Spatial-SAM: Spatially Consistent 3D Electron Microscopy Segmentation with SDF Memory and Semi-Supervised Learning

Supplementary Material

1. More Implementation Details

Hyperparameters and Environment. We implement Spatial-SAM in PyTorch and run training/inference on a single NVIDIA RTX 3090 GPU. For model optimization, we set the sampling probability of slices with annotations to $p = 0.8$, and the Eikonal loss weight to $\lambda = 0.5$.

Data Augmentation. We also apply a series of data augmentation techniques to improve model robustness. In particular, random brightness adjustment is used to simulate the uneven illumination commonly observed across different regions or slices of EM images, while random flipping, random rotation, and elastic deformation are adopted to enhance the generalization ability of the model to diverse morphological variations.

Toolset Implementation. We provide a full-process toolset based on the Napari plugin, covering:

- **Interactive Segmentation**: Supports fast annotation of 2D or 3D slices using points and box prompts;
- **Model Training**: Trains Spatial-SAM model based on a small number of annotated slices to adapt to new datasets;
- **Fully Automatic Segmentation**: Calls Spatial-SAM to perform semantic/instance segmentation on 3D volumes;
- **Annotation Correction and Retraining**: Supports interactive correction of automatic results and iterative model optimization;
- **Hardware Adaptation**: Allows users to set resolution and memory usage based on device conditions.

Input Resolution. All 3D volumes are processed as subvolumes of size $1024 \times 1024 \times 1024$: high-resolution datasets are partitioned into tiles of this size, while volumes with smaller spatial extents are resampled (and, when necessary, upsampled) to $1024 \times 1024 \times 1024$ in our method.

Few-shot Annotations and Pseudo-Labels. Within each dataset, we uniformly sample 1/64 of the 2D slices that contain foreground objects as few-shot annotations. Ground-truth masks on these slices are used to simulate SAM2-assisted interactive segmentation with light manual correction. The corrected masks are then used as conditional frames to generate pseudo-labels for the remaining slices in the subsequent semi-supervised training.

Baseline Protocols. To ensure fairness, we follow official implementations and training hyperparameters for all baselines. For semi-supervised methods[2, 6, 8], we adopt U-Net as the backbone consistent with their protocols and train using the same few-shot annotated slices as in our method. For baseline 3D methods and SAM-based approaches[1, 3, 4, 9], anisotropic volumes are resampled to approximate isotropy. Other 2D methods are processed slice-by-slice[2, 6, 8]. For all baselines, we adopt the input resolutions recommended by their official implementations.

2. Dataset Details and Splits

OpenOrganelle (Mouse Liver). The OpenOrganelle mouse liver dataset [5, 13] provides complete 3D electron microscopy imaging of hepatocytes, acquired using enhanced focused ion beam scanning electron microscopy (FIB-SEM) with an isotropic voxel resolution of $8\,\mathrm{nm}$. The dataset includes annotations of cellular structures such as mitochondria and nuclei. We constructed the mitochondrial and nucleus segmentation datasets by cropping 14 and 9 subvolumes with a voxel size of $1024 \times 1024 \times 1024$. For the mitochondrial dataset we used the first 9 subvolumes for training and the remaining 5 for validation; for the nucleus dataset we used the first 4 for training and the remaining 5 for validation.

MitoEM. The MitoEM dataset [12] contains two sets of volumetric images, one from rat (MitoEM-R) and one from human (MitoEM-H) tissue. Each volume covers $30 \times 30 \times 30\,\mu m^3$ at a voxel resolution of $30 \times 8 \times 8\,\mathrm{nm}$, comprising 1000 consecutive electron microscopy sections with precise mitochondrial instance annotations. The original training, validation, and test sets follow a $4{:}1{:}5$ split. The spatial resolution of each slice is 4096×4096. Ground-truth annotations are publicly available for the training and validation sets. In our experiments, we use the official training set for training and the validation set for evaluation.

3. Supplementary Results

Fig. S1 and S2 provide supplementary visualizations that complement Fig. 5 and Fig. 6 from the main paper, respectively.

