Online Segment Any 3D Thing as Instance Tracking

Hanshi Wang[1,2,3,5]*, **Zijian Cai**[3], **Jin Gao**[1,2,5]†, **Yiwei Zhang**[1,2,5], **Weiming Hu**[1,2,5,6], **Ke Wang**[7], **Zhipeng Zhang**[3,4]†

[1]State Key Laboratory of Multimodal Artificial Intelligence Systems (MAIS), CASIA
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
[3]AutoLab, School of Artificial Intelligence, Shanghai Jiao Tong University [4]Anyverse Intelligence
[5]Beijing Key Laboratory of Super Intelligent Security of Multi-Modal Information
[6]School of Information Science and Technology, ShanghaiTech University [7]KargoBot
{hanshi.wang.cv, zhipeng.zhang.cv}@outlook.com; jin.gao@nlpr.ia.ac.cn

Abstract

Online, real-time, and fine-grained 3D segmentation constitutes a fundamental capability for embodied intelligent agents to perceive and comprehend their operational environments. Recent advancements employ predefined object queries to aggregate semantic information from Vision Foundation Models (VFMs) outputs that are lifted into 3D point clouds, facilitating spatial information propagation through inter-query interactions. Nevertheless, perception, whether human or robotic, is an inherently dynamic process, rendering temporal understanding a critical yet overlooked dimension within these prevailing query-based pipelines. This deficiency in temporal reasoning can exacerbate issues such as the over-segmentation commonly produced by VFMs, necessitating more handcrafted post-processing. Therefore, to further unlock the temporal environmental perception capabilities of embodied agents, our work reconceptualizes online 3D segmentation as an instance tracking problem (AutoSeg3D). Our core strategy involves utilizing object queries for temporal information propagation, where long-term instance association promotes the coherence of features and object identities, while short-term instance update enriches instant observations. Given that viewpoint variations in embodied robotics often lead to partial object visibility across frames, this mechanism aids the model in developing a holistic object understanding beyond incomplete instantaneous views. Furthermore, we introduce spatial consistency learning to mitigate the fragmentation problem inherent in VFMs, yielding more comprehensive instance information for enhancing the efficacy of both long-term and short-term temporal learning. The temporal information exchange and consistency learning facilitated by these sparse object queries not only enhance spatial comprehension but also circumvent the computational burden associated with dense temporal point cloud interactions. Our method establishes a new state-of-the-art, surpassing ESAM by 2.8 AP on ScanNet200 and delivering consistent gains on ScanNet, SceneNN, and 3RScan datasets, corroborating that identity-aware temporal reasoning is a crucial, previously underemphasized component for robust 3D segmentation in real-time embodied intelligence. Code is at `https://github.com/AutoLab-SAI-SJTU/AutoSeg3D`.

1 Introduction

The ability to perform online, real-time, and fine-grained 3D instance segmentation is a cornerstone for embodied intelligent agents to perceive and comprehend their operational environments. Autonomous robots and embodied assistants increasingly depend on such systems for exploring and interacting

*This work was completed during Hanshi's remote internship at SJTU and co-mentored by Prof. Zhipeng Zhang. †Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

with complex scenes. Early approaches predominantly adopts an offline paradigm, which involved accumulating complete point clouds prior to processing, thereby incurring prohibitive latency and memory costs. In pursuit of faster and online perception capabilities, recent research has begun to explore paradigms assisted by Vision Foundation Models (VFMs) such as SAM [1].

Current online VFM-assisted models are engineered to process streaming inputs by initially predicting segmentation results with VFMs and subsequently lifting the generated masks and recorded depth to superpoint representations. However, these pipelines simply concatenate global point features across scans and omit instance level temporal modeling, which worsens fragmentation and over segmentation by VFMs. Post hoc non-maximum suppression only partially corrects these errors and introduces the concurrent loss of valid information not as expected.

Seeking to address these limitations, we draw inspiration from established methodologies for maintaining temporal coherence in online perception. Classical multi-object tracking (MOT) methods, for instance, achieve consistent identity assignment by exploiting spatial continuity and appearance affinities to link detections across frames [2, 3]. Similarly, video instance segmentation frameworks like VisTR [4] and 3D detection models such as Sparse4D [5] employ query-based memory banks to propagate and update object features over time, enabling each instance to maintain a persistent representation robust to occlusion and partial views. The core design principle underpinning these diverse approaches is the explicit maintenance and evolution of instance-specific representations across temporal sequences. Inspired by this paradigm, we recast online 3D instance segmentation as an instance-tracking task. By integrating object-level temporal priors directly into the segmentation pipeline, our approach aims to concurrently rectify over-segmentation errors and enforce identity consistency, thereby substantially enhancing overall segmentation performance and robustness.

More specifically, we introduce a novel, tracking-centric pipeline that directly addresses the two core limitations of VFM-based methods. Our framework decomposes into three lightweight and synergistic modules. First, the Long-Term Memory (LTM) maintains a bounded track bank and employs Hungarian assignment based on confidence-gated affinity matrix to recover identities after prolonged occlusions with constant overhead. Second, the Short-Term Memory (STM) refines instance embeddings via distance-aware cross-frame attention to inject immediate temporal context while filtering out background noise. Third, Spatial Consistency Learning (SCL) merges high-affinity mask fragments at inference by jointly reasoning over 2D appearance and 3D geometry, while concurrently employing one-to-many fragment supervision during training to mitigate over-segmentation and generate coherent, high-fidelity queries for LTM and STM. Together, these components form a cohesive, real-time 3D instance segmentation system that enforces consistent object identities across frames, injects immediate temporal context while filtering out background noise, and merges high-affinity fragments to directly counteract VFM over-segmentation. By integrating these modules, our framework preserves real-time throughput while delivering a 2.8 AP gain over recent ESAM [6] on ScanNet200 [7]. Extensive evaluations on both ScanNet200 and ScanNet [8], as well as zero-shot assessments on SceneNN [9] and 3RScan [10] demonstrate consistent performance gains.

In summary, our contributions are as follows: **1)** We recast online 3D instance segmentation as a continuous instance tracking problem by treating each VFM-derived mask as a track query within a unified framework. **2)** We propose a lightweight architecture with three synergistic modules where LTM propagates identities across frames to ensure continuity, STM injects short-term temporal context while filtering background noise, and SCL merges overlapping fragments to counteract over-segmentation and enrich instance embeddings. **3)** Our framework achieves new state-of-the-art results on ScanNet200, ScanNet, SceneNN, and 3RScan while sustaining real-time throughput, and ablation studies verify the contribution of each component.

