

SegMASt3R: Geometry Grounded Segment Matching

Supplementary Material

Rohit Jayanti^{1*} Swayam Agrawal^{1*} Vansh Garg^{1*} Siddharth Tourani^{2,3}
 Muhammad Haris Khan³ Sourav Garg⁴ Madhava Krishna¹

¹IIIT Hyderabad ²University of Heidelberg ³MBZUAI ⁴Independent

Project Page: <https://segmast3r.github.io/>

1 Cross-Dataset Generalization on Replica

To evaluate cross-dataset generalization, we train our model on ScanNet++[22] and directly evaluate it on the Replica dataset[20], without any fine-tuning. Table 1 reports a detailed comparison against competitive baselines, highlighting the robustness of our approach under distribution shift.

Type	Method	0°–45°			45°–90°			90°–135°			135°–180°		
		AUPRC	R@1	R@5	AUPRC	R@1	R@5	AUPRC	R@1	R@5	AUPRC	R@1	R@5
Local Feature Matching (LFM)	SP-LG [4, 13]	64.69	67.89	72.3	46.89	51.59	56.1	21.86	28.74	32.83	21.82	28.72	30.9
	GiM-DKM [19, 5]	75.46	81.38	83.51	67.26	74.08	76.8	43.94	55.8	61.41	35.37	49.41	53.86
	RoMa [6]	77.3	84.06	86.86	69.35	79.08	84	46.69	68.49	77.48	31.79	60.48	68.94
	MASt3R [12]	78.2	86.5	89.4	69.5	77.6	81.0	48.0	60.4	64.6	32.5	49.0	54.1
Segment Matching (SegMatch)	SAM2 [18]	80.09	82.28	84.61	54.58	62.03	65.41	40.69	53.72	56.67	37.78	54.59	56.42
	DINOv2 [14]	55.85	74.25	96.55	31.12	59.64	92.84	21.71	57.68	92.33	17.29	59.28	89.64
	SegVLAD [7]	67.49	77.65	95.52	46.84	66.46	92.26	40.21	67.23	91.81	37.8	69.6	89.15
	MASt3R [12]	52.2	57.5	81.2	39.1	51.0	78.6	23.6	45.9	77.2	17.2	43.8	75.7
Ours	SegMASt3R	95.0	96.0	98.6	86.2	91.2	96.4	73.4	85.2	95.7	68.4	83.8	94.8

Table 1: Performance of additional baselines across pose-bins on Replica[20]. Blue cells mark the best scores; Orange cells mark the second-best.

2 Local-Feature Matching (LFM) Baseline Setup

A range of local feature matchers (LFMs) exist at varying densities—SuperPoint [4] (sparse), LoFTR [21] (semi-dense), and RoMa or MASt3R [6, 12] (dense). We leverage the EarthMatch (IMM) [1] toolkit, which provides a unified and modular interface for applying a wide range of local feature matchers. This abstraction significantly simplifies the integration of keypoint-based methods into our pipeline, allowing us to systematically evaluate segment correspondence performance across multiple matcher densities.

We use a *voting-based aggregation* scheme (Algorithm 1) to derive segment-level correspondences via LFMs. Given two images and their respective binary segment masks $\mathcal{M}_0 \in \{0, 1\}^{M \times H \times W}$ and $\mathcal{M}_1 \in \{0, 1\}^{N \times H \times W}$, a chosen LFM \mathcal{A} produces K matched keypoint pairs. Each matched pair votes for the segment it belongs to in the source and target masks, incrementing a *vote matrix* $V \in \mathbb{Z}^{M \times N}$.

Finally, segment-level correspondences are assigned by selecting, for each source segment m , the target segment n with the highest vote count. Segments with no valid votes are marked as unmatched (assigned label -1). This voting mechanism provides a simple yet effective bridge between pixel-level keypoint matches and object-level segment associations.

