PARTNER: LEARNING TO COORDINATE PARTS FOR PROMPTABLE 3D HOLISTIC SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

As 3D assets possess inherent hierarchical structures, semantic granularities vary across subjective user intents. Although existing interactive methods refine individual part masks, extending them to holistic segmentation remains challenging due to boundary overlaps or semantic missing. To address this gap, we propose Partner, a learning-based framework for real-time, interactive holistic segmentation. In each round, Partner updates the entire partition based on user prompts for a single part. Through its Semantic Coordinator and adaptive Split-Fill Predictor, Partner achieves unified, instruction-guided global adjustments, outperforming heuristic baselines in both refinement quality and interaction efficiency while serving as a versatile plug-and-play module.

1 INTRODUCTION

Fine-grained segmentation of 3D assets plays a fundamental role in downstream applications such as 3D object understanding, asset repurposing, and structured generation. However, 3D assets possess an inherent hierarchical structure, and the requirement for semantic granularity varies significantly between different application scenarios, making it impossible for automatic holistic segmentation models (Ma et al., 2025a) to achieve flexible semantic control. For instance, in character rigging, the hand and arm might be treated as a single unified component, while for surface texture refinement, it is necessary to decouple the fingers or even individual nails. To accommodate dynamic user intent, interactive models (Zhou et al., 2025; Zhu et al., 2025) take point prompts as conditions to flexibly adjust semantic granularity. Nevertheless, these approaches are predominantly focused on isolated mask extraction. Extending them to holistic segmentation contexts requires complex heuristics such as area-based filtering and flood-filling. However, since the target mask is predicted without knowing the extent of the other parts, these heuristics can only repair the inconsistency after it appears. They often lead to regional overlap or semantic missing, failing to achieve efficient and coherent global adjustments.

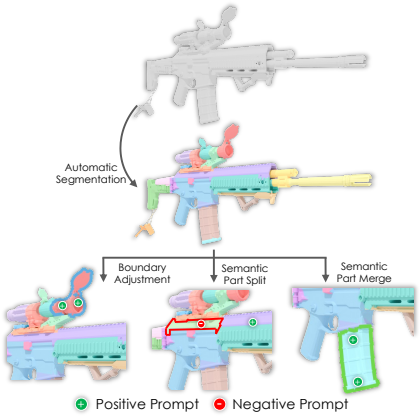


Figure 1: Partner enables efficient and precise promptable holistic segmentation.

To bridge this gap, we propose Partner, an interactive framework designed for holistic segmentation contexts. As shown in Fig. 1, Partner enables efficient and precise 3D holistic segmentation, supporting user-prompted merging, splitting, and boundary adjustments. Departing from the conventional paradigm of predicting masks in isolation, Partner captures and learns the global correlations and semantic relationship between parts to coherently coordinate semantic granularity and mask in strict accordance with specific user intent.

Fig. 2 illustrates the two-stage framework of Partner. In the first stage, a single-mask interactive model adjusts the boundary of the primary part selected by the user, so that each adjustment faithfully follows the semantic boundary intended by the user. In the second stage, **Adaptive Holistic Segmentation Refinement**, two key modules infer the semantic boundaries of the remaining parts from the context:

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

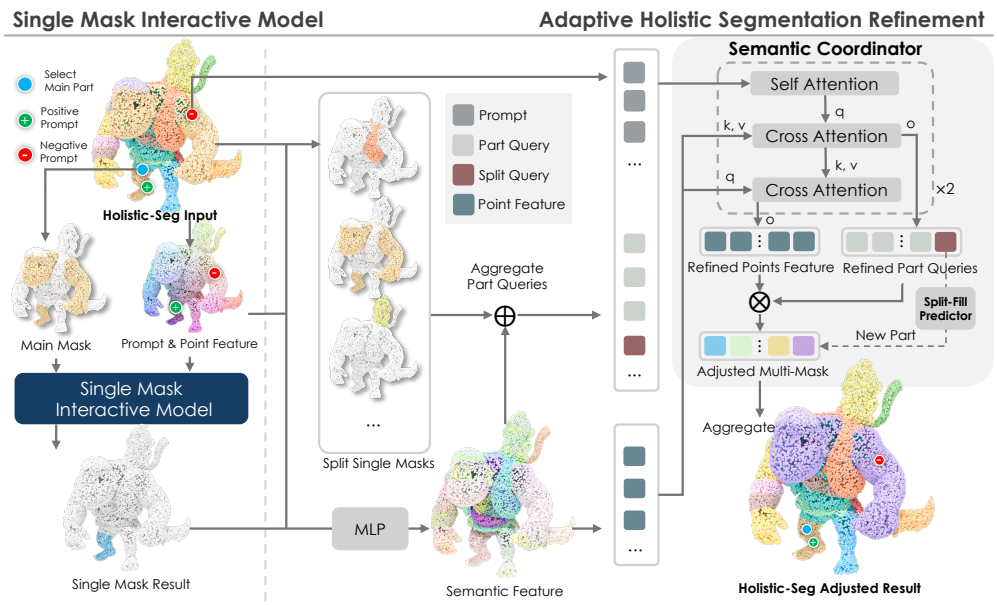


Figure 2: Architecture. (1) The primary part mask is first adjusted interactively. (2) Individual masks decomposed from the holistic context yield part queries via point feature aggregation. The Semantic Coordinator then fuses user prompts into these queries, jointly refining queries and point features to update the holistic segmentation. A Split-Fill predictor with a learnable query is also incorporated for part emergence.

Semantic Coordinator. The Semantic Coordinator translates global part distributions into semantic queries and aligns them with user prompts. Instead of repairing mask inconsistencies via heuristic post-processing, it coordinates all components simultaneously within a unified framework, delivering harmonized semantic boundaries and strict global continuity within one prediction.

Adaptive Split-Fill Predictor. Instead of determining the emergence of new semantic parts heuristically, we design a split-fill predictor that interprets user intent to decide whether to partition a segment (Split) or consolidate redundant regions through semantic filling from neighbors (Fill). This strategic decision-making prevents semantic conflicts and ensures a seamless transition between segments.

Combining both stages, Partner operates like a single-mask interactive model: users select a primary part and add positive/negative prompts to instantly update the holistic segmentation. A full refinement consists of multiple rounds targeting different parts until satisfied.

To address the scarcity of large-scale 3D datasets with multi-level semantic hierarchies, we propose a specialized dynamic data synthesis pipeline. By generating pseudo-labels for under- and over-segmented states from static datasets, the pipeline simulates diverse user intents and derives iterative prompts. Furthermore, mask perturbation is incorporated to force the model to learn robust boundary rectification and fine-grained optimization.

Extensive experiments demonstrate that Partner reaches a higher segmentation quality than the single-mask iterative paradigm under the same interaction budget. The primary contributions of Partner are summarized as follows:

1. Novel Interactive Paradigm: We propose Partner, a learning-based framework tailored for promptable 3D holistic segmentation. Powered by a Semantic Coordinator and an adaptive Split-Fill Predictor, Partner achieves dynamically coordinated merging, splitting, and boundary refinement.

2. Dynamic Data Pipeline: We design a specialized dynamic data synthesis strategy to construct multi-level hierarchical 3D annotations. Our pipeline effectively trains the model to handle complex multi-part interaction scenarios.

3. Interaction Efficiency & Versatility: Partner significantly reduces the required user interaction rounds. As a plug-and-play module, Partner seamlessly integrates with diverse off-the-shelf 3D foundation models, continuously capitalizing on their ongoing advancements.