2 Related Work

VFM-assisted 3D Scene Segmentation. Vision foundation models (VFMs) have emerged as a promising cornerstone for 3D scene understanding in embodied intelligence, especially in the construction and reasoning of 3D spatial information [11, 12, 13, 14, 1, 15, 16, 17, 18]. Large-scale pretrained VFMs such as SAM [1] and CLIP [12] exhibit powerful open-vocabulary segmentation and semantic alignment capabilities, which have been extensively leveraged in downstream 3D perception pipelines. SAM3D [19] first predicts 2D instance masks with SAM and then lifts them to 3D via depth and camera parameters, followed by geometric merging. CLIP2Scene [20] distills multimodal knowl-

edge from CLIP into a 3D backbone through semantic and spatio-temporal consistency regularization, enabling label-efficient scene parsing. OpenMask3D [21] combines CLIP-extracted visual features with SAM-refined masks to generate discriminative per-instance embeddings for open-vocabulary 3D instance segmentation. Despite these advances, several studies have highlighted that the 2D masks produced by VFMs are often over-segmented. SAI3D [22], for example, decomposes the reconstructed mesh into 3D primitives, assigns semantic scores to 2D masks via Semantic-SAM [15], and aggregates the primitives through a graph-based region-growing algorithm. Nevertheless, existing approaches still rely on heuristic post-hoc fusion of projected 3D masks, which often proves brittle in cluttered or dynamically changing robotic environments. In this work, we propose a learnable fusion module that jointly reasons over over-segmentation hypotheses in both 2D and 3D spaces. By optimizing fusion in an end-to-end manner, our method mitigates the impact of erroneous 2D masks and delivers more robust and scalable 3D scene understanding.

Online 3D Scene Perception. Driven by the rapid advancements in autonomous driving and embodied AI, robotic tasks are increasingly demanding higher levels of 3D scene understanding. In these scenarios, the ability to process information in real time, adapt to diverse conditions, and achieve perception is crucial. However, most of the common instance segmentation methods [23, 24, 25, 26, 27, 28, 29, 21, 30, 31, 32, 33] are offline. They can handle large-scale datasets but are highly dependent on the quality of preprocessing and data augmentation, which makes it difficult to apply them to complex and ever-changing robotic environments. Recently, online 3D scene perceptions [34, 35, 36, 37, 38, 39] have attracted increasing attention. INS-Conv [38] proposes an incremental sparse convolutional network for online 3D segmentation, which achieves efficient and accurate inference by processing only the residuals between consecutive frames and incorporating an uncertainty term to adaptively select which residuals to update. MemAda [40] proposes an adapter-based model that equips mainstream offline frameworks with the competence to perform online scene perception, enabling them to process real-time RGB-D sequences efficiently. Building on this foundation, ESAM [6] further advances the field by achieving online scene segmentation and designing a dual-layer decoder along with auxiliary tasks to facilitate the merging of 3D masks. While prevailing methods fuse dense features (*e.g.,* raw point clouds) temporally, they often lack the semantic context crucial for instance-level tasks. We address this by recasting online segmentation as instance tracking, which allows us to propagate semantically rich instance information across frames. This focus on semantic consistency through time yields significantly more precise instance segmentation results, while also being computationally efficient.

3 Method

3.1 Overall Architecture

Fig. 3.1 illustrates our tracking-centric online 3D segmentation framework. The design draws inspiration from the brain's complementary learning systems [41, 42, 43, 44, 45]. Specifically, the hippocampus rapidly forms episodic memories, allowing quick adaptation to novel contexts and interaction with recent experience, whereas the neocortex consolidates these transient traces into durable representations through slow, cumulative learning, producing a stable store of knowledge. This dual mechanism not only enhances adaptability but also ensures the coherence and persistence of memory. Mirroring this division, we decompose our framework into long-term memory for instance association and short-term memory for instance update, realised by three lightweight yet synergistic modules: **1)** Long-term memory (LTM), detailed in Sec. 3.2, matches instance identities over extended periods, enabling recovery after prolonged occlusion. **2)** Short-term memory (STM), detailed in Sec. 3.3, recurrently updates each instance's representation with information from the immediately preceding frame. **3)** Spatial Consistency Learning (SCL) includes Learning-Based Mask Integration at inference and Instance-Consistency Mask Supervision during training, detailed in Sec. 3.4, respectively counteract VFM's intrinsic over-segmentation, thereby reducing query redundancy and furnishing STM and LTM with coherent, high-fidelity mask representations.

3.2 Long-Term Memory for Instance Association

Online 3D segmentation requires that all point-cloud observations of the same instance, collected across successive frames, be fused into a single temporally coherent instance. To improve the

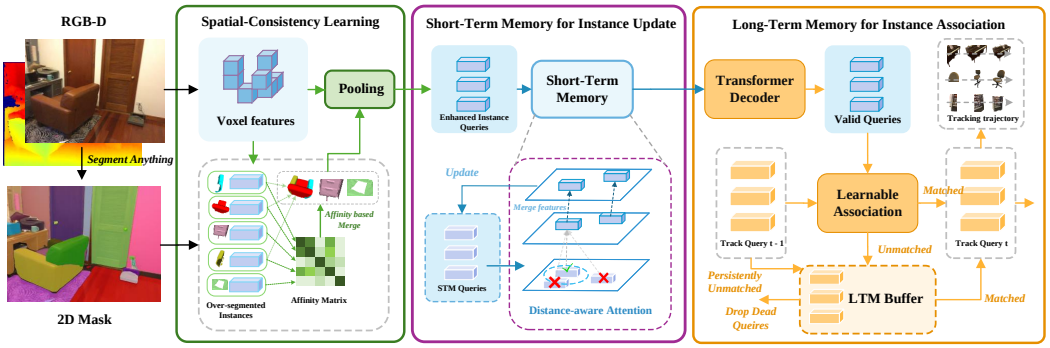

Figure 1: **This diagram delineates the operational mechanisms of our constituent modules.** Spatial Consistency Learning (SCL) mitigates the over-segmentation tendencies of VFM by employing a one-to-many supervision strategy during the training phase and utilizing learning-based mask integration at the inference stage. The Short-term Memory (STM) module enriches current instance representations by integrating observational data from prior frames. Furthermore, the Long-term Memory (LTM) module is engineered to associate instances, segmented by the Visual Front-end Module (VFM), with established tracklets in memory, consequently enhancing temporal consistency.

temporal consistancy, we recast instance aggregation as an explicit instance tracking problem with supervised matching, confidence gating, and Hungarian assignment.