*Equal Contribution

Algorithm 1 Segment Correspondence via Keypoint Voting

Require: Images I_0, I_1 ; segment masks $\mathcal{M}_0 \in \{0, 1\}^{M \times H \times W}$, $\mathcal{M}_1 \in \{0, 1\}^{N \times H \times W}$; keypoint matcher \mathcal{A} ; max matches K

Ensure: Segment assignment vector $\hat{\mathcal{C}} \in \mathbb{N}^M$ (with -1 indicating no confident match)

```
1:  $\{(x_i^0, y_i^0), (x_i^1, y_i^1)\}_{i=1}^K \leftarrow \mathcal{A}(I_0, I_1)$ 
2: Initialize vote matrix  $V \in \mathbb{Z}^{M \times N} \leftarrow 0$ 
3: for  $i = 1$  to  $K$  do
4:   Find  $m_i$  such that  $\mathcal{M}_0[m_i, y_i^0, x_i^0] = 1$ 
5:   Find  $n_i$  such that  $\mathcal{M}_1[n_i, y_i^1, x_i^1] = 1$ 
6:   if  $m_i$  and  $n_i$  are valid then
7:      $V[m_i, n_i] \leftarrow V[m_i, n_i] + 1$ 
8:   end if
9: end for
10: for  $m = 1$  to  $M$  do
11:   if  $\sum_n V[m, n] = 0$  then
12:      $\hat{\mathcal{C}}[m] \leftarrow -1$  ▷ No confident match
13:   else
14:      $\hat{\mathcal{C}}[m] \leftarrow \arg \max_n V[m, n]$ 
15:   end if
16: end for
17: return  $\hat{\mathcal{C}}$ 
```

3 Additional Qualitative Results on ScanNet++ Dataset

We present additional qualitative results (Figure 1) on the ScanNet++ dataset[22], highlighting our method’s robustness to perceptual instance aliasing and wide-baseline viewpoint changes. These examples further demonstrate accurate segment correspondences despite challenging geometric and appearance ambiguities.

4 Details on Instance-Mapping

Method	office0		office1		office2		office3		office4		room0		room1		room2	
	AP / AP@50	AP / AP@50	AP / AP@50	AP / AP@50	AP / AP@50	AP / AP@50	AP / AP@50	AP / AP@50	AP / AP@50	AP / AP@50	AP / AP@50	AP / AP@50	AP / AP@50	AP / AP@50	AP / AP@50	AP / AP@50
ConceptGraphs (MobileSAM Masks) [9]	11.84 / 28.43	20.31 / 43.79	8.63 / 22.82	8.07 / 22.83	9.46 / 24.73	12.23 / 34.34	5.83 / 12.96	7.83 / 23.82								
ConceptGraphs (GT Masks) [9]	43.53 / 69.68	22.48 / 40.71	43.46 / 60.69	32.06 / 53.44	39.63 / 68.22	44.89 / 69.64	17.96 / 36.53	25.93 / 43.63								
SegMASt3R(Ours, GT Masks)	79.93 / 87.17	54.89 / 64.42	64.00 / 85.50	58.02 / 79.93	67.48 / 85.01	71.02 / 91.22	64.09 / 85.50	56.35 / 76.66								

Table 2: Class-Agnostic instance-mapping performance (AP and AP@50) on Replica scenes, shown in percentage. The best value in each column is highlighted in blue.

By default, ConceptGraphs [9] is evaluated on the scanned RGB-D trajectories of the Replica [20] as provided by Nice-SLAM [25]. However, this version does not include ground-truth instance masks and instead relies on MobileSAM [23] for mask generation. We first evaluate the method using this setup, which corresponds to the first row of Table 2. To assess performance with ground-truth instance segmentation, we use the pre-rendered Replica RGB-D sequence released by Semantic-NeRF [24], which provides 900 RGB-D frames per scene along with ground-truth instance masks and camera poses. We subsample each sequence with a stride of 5 and run ConceptGraphs on a total of 180 frames per scene. This evaluation setting corresponds to the second row of Table 2. We observe that the trend remains consistent, with SegMASt3R continuing to outperform ConceptGraphs on the instance mapping task.