2 RELATED WORKS

2.1 AUTOMATIC PART SEGMENTATION

Automatic part segmentation serves as a critical bridge for transforming unstructured 3D assets into industry-grade, editable models. Early works (Qi et al., 2017a;b; Graham et al., 2018; Thomas et al., 2019; Wu et al., 2019) focused on closed-world segmentation, where categories are confined to specific labels within the training dataset (Armeni et al., 2016; Dai et al., 2017; Mo et al., 2019), making it hard to handle unseen data. Driven by the remarkable progress in 2D foundation models (Radford et al., 2021; Li* et al., 2022; Kirillov et al., 2023; Oquab et al., 2023) and VLM models (Google, 2025; OpenAI, 2025), open-world automatic 3D segmentation has evolved significantly by leveraging advanced image-text alignment and zero-shot capabilities from these models. A prevalent and effective strategy involves rendering 3D assets into multi-view images to lift these robust 2D priors into 3D space (Tang et al., 2025; Yang et al., 2023; Xu et al., 2025; Abdelreheem et al., 2023; Zhou et al., 2023; Liu et al., 2023). Specifically, SAMesh, SAM3D, and SAMPro3D (Tang et al., 2025; Yang et al., 2023; Xu et al., 2025) directly apply SAM to rendered views and aggregate multi-view masks to achieve automated, class-agnostic segmentation for arbitrary 3D objects, lifting 2D semantics into 3D space via multi-view projections. While, PartSlip and PartSlip++ (Liu et al., 2023; Zhou et al., 2023) utilized GLIP. More recently, P^3-SAMand PartSAM (Ma et al., 2025a; Zhu et al., 2025) perform large-scale pre-training on native 3D representations, enhancing the holistic segmentation results on complex 3D assets. PartField (Liu et al., 2025) models part-based feature fields, which extends the capability to handle segmentation tasks across diverse 3D input modalities. Some approaches like NaTex (Lai et al., 2025) formulate component partitioning as a surface texturing task, which enables precise prediction of part boundaries.

2.2 INTERACTIVE 3D SEGMENTATION

Interactive segmentation empowers users to guide models according to specific intents. While the Segment Anything Model (SAM) (Kirillov et al., 2023) established a groundbreaking promptable paradigm in 2D, its core principles have inspired a series of works in 3D domain. Pioneering works like InterObject3D (Kontogianni et al., 2023) and AGILE3D (Yue et al., 2024), first achieved success in segmenting indoor scenes within datasets like ScanNet. Building upon these and leveraging 2D-to-3D data engines, recent frameworks like Point-SAM and Part-SAM (Zhou et al., 2025; Zhu et al., 2025) have further extended interactive capabilities, enabling robust, localized component extraction across a much broader range of object domains. Besides, some methods (Ma et al., 2025b; Paul et al., 2025; Jin et al., 2026) leverage text as the segmentation control for interaction. Nevertheless, existing 3D interactive frameworks predominantly focus on the extraction of isolated, individual parts. When applied to a complete holistic segmentation, these methods exhibit limitations: independently adjusting one part often induces semantic overlaps, missing, or erroneous region expansions relative to its neighbors. To bridge this gap, Partner introduce the Semantic Coordinator and Split-Fill Predictor to coordinate part segmentation simultaneously.

2.3 HIERARCHICAL AND MULTI-CRITERION 3D SEMANTIC DATASETS

Traditional 3D part datasets (Deitke et al., 2023; 2022; Dai et al., 2017) are typically constructed upon predefined annotations, supporting only a single semantic partitioning scheme. To better align with the hierarchical nature of 3D assets in real-world applications, recent research has introduced datasets featuring multi-level annotations or multi-criteria semantics, such as PartObj-Tiny and PartNeXt (Yang et al., 2024; Wang et al., 2026). However, the reliance on manual multi-level labeling poses significant challenges for large-scale expansion for training generalized holistic interactive 3D segmentation models. Therefore, we propose a dynamic data annotation construction for handling diverse interactive intents.