Concretely, at the first frame ($t = 1$), we obtain N_1 instance queries that derived from 3D mask (Eq. 5) and their corresponding embeddings $\mathbf{Q}_1 \in \mathbb{R}^{\mathrm{N_1}\times \mathrm{d}}$ that derived by applying a MLP to instance queries, and predicted 3D bounding boxes $\mathbf{B}_1 \in \mathbb{R}^{\mathrm{N_1}\times 6}$. Each box is axis-aligned and specified by its minimum and maximum coordinates $(x_{\min}, y_{\min}, z_{\min}, x_{\max}, y_{\max}, z_{\max})$. Similarly, for every subsequent frame t containing N_t segments, we obtain the instance embeddings $\mathbf{Q}_t \in \mathbb{R}^{\mathrm{N_t}\times \mathrm{d}}$, the corresponding boxes $\mathbf{B}_t \in \mathbb{R}^{\mathrm{N_t}\times 6}$, and the instance embeddings from the tracklets in memory $\mathbf{Q}^{\mathrm{Trk}} \in \mathbb{R}^{\mathrm{N^{Trk}}\times \mathrm{d}}$. Here, N^{Trk} represents the number of active tracklets up to now. Critically, the embedding associated with each tracklet is not merely derived from the immediately preceding frame $t-1$, instead, it encapsulates richer temporal information accumulated across the sequence, thereby reflecting a more comprehensive long-term history (see Sec. 3.3 for more details). Then we measure the similarity between instances (segments) from current frames with tracklets by,

$$\mathbf{E}^{\mathrm{app}}_{\mathrm{ij}} = \mathbf{Q}_{\mathrm{t}}[i] \odot \mathbf{Q}^{\mathrm{Trk}}[j], \quad \mathbf{E}^{\mathrm{geo}}_{\mathrm{ij}} = \mathrm{MLP}\big(\mathrm{IoU}(\mathbf{B}_{\mathrm{t}}[i], \mathbf{B}^{\mathrm{Trk}}[j])\big), \quad \mathbf{E}_{\mathrm{ij}} = \mathbf{E}^{\mathrm{app}}_{\mathrm{ij}} + \mathbf{E}^{\mathrm{geo}}_{\mathrm{ij}}, \quad (1)$$

where $\mathbf{E}^{\mathrm{app}}, \mathbf{E}^{\mathrm{geo}}, \mathbf{E} \in \mathbb{R}^{N_{\mathrm{t}}\times N^{Trk}\times \mathrm{d}}$ respectively correspond to the appearance, geometric, and fused affinity features. We then project the fused affinity features using learned $\mathbf{w}, \mathbf{w}' \in \mathbb{R}^{\mathrm{d}}$,

$$M_{\mathrm{ij}} = \frac{\exp\big(\mathbf{w}^{\top}\mathbf{E}_{\mathrm{ij}}\big)}{\sum_{\mathrm{j}'}\exp\big(\mathbf{w}^{\top}\mathbf{E}_{\mathrm{ij}'}\big)}, \quad C_{\mathrm{ij}} = \sigma\big(\mathbf{w}'^{\top}\mathbf{E}_{\mathrm{ij}}\big), \quad A_{\mathrm{ij}} = M_{\mathrm{ij}}\, C_{\mathrm{ij}}. \quad (2)$$

where M_{ij} is a row-normalised affinity obtained via softmax, which represents the relative similarity that segmented instance i corresponds to tracklet j. Since each row sums to 1, every segment allocates its entire probability mass across the set of candidate tracklets. To modulate this raw affinity we introduce a sigmoid-based confidence gate $C_{\mathrm{ij}} = \sigma(\cdot)$, which down-weights uncertain matches and suppresses spurious associations. To convert these probabilities into one-to-one correspondences, we formulate a bipartite matching problem. Specifically, segments and tracklets form a bipartite graph with edges weighted by gated affinities A_{ij}. Solving this assignment with the Hungarian algorithm selects the set of pairs (i, j) that maximises the summed weights while ensuring that every segment and every track is used at most once. For each matched pair, the track state is updated by,

$$\mathbf{B}^{\mathrm{Trk}}[j] \leftarrow \frac{\alpha_{\mathrm{j}}\,\mathbf{B}^{\mathrm{Trk}}[j] + \mathbf{B}_{\mathrm{t}}[i]}{\alpha_{\mathrm{j}} + 1}, \quad \mathbf{Q}^{\mathrm{Trk}}[j] \leftarrow \frac{\alpha_{\mathrm{j}}\,\mathbf{Q}^{\mathrm{Trk}}[j] + \mathbf{Q}_{\mathrm{t}}[i]}{\alpha_{\mathrm{j}} + 1}, \quad \alpha_{\mathrm{j}} \leftarrow \alpha_{\mathrm{j}} + 1, \quad (3)$$

where α_{j} is the track age. Unmatched instances initialise new tracklets with $\alpha = 1$. Tracks that remain unmatched for more than T_{life} frames are marked stale, removed from the active set, and pushed into a fixed-capacity queue, the oldest entry is evicted when the buffer overflows. At every time step t, any segment that remains unassigned after the active-track matching stage is subsequently

matched against the LTM buffer with the identical Hungarian solver. A successful match reactivates the stored tracklet, restoring its state and resetting its age, thereby recovering instances that reappear after extended occlusion. By coupling confidence-gated Hungarian assignment with a bounded long-term memory, the proposed strategy suppresses spurious instances, maintains identities through prolonged occlusions, and guarantees constant computational overhead.

3.3 Short-Term Memory for Instance Update

In online 3D segmentation, the inherent scene continuity, where instances visible in frame $t-1$ often reappear in frame t, naturally motivates the use of cross-frame attention mechanisms to integrate historical context. For embodied agents, which frequently experience rapid and significant viewpoint variations, the effective fusion of object-level appearance information gathered from these diverse perspectives is particularly crucial. This capability to integrate multi-view object-centric features allows the agent to build a more robust and consistent understanding of instances over time, despite substantial changes in their observed appearance. Therefore, we design a instance update module that reuses and continually refines instance-centric embeddings.