4.1 Evaluation Methodology for Instance Mapping

We assess the accuracy of instance mapping by how well individual object geometries are preserved, *i.e. quality of the instance boundaries and whether they are multi-view consistent*. We follow the class-agnostic evaluation protocol as done by OpenYOLO3D [2], which benchmarks generic 3D

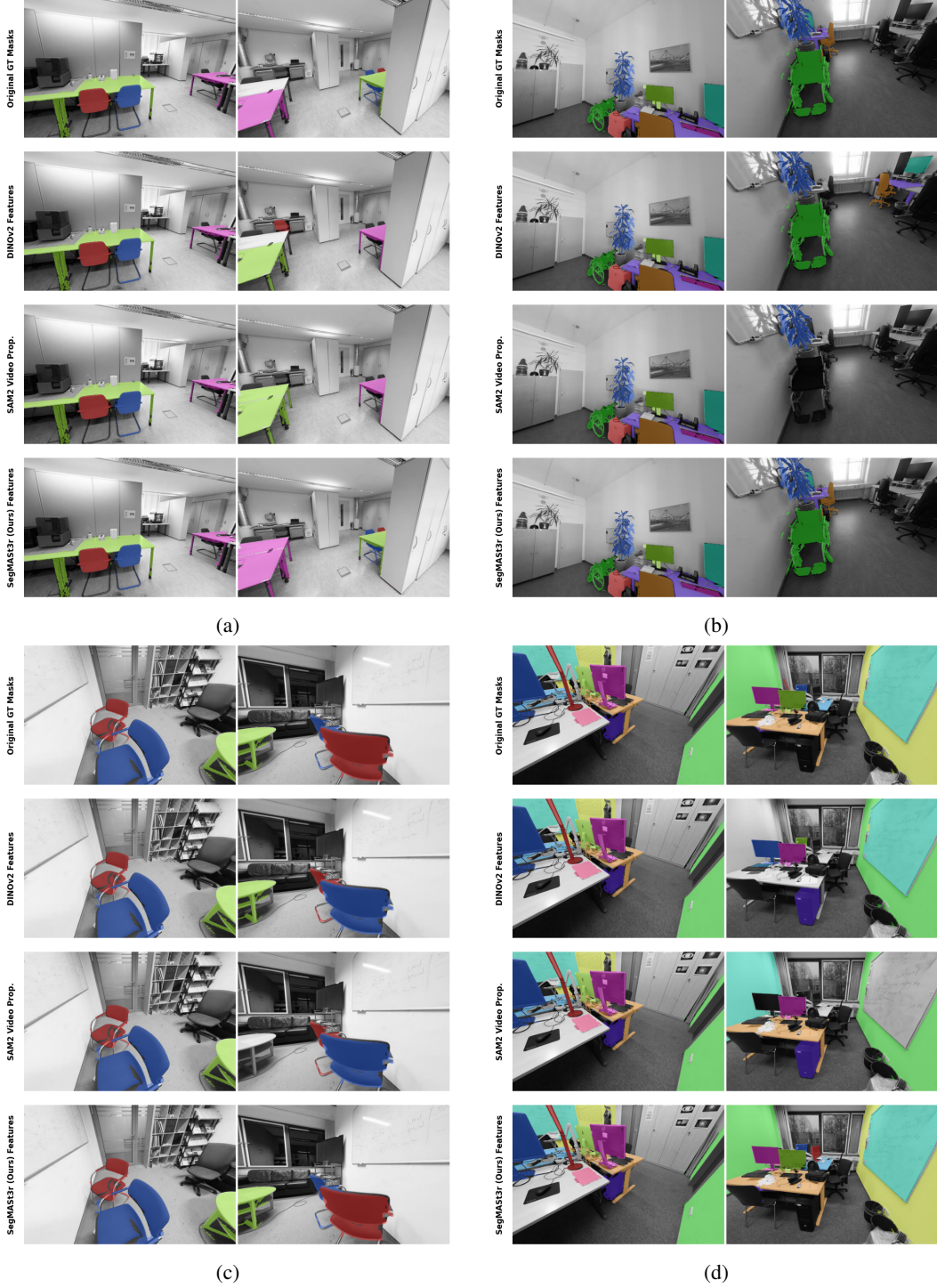


Figure 1: More examples comparing proposed method against DINOv2 [14] and SAM2 [18] for segment-matching on the ScanNet++ Dataset [22] under **Wide-baseline conditions** [135–180° **viewpoint change**]. Both baselines tend to incorrectly assign segment correspondences given an opposite viewing direction-the tables (pink and green) in (a) as well as the chairs (red and blue) in (c). Another failure mode SAM2 specifically exhibits is the inability to propagate masks in challenging view-point change settings as seen for the wheelchair (green) in (b). In contrast, the proposed method demonstrates accurate segment matching, attributed to its conditioning on 3D-aware priors.

Note: Images have been greyed out to improve the visibility of the segments in question.

object instances in the given scene. Formally, given a predicted 3D instance-map point-cloud and the corresponding ground-truth, we compute the Average Precision (AP) over a range of Intersection-over-Union (IoU) thresholds. In particular, we report:

- **AP@0.50** reports precision when IoU = 0.50, measuring correctness at a moderate overlap.
- **AP** is the mean precision over IoU thresholds from 0.50 to 0.95 in steps of 0.05, capturing both coarse and fine alignment.

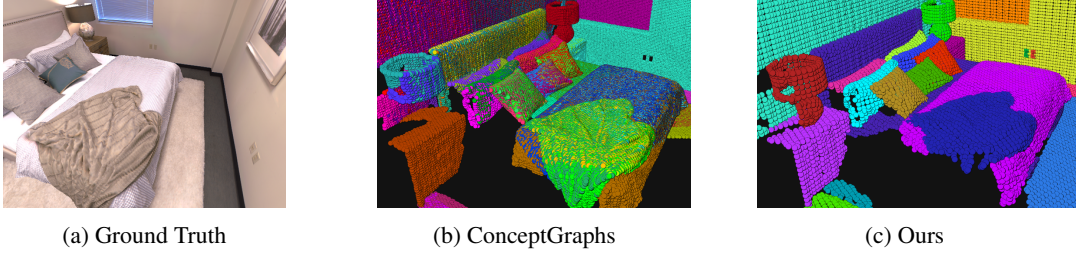