3 PROBLEM FORMULATION

Holistic segmentation. Given a 3D mesh $\mathcal{M} = (\mathcal{V}, \mathcal{F})$, the objective of holistic segmentation is to partition $\mathcal{M}$ into K disjoint semantic components, each identified by a category-agnostic instance label. Following Ma et al. (2025a), we sample a point cloud $\mathcal{X}$ of N points from the mesh surface $\mathcal{F}$

and solve the problem in the point domain, using the sampled points as a proxy for the underlying faces. Formally, we define the holistic segmentation result as a set of K individual part masks. Let $\mathbf{m}_k \in \{0,1\}^N$ denote the binary mask for the k-th component. The holistic segmentation state is constrained to ensure an exhaustive and mutually exclusive partition of the 3D asset:

$$\mathcal{S} = \{\mathbf{m}_k\}_{k=1}^{K}, \quad \text{s.t.} \quad \sum_{k=1}^{K} \mathbf{m}_k = \mathbf{1}^N. \tag{1}$$

Throughout this paper, all masks are point-wise indicator vectors defined over the N sampled points, and the model operates entirely in the point domain. Face-level labels are produced only at the very end, by assigning each face the majority label among its nearest sampled points (Table 4).

Promptable Holistic Segmentation. Given the initial segmentation $\mathcal{S}_{init}$ produced by an existing method, Partner evolves it into an updated holistic result $\mathcal{S}_{opt}$, conditioned on the mesh $\mathcal{M}$ and the user prompts $\mathcal{P}$. For the interaction, the user should first select the primary part as the mask $\mathbf{m}_a$. The primary part is the component that the current round of interaction targets, and it serves as the semantic reference of the prompts. The other components require no prompt, but they are still reassigned in the output. Then, similar to Zhou et al. (2025), the user's interactive adjust intent is represented by a set of sparse prompts $\mathcal{P} = \{(p_j, y_j)\}_{j=1}^{J}$, where p_j denotes the prompt coordinates and $y_j \in \{0,1\}$ distinguishes between positive and negative signals. Both types of prompts are defined with respect to the primary part alone: a positive prompt marks a point that should belong to $\mathbf{m}_a$, and a negative one marks a point that should not. A set of prompts is thus always associated with a single component rather than the entire object: the user selects one component at a time and provides prompts interactively, and repeating this process over components leads to the adjustment of the whole holistic segmentation.

Atomic operations. To precisely characterize the consistency adjustments in holistic segmentation, we reduce the iterative process to three topological atomic operations, namely **Merge**, **Split** and **Boundary Refinement**, as summarized in Table 1. Let K' denote the total number of parts following the adjustment. The result of all three operations still satisfies the partition constraint in Eq. 1, with K replaced by K'. These operations specify the *intended outcome* of an interaction rather than the mechanism used to realize it, and Partner attains them through a single global reassignment, as described in 4.2.

Operation	**Effect on the partition**	$\boldsymbol{K'}$
Merge	$\mathbf{m}_a$ and $\mathbf{m}_b$ are replaced by a single component covering both	$K-1$
Split	A new component emerges from the region of $\mathbf{m}_a$	$K+1$
Boundary Refinement	Points are reassigned among the existing components	K

Table 1: Three atomic operations that characterize an interactive adjustment of a holistic segmentation. $\mathbf{m}_a$ and $\mathbf{m}_b$ denote the masks involved in the user-specified adjustment.

4 METHOD

4.1 TWO-STAGE INTERACTIVE HOLISTIC SEGMENTATION

As defined in 3, for a selected primary part $\mathbf{m}_a$, the user provides multiple rounds of point prompts $\mathcal{P}$ to adjust its boundary, upon which Partner dynamically re-derives the full partition. Repeating this workflow across different parts progressively evolves $\mathcal{S}_{init}$ into the final target segmentation.

To achieve this, each round follows a two-stage process (Fig. 2): first adjusting the primary part's semantic granularity, and then restoring a valid overall partition. In the first stage, the primary part is adjusted in isolation via a promptable single-mask predictor Φ_{single}. Taking point-wise features $\mathbf{F} \in \mathbb{R}^{N \times C}$, prompts $\mathcal{P}$, and the current primary mask $\mathbf{m}_a$ as input, Φ_{single} outputs an updated mask $\mathbf{m}_a^*$. Although optimized, $\mathbf{m}_a^*$ does not yet form a valid partition on its own, as it may introduce boundary overlaps or semantic missing with respect to neighboring components.

The second stage resolves these conflicts by taking $\mathbf{m}_a^*$ alongside $\mathcal{S}_{init}$ as context and reassigning all point labels in a single pass, thereby achieving global coordination without heuristic post-processing.