We recognize that applying global cross-attention between all N_t current queries and the instance embeddings from the previous frame can introduce substantial noise. This occurs because background queries often form irrelevant associations with prior instance features, thereby degrading the fusion process. To instill explicit instance awareness and mitigate this, we introduce the distance-aware Short-Term Memory (STM). Specifically, to filter out irrelevant interactions we adopt the distance-aware attention, which gates attention by Euclidean distance between instance centroids,

$$\mathrm{Attn}(\mathbf{Q}'_{\mathrm{t}}, \mathbf{K}_{\mathrm{t}-1}, \mathbf{V}_{\mathrm{t}-1}) = \mathrm{Softmax}\Big(\frac{\mathbf{Q}'_{\mathrm{t}}\,\mathbf{K}_{\mathrm{t}-1}^{\top}}{\sqrt{d}} - \mathrm{diag}(\boldsymbol{\tau}_{\mathrm{t}})\,\mathbf{D}^{(\mathrm{t}-1,\mathrm{t})}\Big)\,\mathbf{V}_{\mathrm{t}-1}, \tag{4}$$

where $\mathbf{Q}'_{\mathrm{t}} \in \mathbb{R}^{\mathrm{N_t}\times \mathrm{d}}$ denotes current instance queries, the memory key $\mathbf{K}_{\mathrm{t}-1} \in \mathbb{R}^{\mathrm{N_{t-1}}\times \mathrm{d}}$ and value $\mathbf{V}_{\mathrm{t}-1} \in \mathbb{R}^{\mathrm{N_{t-1}}\times \mathrm{d}}$ are derived from $\mathbf{Q}_{\mathrm{t}-1}$, $\mathbf{D}^{(\mathrm{t}-1,\mathrm{t})} \in \mathbb{R}^{\mathrm{N_t}\times \mathrm{N_{t-1}}}$ stores pairwise centroid distances and $\boldsymbol{\tau}_{\mathrm{t}} = [\tau_1, \ldots, \tau_{\mathrm{N_t}}]^{\top}$ contains query-specific receptive-field scales. We predict these scales with a shared linear layer, $\boldsymbol{\tau}_{\mathrm{t}} = \mathrm{Linear}(\mathbf{Q}'_{\mathrm{t}})$, so each query adaptively narrows or widens its spatial scope. Large τ_{i} suppress attention to distant memory slots, encouraging local refinement, whereas small τ_{i} retain a global context when necessary. By suppressing attention to remote regions and modulating each query's receptive field, short-term memory yields temporally enhanced embeddings $\mathbf{Q}_{\mathrm{t}}$.

3.4 Spatial Consistency Learning for Robust Association

As illustrated in Fig 2, VFMs like SAM [1] frequently fragments a single instance into several neighbouring masks. This fragmentation compromises effective cross-frame instance association. Previous methods [6] ignore this, resulting in degraded spatial coherence. To mitigate this gap, we introduce learning-based mask integration (LMI) at inference to merge high-affinity fragments and instance consistency mask supervision (ICMS) during training to apply one-to-many supervision.

Learning-Based Mask Integration. To recover coherent masks, we learn an affinity matrix that merges masks belonging to the same instance at every frame t. Given point cloud features $\mathbf{P}_{\mathrm{t}}$ and corresponding 2D masks set $\mathcal{M}_{\mathrm{t}}$, we can get query features $\mathbf{Q}_{\mathrm{t}}$ and position $\mathbf{X}_{\mathrm{t}}$ through

$$(\mathbf{Q}_{\mathrm{t}}, \mathbf{X}_{\mathrm{t}}) = \mathrm{Pool}(\mathbf{P}_{\mathrm{t}}, \mathcal{M}_{\mathrm{t}}), \quad \mathcal{M}_{\mathrm{t}} = \{\,\mathbf{m}_{\mathrm{i}}\}_{\mathrm{i}=1}^{\mathrm{N_t}}, \tag{5}$$

where Pool aggregates point features within each mask. We then predicts axis-aligned bounding boxes $\mathbf{B}_{\mathrm{t}} = \mathrm{MLP}(\mathbf{Q}_{\mathrm{t}}) \in \mathbb{R}^{N_{\mathrm{t}}\times 6}$, since the boxes generated by corresponding 3D mask may not be a complete object [6]. For each pair (i, j) we compute the affinity feature $\mathbf{E}_{\mathrm{ij}}$ and A_{ij} as in Sec. 3.2. We first perform hierarchical clustering to identify mask groups whose pairwise affinities A_{ij} all exceed δ. We then merge the masks within each such group to form $\tilde{\mathcal{M}}_{\mathrm{t}}$, and finally re-pool features over these merged masks,

$$(\mathbf{Q}'_{\mathrm{t}}, \mathbf{X}'_{\mathrm{t}}) = \mathrm{Pool}\big(\mathbf{P}_{\mathrm{t}}, \tilde{\mathcal{M}}_{\mathrm{t}}\big). \tag{6}$$

The mask-aggregation module is invoked only at inference. During training we intentionally refrain from merging masks to leverage fragment diversity as implicit data augmentation, detailed below.

Instance Consistency Mask Supervision. In addition to retaining fragmented masks as implicit data augmentation, each fragment can also provides a complementary view of the same object. Therefore,

we supervise corresponding fragments (many) with each ground-truth instance (one). This strategy improves robustness to low-quality masks, yields more consistent predictions for fragmented queries, and simplifies duplicate removal, which is vital for instance association in Sec. 3.2.

To formalise this supervision, consider a ground-truth instance $\mathbf{g}_{\mathrm{k}}$ and its corresponding query set

$$\mathcal{Q}_{\mathrm{k}} = \big\{\, \mathbf{q}_{\mathrm{i}} \;\big|\; \tfrac{|\mathcal{P}(\mathbf{m}_{\mathrm{i}}) \cap \mathcal{P}(\mathbf{g}_{\mathrm{k}})|}{|\mathcal{P}(\mathbf{m}_{\mathrm{i}})|} > 0.5 \big\}, \tag{7}$$

where $\mathcal{P}(\cdot)$ denotes the pixel-set of a mask $\mathbf{m}_{\mathrm{i}}$ and its corresponding query $\mathbf{q}_{\mathrm{i}}$. We then enforce consistency across these fragments via

$$L_{1:\mathrm{N}} \;=\; \sum_{k=1}^{\mathrm{N_{gt}}} \sum_{\mathbf{q}_{\mathrm{i}} \in \mathcal{Q}_{\mathrm{k}}} \ell\big(f(\mathbf{q}_{\mathrm{i}}), \mathbf{y}_{\mathrm{k}}\big). \tag{8}$$

where ℓ is the loss and $\mathbf{y}_{\mathrm{k}}$ denotes the ground-truth. However, as shown in Tab. 8, naively replacing the original one-to-one loss with $L_{1:\mathrm{N}}$ leads to a consistent drop in segmentation accuracy. Our experiments demonstrate that this modification erodes the model's capacity to select the highest-quality fragment, which is supported by the self-attention mechanism. To satisfy both objectives, we configure the decoder in two distinct branches. In the first branch we enable self-attention and employ standard one-to-one supervision in order to preserve fragment selection capability. In the second branch we disable self-attention and apply one-to-many supervision in order to strengthen robustness across diverse fragments. Notably, this dual-branch mechanism is active only during training and incurs no additional computational cost at inference time.