Figure 2: **Qualitative comparison of instance maps.** Each color corresponds to a different object instance. The ground truth (left) provides the closest reference RGB view of the scene from the RGB sequence. ConceptGraphs (middle) tends to over-segment objects, assigning multiple instance IDs to the same object, resulting in fragmented coloring (*e.g. brown duvet covered by green and yellow labels*). Our method (right) produces cleaner and more consistent instance groupings.

4.2 ConceptGraphs for Instance Mapping

We use Concept-Nodes, a lightweight implementation of ConceptGraphs [9]. We generate an instance-level 3D map by aggregating detections over time and associating them with previously observed objects in a global map via semantic and geometric matching.

4.2.1 Object-Centric 3-D Representation

Given a stream of RGB-D observations $\mathcal{I} = \{I_1, \dots, I_T\}$, ConceptGraphs incrementally builds a 3-D scene graph $\mathcal{M}_t = (\mathcal{O}_t)$. Each node $o_j \in \mathcal{O}_t$ is represented by a point cloud \mathbf{P}_{o_j} and a unit-normalised semantic descriptor \mathbf{f}_{o_j} . At time t the incoming frame $I_t = \langle I_t^{\text{rgb}}, I_t^{\text{depth}}, \theta_t \rangle$ (colour, depth, pose) is fused into the map by either updating an existing object or instantiating a new one.

4.2.2 Class-Agnostic 2-D Segmentation

For each frame we extract M class-agnostic masks $\{\mathcal{M}_{t,i}\}_{i=1}^M = \text{Seg}(I_t^{\text{rgb}})$ using YOLO-WORLD [3] and MOBILESAM [23] (or ground truth, when available). Every mask is fed to a CLIP [16] encoder to obtain a visual descriptor $\mathbf{f}_{t,i} = \text{SE}(I_t^{\text{rgb}}, \mathcal{M}_{t,i})$. The masked RGB-D region is back-projected and transformed to the global frame, yielding a point cloud $\mathbf{P}_{t,i}$ paired with $\mathbf{f}_{t,i}$.

