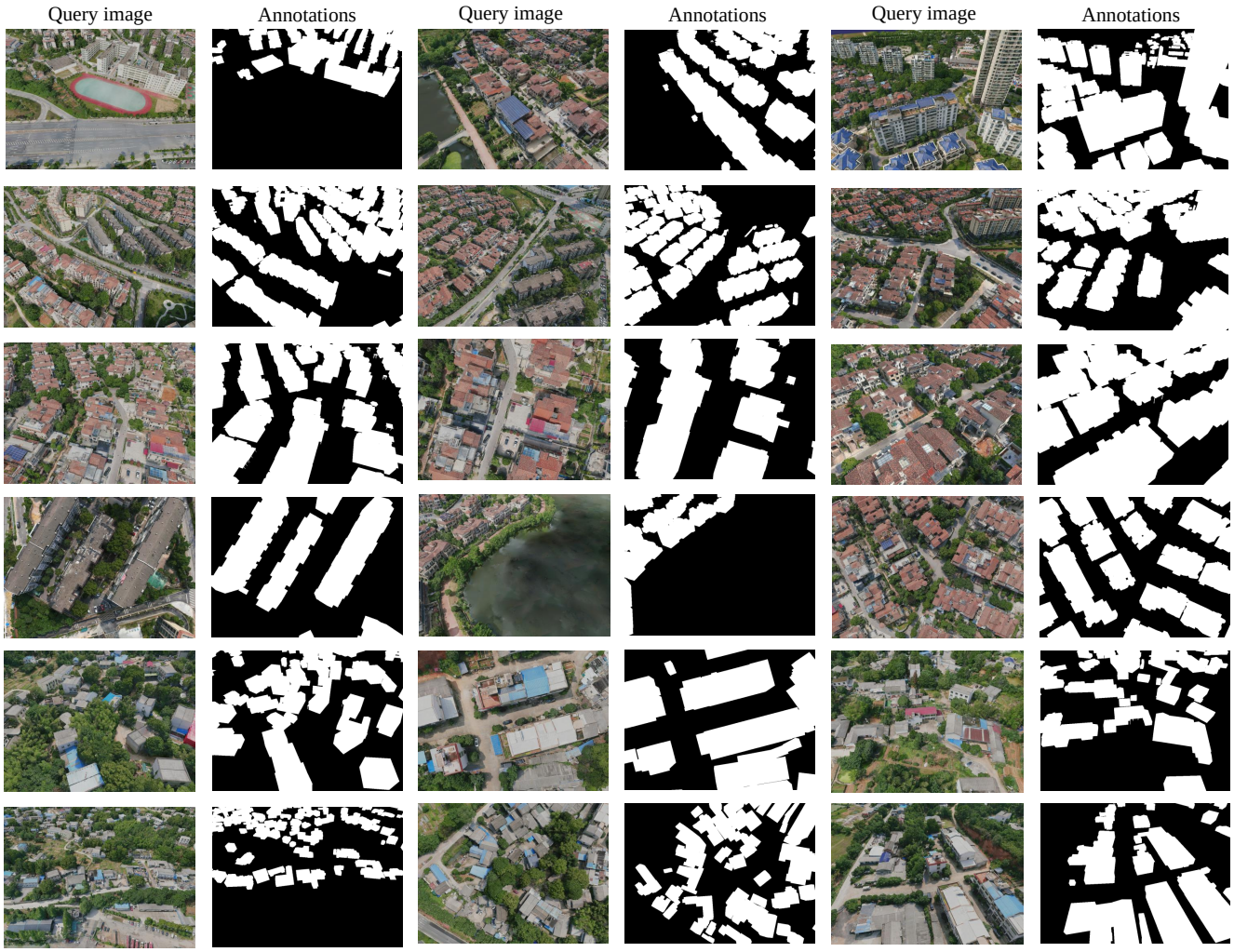


Figure 4. **Visualization of training data.** We avoid the need for complex manual annotations by employing model projection techniques, where the LoD 3D model at the corresponding pose is projected onto a 2D plane to generate building masks. This approach minimizes human intervention and ensures a consistent, automated process for mask generation, reducing potential errors caused by subjective judgment. Additionally, it enables efficient large-scale data processing by leveraging the geometric precision of LoD models, ensuring that the generated masks accurately capture the structural details of buildings without requiring manual effort.

the performance of the *-no select* variant decreases without the support of the Pose Selection stage. Tab. 5 validates the effectiveness of our proposed model.