3.5 Loss Functions

Our framework is trained end-to-end by minimising:

$$\mathcal{L} = \mathcal{L}_{\mathrm{seg}} + \beta_{\mathrm{ltm}}\, \mathcal{L}_{\mathrm{ltm}} + \beta_{\mathrm{agg}}\, \mathcal{L}_{\mathrm{agg}}, \tag{9}$$

where the scalars β_* weight the contribution of each term.

Segmentation loss $\mathcal{L}_{\mathrm{seg}}$. The dual-decoder architecture described in Sec. 3.4 yields three sub-losses:

$$\mathcal{L}_{\mathrm{seg}} = \mathcal{L}_{1:1} + \lambda\, \mathcal{L}_{1:\mathrm{N}} + \gamma\, \mathcal{L}_{\mathrm{bg}}, \tag{10}$$

where $\mathcal{L}_{1:1}$ enforces one-to-one assignment, $\mathcal{L}_{1:\mathrm{N}}$ ensures consistency across masks through multi-target supervision , and L_{bg} penalizes background masks.

Long-term memory loss $\mathcal{L}_{\mathrm{ltm}}$. We introduce a matrix y_{ij}, where $y_{\mathrm{ij}} = 1$ if query q_{i} and track t_{j} refer to the same ground-truth instance, and $y_{\mathrm{ij}} = 0$ otherwise. We compute a one-to-one assignment π^* by applying the Hungarian algorithm to the cost matrix $-\log \widehat{M}_{\mathrm{ij}}$. The matching loss becomes

$$\mathcal{L}_{\mathrm{match}} = -\frac{1}{N_{\mathrm{t}}} \sum_{(\mathrm{i,j}) \in \pi^*} \log \widehat{M}_{\mathrm{ij}}. \tag{11}$$

To generate the sigmoid gate C_{ij} for confidence we add

$$\mathcal{L}_{\mathrm{conf}} = -\frac{1}{N_{\mathrm{t}} N_{\mathrm{t}-1}} \sum_{\mathrm{i,j}} \big[y_{\mathrm{ij}} \log C_{\mathrm{ij}} + (1 - y_{\mathrm{ij}}) \log(1 - C_{\mathrm{ij}}) \big]. \tag{12}$$

The full long-term memory loss is then

$$\mathcal{L}_{\mathrm{ltm}} = \mathcal{L}_{\mathrm{match}} + \beta_{\mathrm{conf}}\, \mathcal{L}_{\mathrm{conf}}, \tag{13}$$

Mask-aggregation loss $\mathcal{L}_{\mathrm{agg}}$. To supervise the affinity predictor in LMI, we employ binary cross-entropy over positive pairs $\mathcal{P}$ and negative pairs $\mathcal{N}$:

$$\mathcal{L}_{\mathrm{agg}} = -\frac{1}{|\mathcal{P}|} \sum_{(\mathrm{i,j}) \in \mathcal{P}} \log A_{\mathrm{ij}} - \frac{1}{|\mathcal{N}|} \sum_{(\mathrm{i,j}) \in \mathcal{N}} \log\big(1 - A_{\mathrm{ij}}\big). \tag{14}$$

Pairs whose masks overlap a ground-truth instance by over 50 % are positive, all others are negative.

Table 1: Class-agnostic 3D instance segmentation results of different methods on ScanNet200 dataset.

Method	Present at	Type	VFM	AP	AP_{50}	AP_{25}	FPS
SAMPro3D [46]	3DV'2025	Offline	SAM	18.0	32.8	56.1	–
Open3DIS [47]	CVPR'2024	Offline	GroundedSAM	34.6	43.1	48.5	–
SAI3D [22]	CVPR'2024	Offline	SemanticSAM	28.2	47.2	67.9	–
SAM3D [19]	ICCVW'2023	Online	SAM	20.2	35.7	55.5	0.4
ESAM [6]	ICLR'2025	Online	SAM	42.2	63.7	79.6	0.7
AutoSeg3D (Ours)	-	Online	SAM	**45.5**	**66.7**	**81.0**	0.7
ESAM-E	ICLR'2025	Online	FastSAM	43.4	65.4	80.9	10.6
AutoSeg3D (Ours)	-	Online	FastSAM	**46.2**	**67.9**	**81.7**	10.1

Table 2: 3D instance segmentation results of different methods on ScanNet and SceneNN datasets. * denotes represent the results we reproduced following the official released config.

Method	Present at	Type	ScanNet			SceneNN		
			AP	AP_{50}	AP_{25}	AP	AP_{50}	AP_{25}
TD3D [48]	ICME'2024	Offline	46.2	71.1	81.3	–	–	–
Oneformer3D [49]	CVPR'2024	Offline	59.3	78.8	86.7	–	–	–
INS-Conv [38]	CVPR'2022	Online	–	57.4	–	–	–	–
TD3D-MA [48]	ICME'2024	Online	39.0	60.5	71.3	26.0	42.8	59.2
ESAM [6]*	ICLR'2025	Online	41.6	59.6	75.2	30.3	47.6	63.4
AutoSeg3D (Ours)	-	Online	**43.4**	**62.5**	**77.4**	**33.1**	**52.6**	**63.8**

4 Experiments

4.1 Experiment Settings

Following our baseline ESAM [6], we begin by training a single-view perception model on ScanNet(200)-25k, a subset of ScanNet200 [7] with RGB-D frames. Then we fine-tune it on RGB-D sequences with full loss functions and randomly sample 8 RGB-D frames per scene at each training step. For the optimization settings, we use an AdamW optimizer with a learning rate of 0.0001 and a weight decay of 0.05 and the batch size is set to 4. All experiments are conducted using PyTorch on a single NVIDIA Tesla A100 GPU. Our experiments are conducted on ScanNet [8], ScanNet200 [7], SceneNN [9], and 3RScan [10] datasets.

4.2 Comparison with State-of-the-arts

Results on ScanNet200 of Class-agnostic Setting. Tab. 1 details the class-agnostic results on ScanNet200, demonstrating the superiority of our approach over existing state-of-the-art methods. Specifically, when SAM serves as the 2D segmentation model, our method achieves gains of 3.3 in AP, 3.0 in AP_{50}, and 1.4 in AP_{25} compared to the recent ESAM [6]. The consistent performance improvements, even with a more lightweight 2D segmentation model such as FastSAM [50], underscore the effectiveness and generalizability of our method.