4.2.3 Object Association

For every newly detected object candidate $\langle \mathbf{P}_{t,i}, \mathbf{f}_{t,i} \rangle$ we measure its similarity to each map object $\langle \mathbf{P}_j, \mathbf{f}_j \rangle$ that exhibits spatial overlap. Geometric consistency is captured by the *nearest-neighbour ratio*

$$s_{\text{geo}}(i, j) = \text{NNRatio}(\mathbf{P}_{t,i}, \mathbf{P}_j),$$

defined as the fraction of points in the candidate cloud whose closest point in the reference cloud lies within the Euclidean tolerance δ_{nn} . Appearance similarity is obtained from the cosine similarity of the CLIP descriptors, rescaled from $[-1, 1]$ to $[0, 1]$,

$$s_{\text{sem}}(i, j) = \frac{1}{2}(\mathbf{f}_{t,i}^\top \mathbf{f}_j) + \frac{1}{2}.$$

The two scores are linearly blended into a fused similarity

$$s(i, j) = \alpha s_{\text{sem}}(i, j) + (1 - \alpha) s_{\text{geo}}(i, j), \quad \alpha \in [0.1, 0.5],$$

where the weight α balances semantic and geometric evidence and is tuned per scene. Greedy matching assigns each detection to the map object with the highest fused similarity; if this peak score falls below the threshold $\delta_{\text{sim}} \in [0.90, 0.96]$, the detection is instead used to instantiate a new object instance. In practice, we sweep and select δ_{sim} in the range $0.90 \dots 0.96$ to accommodate scene-specific variability in texture and clutter.

4.2.4 Object Fusion

Once a detection i has been matched to map object o_j , its semantic and geometric information is merged into the map. Let n_j denote the number of detections previously fused into o_j and δ_{voxel} the voxel size used for down-sampling.

Semantic update. We maintain a running average of CLIP descriptors so that each new observation contributes equally:

$$\mathbf{f}_j \leftarrow \frac{n_j \mathbf{f}_j + \mathbf{f}_{t,i}}{n_j + 1}, \quad n_j \leftarrow n_j + 1.$$

Geometric update. The point cloud of the object is augmented and then compacted to prevent redundancy:

$$\mathbf{P}_j \leftarrow \text{Downsample}(\mathbf{P}_j \cup \mathbf{P}_{t,i}, \delta_{\text{voxel}}),$$

where $\text{Downsample}(\cdot, \delta_{\text{voxel}})$ groups points whose pairwise distances are below the voxel size ($\delta_{\text{voxel}} = 1$ cm in all experiments) and replaces each group by its centroid.

4.3 SegMASt3R for Instance Mapping

We use SEGMASt3R to directly establish object-level correspondences across image pairs, based on both mask-level appearance features and geometric consistency. Unlike Concept-Graphs, which incrementally associates detections with a growing map, our method operates pairwise and enforces geometric alignment through 3D checks.

Pairwise Mask Matching. Given two RGB-D frames $I_s = \langle I_s^{\text{rgb}}, I_s^{\text{depth}}, \theta_s \rangle$ and $I_t = \langle I_t^{\text{rgb}}, I_t^{\text{depth}}, \theta_t \rangle$ and their respective sets of ground-truth instance masks $\{\mathbf{m}_{s,i}\}_{i=1}^{M_s}$ and $\{\mathbf{m}_{t,j}\}_{j=1}^{M_t}$, we extract visual descriptors for each mask using our method, SegMASt3R:

$$\mathbf{f}_{s,i}, \mathbf{f}_{t,j} = \text{SEGMASt3R}(I_s^{\text{rgb}}, I_t^{\text{rgb}}, \mathbf{m}_{s,i}, \mathbf{m}_{t,j})$$

where $\mathbf{f}_{s,i}, \mathbf{f}_{t,j} \in \mathbb{R}^{24}$ are the mask feature vectors. These features are used to construct a pairwise similarity matrix, and a soft assignment between masks is computed using the Sinkhorn algorithm. Tentative correspondences are obtained by row-wise $\arg \max$ over the resulting transport matrix:

$$j^* = \arg \max_j \hat{M}_{i,j},$$

where $\hat{M} \in \mathbb{R}^{M_s \times M_t}$ is the normalized assignment matrix. Matches assigned to the dustbin (i.e., $j^* = M_t + 1$) are discarded. The remaining matches are considered for geometric consistency.