Different orientation prior. The pose prior is directly from the built-in sensor, which we believe is a reasonable input. In this ablation, we expanded the yaw to ±60° while keeping the gravity direction unchanged due to its accuracy [9, 12, 20, 21, 26]. See Tab. 8 for results.

Cross-scene generalization. Tab. 6 illustrates the generalization capability of LoD-Loc v2 through training and testing in diverse regions. On the UAVD4L-LoDv2 dataset (with region A1 representing an urban area and A2 a rural area), cross-scene testing yields slightly lower performance than training on the entire scene. These results demonstrate strong generalization to different regions in real-world data.

Computational cost comparison. We conducted test experiments on UAVD4L-LoDv2 *in-Traj.* queries using the NVIDIA GeForce RTX 4090 device, running five trials and averaging the results. We recorded the average peak CUDA usage as well as the average inference time. The details are provided in Tab. 7.

5.4. Failure Cases

Our method encounters challenges when images are captured too close to buildings, leading to images that are predominantly occupied by the building itself (as shown in Fig. 5). However, it is important to note that such situations are rare in our drone-based visual localization

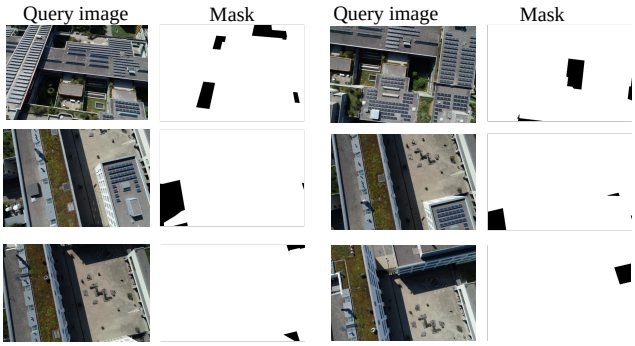


Figure 5. **Failure cases.** The masks reveal a high proportion of building (white regions), which increases the difficulty of accurate pose localization.

	Train	Test	Recall (%)		
	Synthesis	*Real*	2m-2°	3m-3°	5m-5°
LoD-Loc v2	A2	A1	89.44	95.61	98.30
	A1, A2	A1	96.46	99.08	99.79
	A1	A2	93.00	98.20	99.80
	A1, A2	A2	97.94	99.79	99.90

Table 6. **Cross-scene generalization**. We evaluate the generalization ability of our method by training and testing on different regions. The regional divisions are illustrated in Figure 6, where each region is marked with a specific color and letter.

Method	Mem.(Mb)	Time(s)
LoD-Loc	4810	**0.34**
LoD-Loc v2	**853**	2.15

Table 7. **Computational cost comparison.**

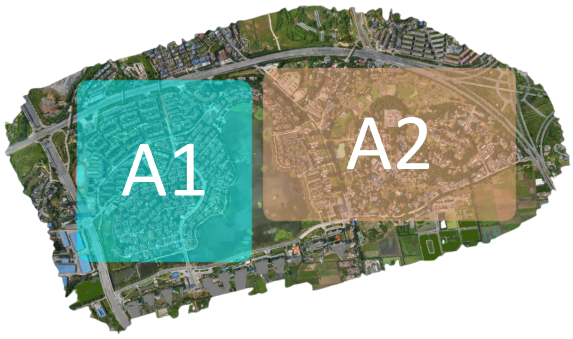
Δ	Variant	*in-Traj.*	*out-of-Traj.*
30°	no select	83.0 / 86.7 / 88.5	79.7 / 81.4 / 82.3
	no refine	49.2 / 59.5 / 97.0	41.3 / 62.3 / 97.9
	full	**95.1 / 98.5 / 99.7**	**96.4 / 99.7 / 100**
60°	no select	39.3 / 43.3 / 45.6	42.9 / 44.0 / 45.1
	no refine	39.8 / 58.4 / 97.0	40.8 / 62.0 / 97.5
	full	**93.8 / 98.6 / 99.8**	**95.1 / 99.8 / 100**

Table 8. **Ablation study on orientation priors.**

task. This scenario accounts for only 0.32% (12/3796) of the UAVD4L-LoDv2 dataset and 9.82% (107/1091) of the Swiss-EPFLv2 dataset, amounting to 2.43% (119/4887) overall.