Results on ScanNet and SceneNN. Following the experimental setup of ESAM [6], Tab. 2 reports the results of our method, which is trained on ScanNet and subsequently evaluated on both ScanNet and SceneNN to assess its generalization performance. The notable improvements across multiple evaluation metrics and datasets strongly demonstrate the effectiveness and generalizability of our approach. Specifically, our method achieves significant gains of 1.8 in AP, 2.9 in AP_{50}, and 2.2 in AP_{25} on ScanNet evaluation compared to ESAM.

Results on SceneNN and 3RScan. Tab. 3 reports the results of our method, trained on ScanNet200 and evaluated on SceneNN and 3RScan, which again demonstrate its strong generalization capabilities. Our approach surpasses previous methods, achieving significantly higher AP_{50} and AP_{25} scores. This underscores the effectiveness and adaptability of our method for robotic applications.

Table 3: Results of transferring different methods trained on ScanNet200 to SceneNN and 3RScan. "-E" indicates using FastSAM instead of SAM for 2D segmentation.

Method	Present at	Type	ScanNet200→SceneNN			ScanNet200→3RScan		
			AP	AP_{50}	AP_{25}	AP	AP_{50}	AP_{25}
SAMPro3D [46]	3DV'2025	Offline	12.6	25.8	53.2	3.9	8.0	21.0
Open3DIS [47]	CVPR'2024	Offline	18.2	32.2	48.9	9.5	21.8	47.0
SAI3D [22]	CVPR'2024	Offline	18.6	34.7	65.7	8.1	16.9	37.0
SAM3D [19]	ICCVW'2023	Online	15.1	30.0	51.8	6.2	13.0	33.9
ESAM [6]	ICLR'2025	Online	28.8	52.2	69.3	14.1	31.2	59.6
AutoSeg3D (Ours)	-	Online	**29.7**	**53.6**	**71.9**	**16.0**	**32.4**	**60.7**
ESAM-E [6]	ICLR'2025	Online	28.6	50.4	71.0	13.9	29.4	58.8
AutoSeg3D (Ours)	-	Online	**30.2**	**54.1**	**72.8**	**16.8**	**34.3**	**61.0**

4.3 Ablation Studies and Further Analysis

Table 4: Component-wise ablation.

	LTM	STM	LMI	ICMS	AP	AP_{50}	AP_{25}
①	–	–	–	–	41.6	62.9	78.7
②	✓	–	–	–	44.1	65.8	80.7
③	–	✓	–	–	42.9	63.8	80.0
④	✓	✓	–	–	44.8	66.7	81.0
⑤	✓	✓	–	✓	45.6	66.9	81.2
⑥	✓	✓	✓	–	45.5	67.0	81.3
⑦	✓	✓	✓	✓	46.2	67.9	81.7

Component-wise Ablation. To further investigate the effects of our designs, we conduct an ablation studies on the ScanNet200. As depicted in Tab. 4, the introduction of long-term memory obtains gains of 2.5 and 2.9 in AP and AP_{50} (② $vs.$ ①), because of its effectiveness in instance association. The integrating of short-term memory enhances our model's ability to capture positional and content details from previous frames, resulting in performance improvements of 1.3 and 0.9 in AP and AP_{50} respectively (③ $vs.$ ①). ⑤ and ⑥ proves the effectiveness of components in the proved spatial consistency learning. The synergistic combination of all proposed elements constitutes an effective tracking-centric 3D segmentation framework, as in ⑦.

Table 5: Ablation for LTM.

	Strategy	AP	AP_{50}	AP_{25}
①	w/o LTM	43.2	64.8	80.4
②	+ Geometric	43.7	65.4	80.5
③	+ Appearance	45.0	66.5	81.3
④	+ Confidence	45.5	67.2	81.5
⑤	+ Recall	46.2	67.9	81.7

Long-Term Memory. As shown in Tab. 5, compared to the baseline without LTM, adding geometric and appearance cues leads to steady gains (② $vs.$ ①). Incorporating confidence estimation brings a notable boost, while further combining the recall mechanism achieves the highest scores, with AP increasing from 43.2 to 46.2 (+3.0), AP_{50} from 64.8 to 67.9 (+3.1), and AP_{25} from 80.4 to 81.7 (+1.3) (⑤ $vs.$ ①). The recall and confidence strategies enable the model to effectively handle challenging cases such as long-term occlusions and ambiguous matches, resulting in more reliable temporal consistency throughout the sequence.

Table 6: Ablation for STM.

	Strategy	AP	AP_{50}	AP_{25}
①	w/o STM	44.7	66.4	81.3
②	+ cross	45.1	66.5	81.2
③	+ distance	45.8	67.5	81.6
④	+ scale	46.2	67.9	81.7

Short-Term Memory. As shown in Tab. 6, starting from the baseline without STM, solely introducing cross-frame attention brings limited improvement due to potential noise from irrelevant associations (② $vs.$ ①). By further incorporating our distance-aware attention, which gates memory updates based on instance centroid distances, we observe a clear performance boost (③ $vs.$ ①). Equipped with query-specific receptive-field scales, the final STM boosts AP from 44.7 to 46.2 (+1.5), AP_{50} from 66.4 to 67.9 (+1.5), and AP_{25} from 81.3 to 81.7 (+0.4) (④ $vs.$ ①). These results demonstrate that explicitly modeling spatial proximity and adaptive receptive fields effectively suppresses noisy associations and enhances instance update accuracy in dynamic scenes.