4.2 Adaptive Holistic Segmentation Refinement

To address the boundary overlaps and missing semantics left by the isolated mask $\mathbf{m}_a^*$, we design the holistic coordination module $\Phi_{holistic}$ in the second stage. Taking $\mathbf{F}$, $\mathbf{m}_a^*$, $\mathcal{S}_{init}$, and $\mathcal{P}$ as inputs, $\Phi_{holistic}$ reassigns every point to produce the optimized holistic segmentation $\mathcal{S}_{opt}$, transitioning the model from single-part prediction to global context awareness.

Part-aware Holistic Semantic Coordinator. First, to enhance the semantic discriminability of point-wise features and incorporate the adjusted single-mask prior from the first stage, we encode the initial segmentation $\mathcal{S}_{init}$ and the mask $\mathbf{m}_a^*$, concatenate the two encodings with $\mathbf{F}$ and compress the result back to the original width through a two-layer MLP, yielding the semantic feature $\mathbf{F}'$. Then, for each part k, we derive the query vector $\mathbf{q}_k$ by masked average pooling, averaging $\mathbf{F}'$ over the points covered by $\mathbf{m}_k$. Each query thus summarizes the geometry and semantics of one part with a single vector, and the whole segmentation state is represented by the query set $\{\mathbf{q}_k\}_{k=1}^K$ instead of K separate masks.

The queries are then coordinated in two steps. We first apply an attention layer over the query set together with the encoded user prompts $\mathcal{P}$, so that each query carries the user intent. The resulting queries and the semantic features $\mathbf{F}'$ are fed into a Two-Way Transformer. Within each block, the queries first exchange information among themselves through self-attention, then attend to the point features, and the coordinated part-level context is in turn used to update the point features. We denote the refined part queries and point features as $\{\mathbf{q}_k^{ref}\}$ and $\mathbf{F}^{ref}$. Note that $\mathcal{P}$ has already been consumed by the first stage; re-injecting it here lets the model recover the exact adjustment boundary intended by the user, rather than inferring it from $\mathbf{m}_a^*$ alone. We ablate this design in 5.3.

Then, we generalize the single-mask prediction head in Ma et al. (2025a) to a multi-mask generator. The final segmentation result is derived by matching the refined features $\mathbf{F}_i^{ref}$ and the part-level queries $\mathbf{q}_k^{ref}$:

$$l_i = \arg\max_{k \in \mathcal{K}} \frac{\mathbf{F}_i^{ref} \cdot \mathbf{q}_k^{ref}}{\|\mathbf{F}_i^{ref}\| \|\mathbf{q}_k^{ref}\|}, \quad i = 1, \ldots, N, \tag{2}$$

where l_i denotes the point-wise label for each point i, and $\mathcal{K}$ represents the dynamic index set of component queries. By calculating the similarity between each point and queries, the model effectively reassigns the membership of every point to the most semantically compatible part, ensuring a consistent and globally optimized holistic output. Finally, we gather each label to generate masks $\{\mathbf{m}_1^{opt}, \ldots, \mathbf{m}_K^{opt}\}$ for all parts (see Appendix.B for details).

Adaptive Split-Fill Predictor. To address the emergence of a new semantic part during atomic operations, in addition to part queries $\{\mathbf{q}_k\}$, we introduce a dedicated learnable query $\mathbf{q}_{split}$ to represent the potential emerging part, together with a learnable decision token $\mathbf{q}_{dec}$ that carries no mask. Both are randomly initialized, shared across all objects, and coordinated jointly with the part queries, so that a single pass produces $K+2$ refined queries: $K+1$ of them generate masks, while $\mathbf{q}_{dec}^{ref}$ is reserved for the split decision. Having already aggregated global context and user prompts $\mathcal{P}$ via the Semantic Coordinator, $\mathbf{q}_{split}^{ref}$ serves as a comprehensive descriptor of the latent semantics of the emerging area. Then, the Split-Fill predictor evaluates $\mathbf{q}_{dec}^{ref}$ to generate a split probability ρ. If ρ exceeds a predefined threshold τ, $\mathbf{q}_{split}^{ref}$ is considered as an active query and added to $\{\mathbf{q}_1^{ref}, \ldots, \mathbf{q}_K^{ref}, \mathbf{q}_{split}^{ref}\}$ for mask generation in Eq. 2. Otherwise, $\mathbf{q}_{split}^{ref}$ is discarded and the model retains the original query set.