5.5. Visualization of Results

Fig. 7 illustrates the building masks extracted by different segmentation modules. The SAM2-Unet adopted in our work achieves superior segmentation results.



UAVD4L-LoD v2

Figure 6. **Region of training and testing.** We use boxes with different colors and symbols to denote different regions.

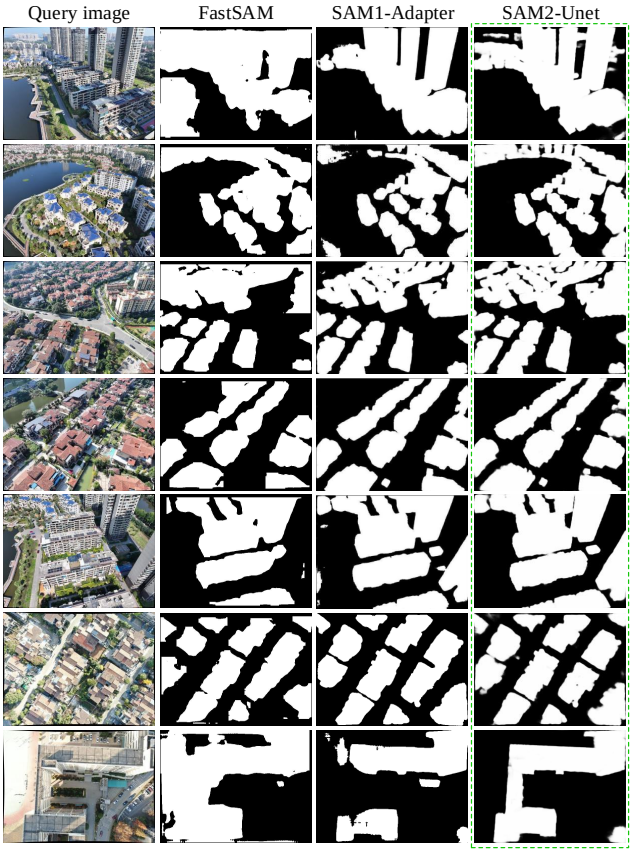


Figure 7. **Segmentation results of different segmentation modules.**

References

[1] Dji zenmuse h20 series, . https://enterprise.dji.com/cn/zenmuse-h20-series?site=enterprise&from=nav. 2

[2] Dji m300 rtk, . https://enterprise.dji.com/cn/

matrice-300. 2

[3] Dp modeler. https://www.whulabs.com/DPModeler/index.aspx. 1

[4] Mavic 3 pro. https://www.dji.com/cn/mavic-3-pro. 2

[5] Openscenegraph. http://www.openscenegraph.com/. 3

[6] swissbuildings3d. https://www.swisstopo.admin.ch/en/landscape-model-swissbuildings3d-2-0. 1

[7] Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. Roma: Robust dense feature matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19790–19800, 2024. 2

[8] Deng-Ping Fan, Ge-Peng Ji, Tao Zhou, Geng Chen, Huazhu Fu, Jianbing Shen, and Ling Shao. Pranet: Parallel reverse attention network for polyp segmentation. In *International conference on medical image computing and computer-assisted intervention*, pages 263–273. Springer, 2020. 2

[9] Victor Fragoso, Joseph DeGol, and Gang Hua. gdls*: Generalized pose-and-scale estimation given scale and gravity priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 4

[10] Ruozhen He, Jiaying Lin, and Rynson WH Lau. Efficient mirror detection via multi-level heterogeneous learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 790–798, 2023. 2