Additional Evaluation Metrics. Table S1 reports voxel-wise precision and recall, complementing the Dice/mIoU results in the main paper. Across all datasets, Spatial-SAM achieves consistently high precision while maintaining strong recall. For instance, on MitoEM-R it reaches 96.40% precision and 92.62% recall, improving recall by +2.68 points over μSAM and by +8.61 points over Cellpose-SAM, indicating fewer missed mitochondria without inflating false positives. Unlike some semi-supervised approaches that can exhibit a pronounced precision–recall imbalance on particular benchmarks (e.g., CPS U-Net attains 99.58% precision but only 56.08% recall on OOMLN),

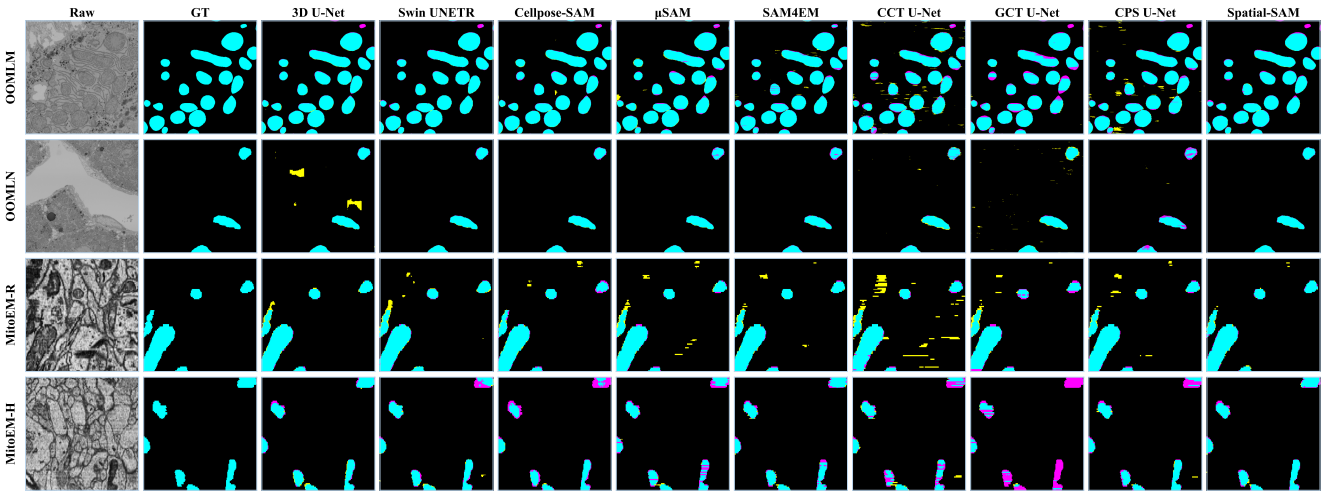

Figure S1. Supplementary visualization of segmentation results on x-z plane.

Figure S2. Supplementary 3D visualization comparison of different methods on mitochondria.

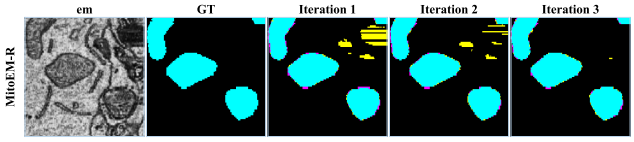

Figure S3. Pseudo-label evolution on MitoEM-R across training iterations. As training proceeds, false positives and false negatives are progressively reduced, indicating improved pseudo-label quality.

Spatial-SAM preserves a more consistent balance between precision and recall across all evaluated datasets, demonstrating its robustness with limited supervision.

Table S2 evaluates boundary quality using the average symmetric surface distance and the 95th-percentile Hausdorff distance (HD95), two standard surface-based metrics for biomedical image segmentation [11]. The average surface distance measures the mean bidirectional distance (in nm) between predicted and reference boundaries, while HD95 summarizes the worst-case deviations (in nm) after discarding the most extreme 5% of surface outliers, which

Table S1. **Comparison of precision and recall on different datasets (%).** The **best** and second-best results for each metric are highlighted.

Method	OOMLM		OOMLN		MitoEM-R		MitoEM-H	
	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
3D U-Net [3]	86.07	96.86	87.48	**99.11**	92.37	93.80	90.47	**92.92**
Swin UNETR [4]	96.79	96.69	97.97	98.48	93.90	89.31	76.94	88.89
Cellpose-SAM [9]	96.79	96.10	98.81	98.05	95.93	84.01	90.35	81.33
μSAM [1]	96.23	95.63	98.63	88.58	95.83	89.94	93.66	85.08
SAM4EM [10]	96.30	**97.20**	98.79	94.54	95.08	**95.17**	91.68	90.67
CCT U-Net [8]	88.77	95.04	93.84	81.97	72.61	89.05	63.78	78.69
GCT U-Net [6]	96.83	93.40	92.37	95.03	95.15	83.05	93.66	60.99
CPS U-Net [2]	95.62	95.88	**99.58**	56.08	94.57	92.25	91.41	86.38
Spatial-SAM	**97.32**	95.71	99.03	97.26	**96.40**	92.62	**93.74**	86.82

Table S2. **Comparison of average surface distance and 95th-percentile Hausdorff distance on different datasets (nm).** The **best** and second-best results for each metric are highlighted (lower is better).