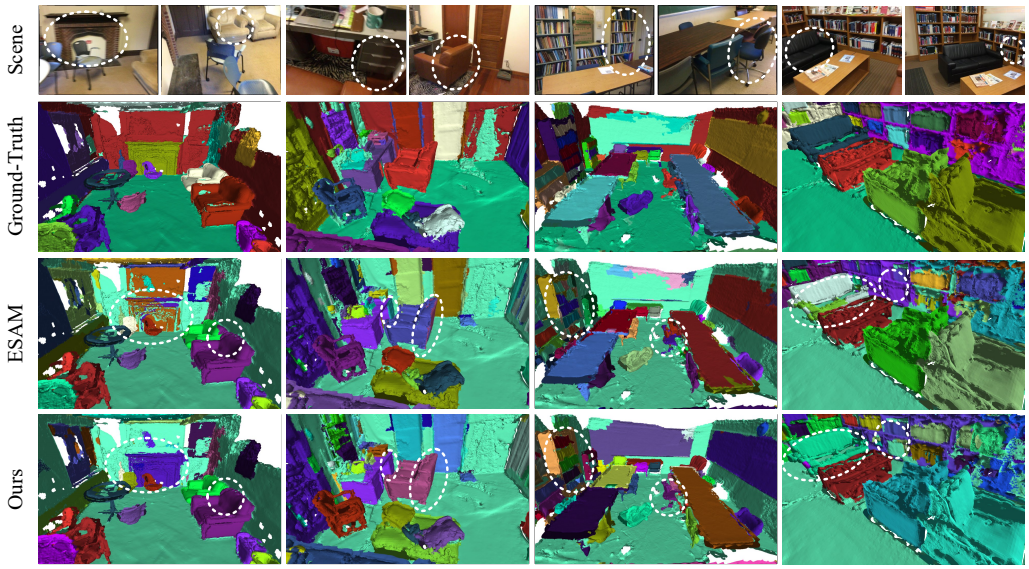

Figure 2: Visualization of segmentation results on ScanNet200 dataset.

Table 7: Ablation for LMI.

	Strategy	AP	AP_{50}	AP_{25}
①	w/o LMI	45.6	66.9	81.0
②	train and infer.	44.5	66.1	80.4
③	infer. only	46.2	67.9	81.7

Learning-Based Mask Integration. As shown in Tab. 7, applying Learning-Based Mask Integration (LMI) only during inference yields the best performance, improving AP from 45.6 to 46.2 (+0.6), AP_{50} from 66.9 to 67.9 (+1.0) (③ $vs.$ ①). By contrast, incorporating LMI during training degrades performance, as early-stage inaccuracies introduce false mask fusions that hinder model convergence (② $vs.$ ①).

Table 8: Ablation of ICMS.

	Strategy	TopK	AP	AP_{50}	AP_{25}
①	w/o ICMS	–	45.5	67.0	81.3
②	single-branch	4	44.2	65.8	80.6
③	dual-branch	2	46.2	67.4	81.4
④	dual-branch	4	46.2	67.9	81.7
⑤	dual-branch	6	46.1	67.3	81.3
⑥	dual-branch	8	46.1	67.2	81.4

Instance Consistency Mask Supervision. As shown in Tab. 8, introducing Instance Consistency Mask Supervision (IMCS) with dual branches and TopK=4 achieves the best performance, improving AP from 45.5 to 46.2 (+0.7) (④$vs.$①). Here, setting K=4 indicating chosing the four masks exhibiting the highest similarity scores when compared to the ground truth of one object. By contrast, the single-branch configuration incurs a substantial drop in all metrics, underscoring the importance of combining one-to-one and one-to-many supervision signals.

Qualitative Analysis. We present a qualitative analysis conducted on the ScanNet validation set, with illustrative examples provided in Fig. 2. These results further substantiate the superior instance segmentation capabilities of our proposed model. The visualizations demonstrate that our model not only accurately segments target objects but also effectively rectifies over-segmented masks.

5 Conclusion and Limitation

Conclusion. In this paper, we present a novel, tracking-centric framework for online, real-time, and fine-grained 3D instance segmentation. By recasting the task as continuous instance tracking, our approach integrates Long-Term Memory for robust identity propagation, Short-Term Memory for immediate temporal context, and Spatial Consistency Learning to suppress over-segmentation. Extensive experiments on multiple benchmarks demonstrate that our lightweight system achieves state-of-the-art accuracy while maintaining real-time efficiency. **Limitation.** Both our and previous methods do not explicitly model relative motion of moving objects. We leave this for future research.

Acknowledgements

This work was supported in part by the Natural Science Foundation of China (Grant No. 62036011, 62422317, U22B2056, 62192782, 62503323), the Beijing Natural Science Foundation (Grant No. JQ22014, L223003). The work of Weiming Hu was also supported in part by the Natural Science Foundation of China (Grant No. U2441241, U24A20331, 62202470, 62473363).

References

[1] 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.

[2] Fangao Zeng, Bin Dong, Yuang Zhang, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. In *European conference on computer vision*, pages 659–675. Springer, 2022.

[3] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixe, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8844–8854, 2022.

[4] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8741–8750, 2021.

[5] Xuewu Lin, Tianwei Lin, Zixiang Pei, Lichao Huang, and Zhizhong Su. Sparse4d: Multi-view 3d object detection with sparse spatial-temporal fusion. *arXiv preprint arXiv:2211.10581*, 2022.

[6] Xiuwei Xu, Huangxing Chen, Linqing Zhao, Ziwei Wang, Jie Zhou, and Jiwen Lu. Embodiedsam: Online segment any 3d thing in real time. *arXiv preprint arXiv:2408.11811*, 2024.

[7] David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3d semantic segmentation in the wild. In *European Conference on Computer Vision*, pages 125–141. Springer, 2022.

[8] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017.

[9] Binh-Son Hua, Quang-Hieu Pham, Duc Thanh Nguyen, Minh-Khoi Tran, Lap-Fai Yu, and Sai-Kit Yeung. Scenenn: A scene meshes dataset with annotations. In *2016 fourth international conference on 3D vision (3DV)*, pages 92–101. Ieee, 2016.

[10] Johanna Wald, Armen Avetisyan, Nassir Navab, Federico Tombari, and Matthias Nießner. Rio: 3d object instance re-localization in changing indoor environments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7658–7667, 2019.

[11] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021.

[12] 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.

[13] Tianhe Ren, Yihao Chen, Qing Jiang, Zhaoyang Zeng, Yuda Xiong, Wenlong Liu, Zhengyu Ma, Junyi Shen, Yuan Gao, Xiaoke Jiang, et al. Dino-x: A unified vision model for open-world object detection and understanding. *arXiv preprint arXiv:2411.14347*, 2024.

[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] Feng Li, Hao Zhang, Peize Sun, Xueyan Zou, Shilong Liu, Chunyuan Li, Jianwei Yang, Lei Zhang, and Jianfeng Gao. Segment and recognize anything at any granularity. In *European Conference on Computer Vision*, pages 467–484. Springer, 2024.

[16] Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. Slip: Self-supervision meets language-image pre-training. In *European conference on computer vision*, pages 529–544. Springer, 2022.

[17] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[18] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[19] Yunhan Yang, Xiaoyang Wu, Tong He, Hengshuang Zhao, and Xihui Liu. Sam3d: Segment anything in 3d scenes. *arXiv preprint arXiv:2306.03908*, 2023.