Geometric Consistency. For each matched pair $(\mathbf{m}_{s,i}, \mathbf{m}_{t,j^*})$, we project both masks into 3D using the depth maps and camera poses to obtain their corresponding point clouds, $\mathbf{p}_{s,i}$ and \mathbf{p}_{t,j^*} . To evaluate geometric consistency, we compute the geometric similarity between the two point-clouds in the same way as done by Concept-Graphs. A match is accepted if $\phi_{\text{geo}}(i, j^*) > 0.5$. All matches failing this geometric filter are rejected.

Evaluation on Replica. We use the RGB-D frames from Semantic-NeRF [24] and subsample each 900-frame sequence with a stride of 20, resulting in 45 frames per scene. We perform pairwise matching across all possible frame pairs (990 in total) and aggregate the correspondences to construct the final instance map. The corresponding results are reported in the third row of Table 2.

5 Additional Details on Object-Relative Topological Navigation

Background: We used RoboHop [8] for benchmarking on the topological navigation task. RoboHop uses a segment-based topological map to enable open-world visual navigation. It extracts semantically-meaningful image segments using SAM [10]. These segments become the nodes of a topological graph, where edges are formed both intra-image—linking segments within the same image based on pixel-level proximity—and inter-image—connecting corresponding segments across consecutive frames. For given query image segments, segment matching is performed using SuperPoint+LightGlue to retrieve matching segment nodes from the map, each of which has a precomputed path length to the given goal. Using these path lengths along with the pixel centers of the image segments, yaw (ψ) is computed as below [15, 8]:

$$\psi = \frac{K}{W} \sum_i w_i (x_i - c); \quad w_i = \frac{e^{\tau l_i}}{\sum_i e^{\tau p_i}} \quad (1)$$

where c is the image center, x_i is the segment center, p_i is the path length, τ is the temperature parameter (set to 5), w_i is the softmax weight per query segment, W is the image width, and K is the proportional gain (set to 0.4). For forward translation, a fixed velocity of $0.5m/s$ is used. We obtained the source code from the original authors of RoboHop [8], and followed [15] to evaluate using HM3Dv0.2 [17]. This benchmark data is based on the validation set of the InstanceImageNav (IIN) challenge [11], where 108 episodes (3 each from the 36 unique environments) are used to define the start position, object goal, and map trajectory. A topological graph is constructed a priori from the map trajectory, which is then used during the execution phase for localization, planning, and control.