Method	OOMLM		OOMLN		MitoEM-R		MitoEM-H	
	Avg Dist	HD95	Avg Dist	HD95	Avg Dist	HD95	Avg Dist	HD95
3D U-Net [3]	437.94	2651.61	961.33	8886.12	127.66	1086.18	88.84	955.18
Swin UNETR [4]	**13.49**	**25.26**	145.70	1506.97	49.60	578.44	186.15	1392.11
Cellpose-SAM [9]	23.71	77.79	**36.44**	**65.51**	100.93	661.20	109.17	729.53
μSAM [1]	23.29	132.21	214.69	2771.11	19.03	131.28	33.50	381.12
SAM4EM [10]	33.79	455.76	107.72	370.61	31.84	307.64	36.18	430.87
CCT U-Net [8]	197.29	1601.93	659.70	5384.88	263.27	1777.59	345.64	1671.54
GCT U-Net [6]	23.65	94.84	835.26	6640.49	32.06	167.30	66.21	532.50
CPS U-Net [2]	58.52	727.80	469.99	3552.18	31.20	321.06	46.18	521.85
Spatial-SAM	14.37	31.97	57.98	124.35	**14.55**	**47.97**	**29.12**	**259.46**

Table S3. **Comparison of surface Dice at 16nm on different datasets (%).** The **best** and second-best results for each metric are highlighted.

Method	OOMLM	OOMLN	MitoEM-R	MitoEM-H
3D U-Net [3]	80.14	56.76	85.08	**87.31**
Swin UNETR [4]	**94.55**	54.78	82.08	71.40
Cellpose-SAM [9]	92.89	54.88	74.92	72.76
μSAM [1]	88.77	38.65	85.43	81.40
SAM4EM [10]	92.23	36.02	**91.05**	84.86
CCT U-Net [8]	69.00	16.13	51.81	44.44
GCT U-Net [6]	83.87	21.02	72.75	57.90
CPS U-Net [2]	86.78	9.74	85.30	78.85
Spatial-SAM	92.03	**56.79**	90.42	85.75

is more robust than the classical maximum Hausdorff distance. Lower values indicate more accurate and less erratic boundaries. Across all evaluated datasets, Spatial-SAM ranks among the top two methods for both average surface distance and HD95. Notably, it achieves the best surface distances on MitoEM-R and MitoEM-H, and remains highly competitive on OpenOrganelle subsets (second-best on OOMLM and OOMLN). For example, on MitoEM-R it reduces the average surface distance from 19.03 nm (best baseline, μSAM) to 14.55 nm and HD95 from 131.28 nm to 47.97 nm, corresponding to substantially tighter and more stable mitochondrial surfaces. On OOMLN, Spatial-SAM delivers a competitive average distance and a large reduction in HD95 relative to semi-supervised and 3D baselines, while approaching the strongest 2D SAM-based model.

We further report the surface Dice similarity coefficient [7] at a 16 nm tolerance in Table S3. In cases where some methods produce substantial long-range segmentation errors and/or much noise on certain datasets, average surface distance and HD95 may not fully capture boundary quality; therefore we additionally include the surface Dice to reflect practical boundary agreement. Spatial-SAM attains surface Dice values that are uniformly within 2.6 percentage points of the highest score

on every dataset (differences: 2.52 on OOMLM, 0.00 on OOMLN, 0.63 on MitoEM-R, 1.56 on MitoEM-H), evidencing consistently strong boundary alignment across domains. Simultaneously, it delivers large gains over all semi-supervised baselines: +23.03/+8.16/+5.25 (OOMLM), +40.66/+35.77/+47.05 (OOMLN), +38.61/+17.67/+5.12 (MitoEM-R), and +41.31/+27.85/+6.90 (MitoEM-H) percentage points versus CCT/GCT/CPS respectively. These results show that Spatial-SAM provides boundary performance effectively on par with state-of-the-art fully supervised baselines while markedly surpassing semi-supervised approaches under limited annotation.