The opposite case, Fill, arises when the primary component $\mathbf{m}_a^*$ shrinks or is merged into a neighbor, leaving points that no longer belong to any intended part. No extra query is needed here: since Eq. 2 assigns every point to its most similar query, these points are taken over by the neighboring component queries in the same pass, requiring no flood-fill or other heuristic post-processing.

4.3 Dynamic Data Annotation Construction

To equip the model with the ability to handle diverse interactive intents, we construct our training dataset derived from the existing annotation set $\mathcal{S}_{anno}$. By applying random transformations to

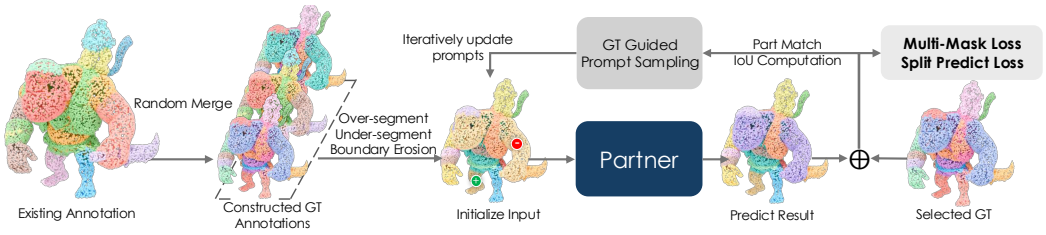


Figure 3: Data pipeline and training strategy. To map static annotations to diverse interactions, we apply data augmentations, iteratively sample prompts based on primary prediction errors, and optimize the model via average holistic segmentation loss across interaction rounds.

the original ground-truth labels, we simulate three typical scenarios: over-segmentation, under-segmentation, and boundary shifts. We demonstrate our data pipeline in Fig. 3.

First, we derive a hierarchical ground-truth set $\mathcal{S}_{gt}$ from a static holistic segmentation annotation $\mathcal{S}_{anno} = \{\mathbf{m}_k\}_{k=1}^{K}$ of the mesh $\mathcal{M}$ by performing random merging operations. Then, to construct the initial state $\mathcal{S}_{init}$ from the ground truth $\mathcal{S}_{gt}$, we randomly select a primary part $\mathbf{m}_a^{gt}$ and apply three distinct semantic transformations. These transformations are designed to force the model to learn the fundamental "atomic operations" required for interactive refinement.

Over-segmentation. We simulate over-segmentation by randomly partitioning $\mathbf{m}_a^{gt}$ into two disjoint subsets. This configuration compels the model to execute a "merge" operation to unify parts.

Under-segmentation. We simulate under-segmentation by pre-merging the primary part $\mathbf{m}_a^{gt}$ with a randomly selected part. In this scenario, the model must learn to perform a "split" operation to recover the original semantic parts.

Boundary Erosion. To simulate the imprecision inherent in initial segmentations, we apply a random erosion operator to all part masks. The faces are randomly reassigned to neighboring parts based on spatial proximity. This strategy mandates that the model develops robust boundary refinement capabilities to rectify local misalignments. Furthermore, by incorporating under-segmentation scenarios during training, the model concurrently learns to distinguish between different user intents, effectively optimizing the adaptive Split-Fill decisions to achieve a coherent semantic partition.