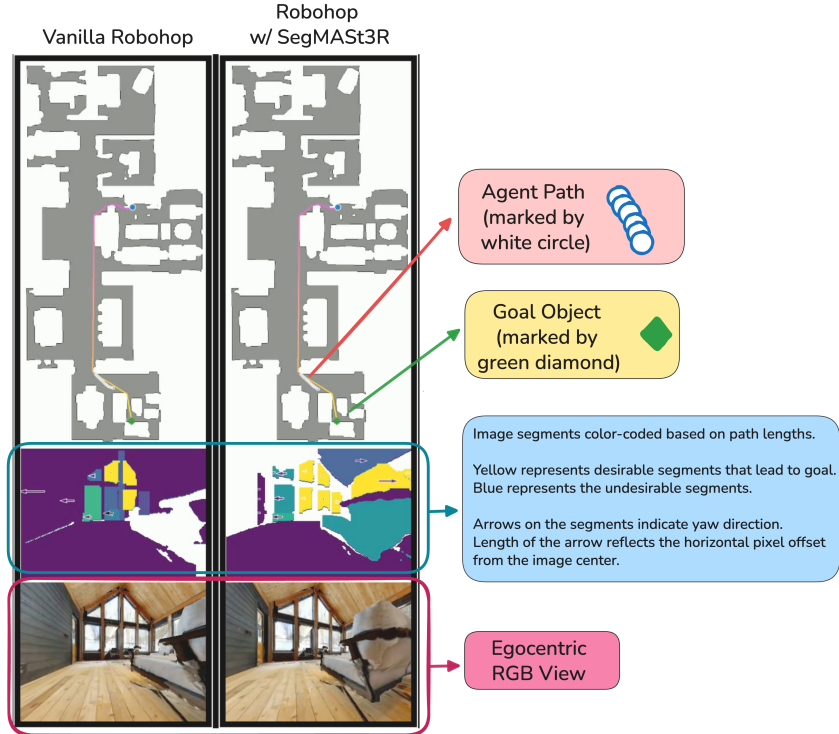


Figure 3: Different components of the image based goal navigation video.

Qualitative Visualizations: In our experiments, we replaced RoboHop’s segment matching during localization with SegMASt3R, and compared vanilla RoboHop with this version. In this supplementary material, we include navigation videos that qualitatively compare the two methods. The left half of the video corresponds to vanilla RoboHop, while the right half corresponds to our SegMASt3R-enhanced version. Here, we provide an overview of the visualization panel of the video. At the top, we show an overhead view of the simulator, where the map trajectory is displayed with a color gradient starting from the blue circle and ending at the green diamond (goal object). The agent is initialized $5m$ away

from the goal along the map trajectory, and its current state is displayed with a white circle. In the middle, we show the image segments which are color-coded based on the path lengths (normalized per image): yellow represents desirable segments (high weight w_i) that lead to the goal and blue represents the undesirable segments. The arrows on the segments indicate yaw direction (ψ), and the length of the arrow reflects its horizontal pixel offset from the image center ($(x_i - c)$). At the bottom, we show the egocentric RGB view from the agent’s current state.

We compare our method against the vanilla RoboHop baseline on four HM3D episodes via side-by-side video clips, emphasizing two key aspects:

1. **Success vs. Failure:** `nav_bed_20.mp4` and `nav_chair_55.mp4` demonstrate that, with our segment-matching module, the agent successfully reaches the goal, whereas with the vanilla RoboHop method it fails to reach the goal.
2. **Fewer Steps:** `nav_sofa_9.mp4` and `nav_chair_8.mp4` show that our model not only completes the task reliably but does so in fewer steps (around 40 steps in the former episode), while the baseline occasionally hesitates or deviates for a short interval before recovering.

6 Checklist Justifications

Limitations of Proposed Method

- **Reliance on 3D Instance-Segmentation Annotations:** Training is straightforward when 3D instance-level segmentation ground truth is available, as in datasets like ScanNet++ and Replica. For datasets without such annotations such as MapFree, our method can be trained using pseudo-ground truth generated via segment tracking approaches such as SAM2 [18]. However, tracking errors and inconsistencies in pseudo-labels may propagate through training and impact final matching performance. Investigating robust loss functions, confidence weighting, or semi-supervised training schemes to better handle imperfect annotations remains an important direction for broader applicability.

Code of Ethics

- **Research involving human subjects or participants:** We work on publicly available datasets that have been cited. No human subjects or their participation is involved.
- **Data-related concerns:** We utilized commonly used datasets without any human beings in them. So privacy is not a concern. The datasets are publicly available and have been cited. We use them in accordance with their respective licenses.
- **Societal Concerns and Broader Impact:** While the work poses no immediate societal risks, its automation potential could influence future labor markets. Conversely, it promises clear benefits for a range of indoor robot-navigation tasks.
- **Impact Mitigation Measures:** We will make the code, trained models and research artifacts publicly available to allow for reproducibility and mitigation in case unforeseen negative consequences arise from this research.