Table S4. Comparison of inference efficiency and resource consumption across different methods (Time: seconds; GPU RAM: graphics processing unit memory, GB; RAM: system memory, GB).

Method	2D Patch	3D Patch	Time	GPU RAM	RAM
3D U-Net	–	128^3	73	2.73	11.02
3D U-Net*	–	144^3	109	5.30	15.47
μSAM	512^2	–	608	4.19	18.59
Cellpose-SAM	256^2	–	4783	4.15	12.60
U-Net	512^2	–	99	1.33	2.90
Swin UNETR	–	96^3	1842	13.35	47.85
SAM4EM	512^2	–	182	1.41	4.46
Spatial-SAM	1024^2	144^3	106	9.41	13.68

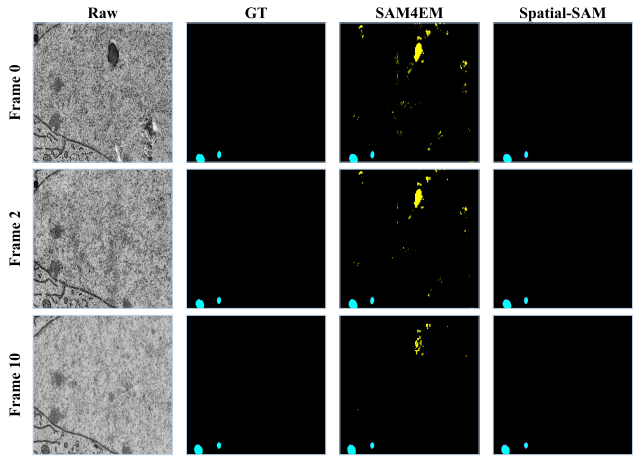

Figure S4. Visual comparison of error accumulation caused by sequential memory (SAM4EM) vs. geometry-aware SDF guidance (Spatial-SAM).

Performance Analysis. We evaluate the inference performance of different segmentation models on a volumetric electron microscopy dataset of size $1024 \times 1024 \times 1024$. All experiments are conducted on an NVIDIA RTX 3090 GPU. It is noted that different patch sizes, overlap and batch sizes could have a significant impact on the runtime and resource consumption, and our reported results serve as a representative example. For μSAM and Cellpose-SAM are tested using their default input resolutions as patch sizes (512^2 and 256^2, respectively). Our proposed method integrates both 2D and 3D pathways with patch sizes of 1024^2 and 144^3. Cellpose-SAM uses a batch size of 32, U-Net uses a batch size of 8, while other methods use a batch size of 1. The GPU RAM and RAM usages are measured as the peak consumption during the full inference of the volume.

In terms of efficiency, the 3D U-Net without overlap achieves the shortest runtime of 73.71 seconds due to the absence of redundant computation, whereas enabling overlap increases the runtime to 109.48 seconds and raises memory consumption accordingly. μSAM and Cellpose-SAM, both operating slice by slice in 2D, require substantially longer inference times of 608 seconds and 4783 seconds. Our method completes inference in 106 seconds, achieving a balance between 3D volumetric processing and 2D contextual efficiency. Compared with the SAM-based methods, our approach provides a significant speed advantage when scaled to volumetric data, demonstrating improved computational efficiency for large-scale 3D inference. Regarding training overhead, Spatial-SAM requires approximately 14.79 hours on MitoEM-R using a single NVIDIA RTX 3090.

4. Additional Discussion

Sequential Memory vs. SDF Guidance. While SAM2's stateful memory is intrinsically prone to error accumulation, SAM4EM exacerbates this issue by employing a momentum-updated feature memory. This momentum mechanism mathematically induces a delayed response, which results in slower memory updates compared to SAM2 and causes spatial misalignments during morphological changes. As shown in Fig. S4, when SAM4EM misclassifies an isolated slice-level artifact as a mitochondrion at Frame 0, the slow momentum update severely amplifies the error accumulation. This causes the false positive to linger and leave residual segmentations at Frame 10, long after the artifact has disappeared.

Spatial-SAM circumvents this issue by utilizing a geometry-driven signed distance field (SDF) memory. Because transient 2D artifacts rarely form coherent 3D structures across slices, the spatially continuous 3D SDF intrinsically acts as a structural filter. Consequently, Spatial-SAM not only suppresses the initial artifact misclassification at Frame 0 but also completely avoids the delayed reactions and error propagation seen in sequential memory mechanisms, demonstrating the robustness of explicit 3D spatial guidance.