[11] Songtao Liu, Di Huang, et al. Receptive field block net for accurate and fast object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 385–400, 2018. 2

[12] Simon Lynen, Bernhard Zeisl, Dror Aiger, Michael Bosse, Joel Hesch, Marc Pollefeys, Roland Siegwart, and Torsten Sattler. Large-scale, real-time visual–inertial localization revisited. *The International Journal of Robotics Research*, 39 (9):1061–1084, 2020. 4

[13] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2, 3

[14] Vojtech Panek, Zuzana Kukelova, and Torsten Sattler. Visual localization using imperfect 3d models from the internet. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13175–13186, 2023. 2

[15] Zhongxi Qiu, Yan Hu, Heng Li, and Jiang Liu. Learnable ophthalmology sam. *arXiv preprint arXiv:2304.13425*, 2023. 2

[16] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 2

[17] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 2

[18] Chaitanya Ryali, Yuan-Ting Hu, Daniel Bolya, Chen Wei, Haoqi Fan, Po-Yao Huang, Vaibhav Aggarwal, Arkabandhu Chowdhury, Omid Poursaeed, Judy Hoffman, et al. Hiera: A hierarchical vision transformer without the bells-and-whistles. In *International Conference on Machine Learning*, pages 29441–29454. PMLR, 2023. 2

[19] Paul-Edouard Sarlin, Ajaykumar Unagar, Måns Larsson, Hugo Germain, Carl Toft, Victor Larsson, Marc Pollefeys, Vincent Lepetit, Lars Hammarstrand, Fredrik Kahl, and Torsten Sattler. Back to the Feature: Learning Robust Camera Localization from Pixels to Pose. In *CVPR*, 2021. 3

[20] Paul-Edouard Sarlin, Daniel DeTone, Tsun-Yi Yang, Armen Avetisyan, Julian Straub, Tomasz Malisiewicz, Samuel Rota Bulo, Richard Newcombe, Peter Kontschieder, and Vasileios Balntas. Orienternet: Visual localization in 2d public maps with neural matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21632–21642, 2023. 4

[21] Paul-Edouard Sarlin, Eduard Trulls, Marc Pollefeys, Jan Hosang, and Simon Lynen. Snap: Self-supervised neural maps for visual positioning and semantic understanding. *Advances in Neural Information Processing Systems*, 36, 2024. 4

[22] Gabriele Trivigno, Carlo Masone, Barbara Caputo, and Torsten Sattler. The unreasonable effectiveness of pretrained features for camera pose refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12786–12798, 2024. 2, 3

[23] Rouwan Wu, Xiaoya Cheng, Juelin Zhu, Yuxiang Liu, Maojun Zhang, and Shen Yan. Uavd4l: A large-scale dataset for uav 6-dof localization. In *2024 International Conference on 3D Vision (3DV)*, pages 1574–1583. IEEE, 2024. 1, 2, 3

[24] Olaf Wysocki, Benedikt Schwab, Christof Beil, Christoph Holst, and Thomas H Kolbe. Reviewing open data semantic 3d city models to develop novel 3d reconstruction methods. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48:493–500, 2024. 2

[25] Xinyu Xiong, Zihuang Wu, Shuangyi Tan, Wenxue Li, Feilong Tang, Ying Chen, Siying Li, Jie Ma, and Guanbin Li. Sam2-unet: Segment anything 2 makes strong encoder for natural and medical image segmentation. *arXiv preprint arXiv:2408.08870*, 2024. 2

[26] Shen Yan, Yu Liu, Long Wang, Zehong Shen, Zhen Peng, Haomin Liu, Maojun Zhang, Guofeng Zhang, and Xiaowei Zhou. Long-term visual localization with mobile sensors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17245–17255, 2023. 4

[27] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *CVPR*, 2024. 3

[28] Juelin Zhu, Shen Yan, Long Wang, zhang shengYue, Yu Liu, and Maojun Zhang. Lod-loc: Aerial visual localization using

lod 3d map with neural wireframe alignment. In *Advances in Neural Information Processing Systems*, pages 119063–119098, 2024. 1, 2, 3