Compute Resources Our approach fine-tunes an existing foundation model, keeping computational demands and therefore its carbon footprint low. Our largest proposed model requires only a **single RTX A6000 GPU and 22 hours of training**. Using a CO_2 -to-power ratio of $0.7 \text{ kg CO}_2\text{e} / \text{kWh}$ factor, one training run emits $\approx 4.8 \text{ kg CO}_2\text{e}$.

References

- [1] Gabriele Berton, Gabriele Goletto, Gabriele Trivigno, Alex Stoken, Barbara Caputo, and Carlo Masone. Earthmatch: Iterative coregistration for fine-grained localization of astronaut photography. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2024.
- [2] Mohamed El Amine Boudjoghra, Angela Dai, Jean Lahoud, Hisham Cholakkal, Rao Muhammad Anwer, Salman Khan, and Fahad Shahbaz Khan. Open-YOLO 3d: Towards fast and accu-

- rate open-vocabulary 3d instance segmentation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL <https://openreview.net/forum?id=CRmiX0v16e>.
- [3] Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2024.
 - [4] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPR Deep Learning for Visual SLAM Workshop*, 2018. URL <http://arxiv.org/abs/1712.07629>.
 - [5] Johan Edstedt, Ioannis Athanasiadis, Mårten Wadenbäck, and Michael Felsberg. Dkm: Dense kernelized feature matching for geometry estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17765–17775, 2023.
 - [6] Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. RoMa: Robust Dense Feature Matching. *IEEE Conference on Computer Vision and Pattern Recognition*, 2024.
 - [7] Kartik Garg, Sai Shubodh Puligilla, Shishir Kolathaya, Madhava Krishna, and Sourav Garg. Revisit anything: Visual place recognition via image segment retrieval. In *European Conference on Computer Vision (ECCV)*, 2024.
 - [8] Sourav Garg, Krishan Rana, Mehdi Hosseinzadeh, Lachlan Mares, Niko Suenderhauf, Feras Dayoub, and Ian Reid. Robohop: Segment-based topological map representation for open-world visual navigation. *arXiv*, 2023.
 - [9] Qiao Gu, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, et al. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5021–5028. IEEE, 2024.
 - [10] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.
 - [11] Jacob Krantz, Stefan Lee, Jitendra Malik, Dhruv Batra, and Devendra Singh Chaplot. Instance-specific image goal navigation: Training embodied agents to find object instances. *arXiv preprint arXiv:2211.15876*, 2022.
 - [12] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European Conference on Computer Vision*, pages 71–91. Springer, 2024.
 - [13] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. LightGlue: Local Feature Matching at Light Speed. In *ICCV*, 2023.
 - [14] 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.
 - [15] Stefan Podgorski, Sourav Garg, Mehdi Hosseinzadeh, Lachlan Mares, Feras Dayoub, and Ian Reid. Tango: Traversability-aware navigation with local metric control for topological goals. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025.
 - [16] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.
 - [17] Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, Manolis Savva, Yili Zhao, and Dhruv Batra. Habitat-matterport 3d dataset

- (HM3d): 1000 large-scale 3d environments for embodied AI. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021. URL <https://arxiv.org/abs/2109.08238>.
- [18] 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.
 - [19] Xuelun Shen, Zhipeng Cai, Wei Yin, Matthias Müller, Zijun Li, Kaixuan Wang, Xiaozhi Chen, and Cheng Wang. Gim: Learning generalizable image matcher from internet videos. In *The Twelfth International Conference on Learning Representations*, 2024.
 - [20] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke M. Strasdat, Renzo De Nardi, Michael Goesele, Steven Lovegrove, and Richard Newcombe. The Replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019.
 - [21] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8922–8931, 2021.
 - [22] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2023.
 - [23] Chaoning Zhang, Dongshen Han, Yu Qiao, Jung Uk Kim, Sung-Ho Bae, Seungkyu Lee, and Choong Seon Hong. Faster segment anything: Towards lightweight sam for mobile applications. *arXiv preprint arXiv:2306.14289*, 2023.
 - [24] Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew J. Davison. In-place scene labelling and understanding with implicit scene representation. In *ICCV*, 2021.
 - [25] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R. Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.