[20] Runnan Chen, Youquan Liu, Lingdong Kong, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, Yu Qiao, and Wenping Wang. Clip2scene: Towards label-efficient 3d scene understanding by clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7020–7030, 2023.

[21] Ayça Takmaz, Elisabetta Fedele, Robert W Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. Openmask3d: Open-vocabulary 3d instance segmentation. *arXiv preprint arXiv:2306.13631*, 2023.

[22] Yingda Yin, Yuzheng Liu, Yang Xiao, Daniel Cohen-Or, Jingwei Huang, and Baoquan Chen. Sai3d: Segment any instance in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3292–3302, 2024.

[23] Ji Hou, Angela Dai, and Matthias Nießner. 3d-sis: 3d semantic instance segmentation of rgb-d scans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4421–4430, 2019.

[24] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6411–6420, 2019.

[25] Le Hui, Linghua Tang, Yaqi Shen, Jin Xie, and Jian Yang. Learning superpoint graph cut for 3d instance segmentation. *Advances in Neural Information Processing Systems*, 35:36804–36817, 2022.

[26] Zhihao Liang, Zhihao Li, Songcen Xu, Mingkui Tan, and Kui Jia. Instance segmentation in 3d scenes using semantic superpoint tree networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2783–2792, 2021.

[27] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3d: Mask transformer for 3d semantic instance segmentation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8216–8223. IEEE, 2023.

[28] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3075–3084, 2019.

[29] Lei Han, Tian Zheng, Lan Xu, and Lu Fang. Occuseg: Occupancy-aware 3d instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2940–2949, 2020.

[30] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11108–11117, 2020.

[31] Siqi Fan, Qiulei Dong, Fenghua Zhu, Yisheng Lv, Peijun Ye, and Fei-Yue Wang. Scf-net: Learning spatial contextual features for large-scale point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14504–14513, 2021.

[32] Weiyue Wang, Ronald Yu, Qiangui Huang, and Ulrich Neumann. Sgpn: Similarity group proposal network for 3d point cloud instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2569–2578, 2018.

[33] Jiahao Sun, Chunmei Qing, Junpeng Tan, and Xiangmin Xu. Superpoint transformer for 3d scene instance segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 2393–2401, 2023.

[34] John McCormac, Ankur Handa, Andrew Davison, and Stefan Leutenegger. Semanticfusion: Dense 3d semantic mapping with convolutional neural networks. In *2017 IEEE International Conference on Robotics and automation (ICRA)*, pages 4628–4635. IEEE, 2017.

[35] Gaku Narita, Takashi Seno, Tomoya Ishikawa, and Yohsuke Kaji. Panopticfusion: Online volumetric semantic mapping at the level of stuff and things. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4205–4212. IEEE, 2019.

[36] Shi-Sheng Huang, Ze-Yu Ma, Tai-Jiang Mu, Hongbo Fu, and Shi-Min Hu. Supervoxel convolution for online 3d semantic segmentation. *ACM Transactions on Graphics (TOG)*, 40(3):1–15, 2021.

[37] Jiazhao Zhang, Chenyang Zhu, Lintao Zheng, and Kai Xu. Fusion-aware point convolution for online semantic 3d scene segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4534–4543, 2020.

[38] Leyao Liu, Tian Zheng, Yun-Jou Lin, Kai Ni, and Lu Fang. Ins-conv: Incremental sparse convolution for online 3d segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18975–18984, 2022.

[39] Yijie Tang, Jiazhao Zhang, Yuqing Lan, Yulan Guo, Dezun Dong, Chenyang Zhu, and Kai Xu. Onlineanyseg: Online zero-shot 3d segmentation by visual foundation model guided 2d mask merging. *arXiv preprint arXiv:2503.01309*, 2025.

[40] Xiuwei Xu, Chong Xia, Ziwei Wang, Linqing Zhao, Yueqi Duan, Jie Zhou, and Jiwen Lu. Memory-based adapters for online 3d scene perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21604–21613, 2024.

[41] Rajesh PN Rao. A sensory–motor theory of the neocortex. *Nature neuroscience*, 27(7):1221–1235, 2024.

[42] Marina A Lynch. Long-term potentiation and memory. *Physiological reviews*, 84(1):87–136, 2004.

[43] Randall C O'Reilly, Rajan Bhattacharyya, Michael D Howard, and Nicholas Ketz. Complementary learning systems. *Cognitive science*, 38(6):1229–1248, 2014.

[44] Larry R Squire, Lisa Genzel, John T Wixted, and Richard G Morris. Memory consolidation. *Cold Spring Harbor perspectives in biology*, 7(8):a021766, 2015.

[45] Yadin Dudai, Avi Karni, and Jan Born. The consolidation and transformation of memory. *Neuron*, 88(1):20–32, 2015.

[46] Mutian Xu, Xingyilang Yin, Lingteng Qiu, Yang Liu, Xin Tong, and Xiaoguang Han. Sampro3d: Locating sam prompts in 3d for zero-shot scene segmentation. *arXiv preprint arXiv:2311.17707*, 2023.

[47] Phuc Nguyen, Tuan Duc Ngo, Evangelos Kalogerakis, Chuang Gan, Anh Tran, Cuong Pham, and Khoi Nguyen. Open3dis: Open-vocabulary 3d instance segmentation with 2d mask guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4018–4028, 2024.

[48] Jinglin Zhao, Debin Liu, Laurence T Yang, Ruonan Zhao, Zheng Wang, and Zhe Li. Td3d: Tensor-based discrete diffusion process for 3d shape generation. In *2024 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2024.

[49] Maxim Kolodiazhnyi, Anna Vorontsova, Anton Konushin, and Danila Rukhovich. Oneformer3d: One transformer for unified point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20943–20953, 2024.

[50] Xu Zhao, Wenchao Ding, Yongqi An, Yinglong Du, Tao Yu, Min Li, Ming Tang, and Jinqiao Wang. Fast segment anything. *arXiv preprint arXiv:2306.12156*, 2023.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Please refer to Sec. 1.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please refer to Sec. 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Please refer to Sec. 3.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Please refer to Sec. 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Code and raw results will be publicly available upon acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Please refer to Sec. 4.1.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: It would be too computationally expensive to report Error bars.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Please refer to Sec. 4.1.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics and we confirm that the research conducted in the paper conforms, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Please refer to Sec. 1.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The assets used in this paper are credited and the license is respected.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: Code and raw results will be publicly available upon acceptance and we will include details about training, license, limitations, etc.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The research in this paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The research in this paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: We have reviewed the LLM policy and we confirm that the core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.