Transferability and Adaptation Cost. Although the SAM2 module provides strong promptable priors, transferring to a new EM domain still typically requires fine-tuning or retraining due to the limited out-of-domain generalization of the U-Net SDF branch. To reduce adaptation over-

head, parameter-efficient fine-tuning (PEFT), such as LoRA and partial encoder freezing, represents a promising future direction. We expect such strategies to potentially maintain competitive segmentation quality while lowering both compute and memory requirements during model adaptation.

Applicability to More Complex Structures. The proposed pipeline is generally applicable to binary segmentation tasks beyond mitochondria. The SDF memory enforces geometry-aware cross-slice consistency, while the SAM2 module captures appearance variations. For objects with more complex topology or larger inter-slice deformation, we expect the same mechanism to remain beneficial, potentially with a larger memory neighborhood K and/or a moderately increased slice-level annotation ratio.

Multi-class Extension. For multi-class segmentation, a straightforward extension is to model class-wise SDFs in a multi-channel representation. A more compact alternative is to keep a shared foreground-background SDF memory for geometric guidance and add a semantic class head in the SAM2 branch for class discrimination. This design decouples geometric consistency from semantic categorization and can retain the efficiency advantages of the current framework.

References

[1] Anwai Archit, Luca Freckmann, Sushmita Nair, Nabeel Khalid, Paul Hilt, Vikas Rajashekar, Marei Freitag, Carolin Teuber, Melanie Spitzner, Constanza Tapia Contreras, et al. Segment anything for microscopy. *Nature Methods*, 22(3): 579–591, 2025. 1, 3

[2] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *CVPR*, pages 2613–2622, 2021. 1, 3

[3] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pages 424–432. Springer, 2016. 1, 3

[4] Ali Hatamizadeh, Vishwesh Nath, Yucheng Tang, Dong Yang, Holger R Roth, and Daguang Xu. Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images. In *MICCAI Brainlesion Workshop*, pages 272–284, 2021. 1, 3

[5] Larissa Heinrich, Davis Bennett, David Ackerman, Woohyun Park, John Bogovic, Nils Eckstein, Alyson Petruncio, Jody Clements, Song Pang, C Shan Xu, et al. Whole-cell organelle segmentation in volume electron microscopy. *Nature*, 599(7883):141–146, 2021. 1

[6] Zhanghan Ke, Di Qiu, Kaican Li, Qiong Yan, and Rynson WH Lau. Guided collaborative training for pixel-wise semi-supervised learning. In *ECCV*, pages 429–445, 2020. 1, 3

[7] Stanislav Nikolov, Sam Blackwell, Alexei Zverovitch, Ruheena Mendes, Michelle Livne, Jeffrey De Fauw, Yojan Patel, Clemens Meyer, Harry Askham, Bernardino Romera-Paredes, et al. Deep learning to achieve clinically applicable segmentation of head and neck anatomy for radiotherapy. *arXiv preprint arXiv:1809.04430*, 2018. 3

[8] Yassine Ouali, Céline Hudelot, and Myriam Tami. Semi-supervised semantic segmentation with cross-consistency training. In *CVPR*, pages 12674–12684, 2020. 1, 3

[9] Marius Pachitariu, Michael Rariden, and Carsen Stringer. Cellpose-sam: superhuman generalization for cellular segmentation. *bioRxiv*, pages 2025–04, 2025. 1, 3

[10] Uzair Shah et al. Sam4em: Efficient memory-based two stage prompt-free segment anything model adapter for complex 3d neuroscience electron microscopy stacks. In *CVPR*. 3

[11] Abdel Aziz Taha and Allan Hanbury. Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool. *BMC medical imaging*, 15(1):29, 2015. 2

[12] Donglai Wei, Zudi Lin, Daniel Franco-Barranco, Nils Wendt, Xingyu Liu, Wenjie Yin, Xin Huang, Aarush Gupta, Won-Dong Jang, Xueying Wang, et al. Mitoem dataset: large-scale 3d mitochondria instance segmentation from em images. In *MICCAI*, pages 66–76, 2020. 1

[13] C Shan Xu, Song Pang, Gleb Shtengel, Andreas Müller, Alex T Ritter, Huxley K Hoffman, Shin-ya Takemura, Zhiyuan Lu, H Amalia Pasolli, Nirmala Iyer, et al. An open-access volume electron microscopy atlas of whole cells and tissues. *Nature*, 599(7883):147–151, 2021. 1