4.4 Multi-objective Loss Optimization

Since holistic segmentation involves reassigning labels across multiple parts, traditional loss functions designed for single-mask training cannot be directly applied. We therefore supervise the predicted masks individually and average their losses, pairing each predicted mask with its ground-truth counterpart through the correspondence π from 4.3. Supervision is applied to the $K+1$ masks $\{\mathbf{m}_k^{opt}\}_{k=1}^{K+1}$ predicted before the label assignment, in which the $(K+1)$-th mask is produced by the split query $\mathbf{q}_{split}$ and is supervised jointly with the K part masks:

$$\mathcal{L}_{mask} = \frac{1}{K+1}\sum_{k=1}^{K+1}\mathcal{L}_{seg}\big(\mathbf{m}_k^{opt}, \mathbf{m}_{\pi(k)}^{gt}\big), \quad \text{where } \mathcal{L}_{seg} = \mathcal{L}_{focal} + \lambda_{dice}\mathcal{L}_{dice}. \tag{3}$$

The summation runs to $K+1$ instead of K because the split query is always instantiated and always receives supervision, no matter whether the current interaction introduces a new part. When no new part should emerge, its target $\mathbf{m}_{\pi(K+1)}^{gt}$ is an all-zero vector, so the model is explicitly trained to predict an empty mask for this query. Supervising all queries uniformly allows the number of active parts to vary without modifying the network structure: a query whose mask is driven towards zero wins no point in Eq. 2 and thus disappears from the output. Both $\mathcal{L}_{focal}$ and $\mathcal{L}_{dice}$ are computed over the N sampled points.

Predicting the masks alone is not sufficient, as the model must also decide *whether* an additional component is required at all. We supervise the split probability ρ estimated by the Split-Fill Predictor with a binary cross-entropy term:

$$\mathcal{L}_{split} = \text{BCE}(\rho, y_{split}), \quad y_{split} \in \{0, 1\}. \tag{4}$$

Dataset	Method	Init	mIoU@1	mIoU@3	mIoU@5	mIoU@7	mIoU@10	Gain@10
PartObj-Tiny	PointSAM	32.9	36.8	40.5	42.0	42.9	43.2	+10.3
	PartSAM	47.3	48.9	53.2	56.1	57.1	57.5	+10.2
	Partner (single mask)	47.6	52.7	59.0	61.7	62.3	62.9	+15.2
	Partner (PointSAM init)	32.9	40.0	51.5	59.5	64.6	69.4	+36.5
	Partner (PartSAM init)	47.3	51.8	59.4	64.8	68.5	72.2	+24.9
	Partner (Ours)	47.6	**53.3**	**62.2**	**67.9**	**71.5**	**74.6**	+27.0
PartNext	PointSAM	46.4	54.5	59.2	60.7	61.3	61.5	+15.2
	PartSAM	55.8	63.6	69.6	70.8	71.0	71.3	+15.4
	Partner (single mask)	54.7	65.4	72.2	74.0	74.5	74.6	+19.9
	Partner (PointSAM init)	46.4	55.8	67.1	72.3	75.3	77.7	+31.3
	Partner (PartSAM init)	55.8	**63.9**	**73.6**	**78.0**	**80.2**	81.7	+25.9
	Partner (Ours)	54.7	63.9	73.4	77.7	79.9	**81.7**	+27.0
PartNet-Mobility	PointSAM	31.7	34.4	36.9	37.5	38.2	38.6	+6.89
	PartSAM	30.2	31.9	34.4	35.0	36.5	36.4	+6.12
	Partner (single mask)	37.5	39.8	42.6	43.2	43.9	44.5	+6.97
	Partner (PointSAM init)	31.7	37.5	46.9	52.5	56.6	60.7	+29.0
	Partner (PartSAM init)	30.2	34.8	43.5	48.8	52.6	56.7	+26.5
	Partner (Ours)	37.5	**44.3**	**51.1**	**56.5**	**60.4**	**62.7**	+25.2
Scannet	PointSAM	39.6	41.4	43.4	44.6	45.3	45.9	+6.38
	PartSAM	22.6	25.2	27.6	29.0	30.1	30.9	+8.25
	Partner (single mask)	24.2	27.4	31.4	33.8	35.4	37.0	+12.8
	Partner (PointSAM init)	39.6	**41.9**	**45.1**	**47.2**	**48.6**	**50.0**	+10.5
	Partner (PartSAM init)	22.6	26.0	31.3	35.0	37.7	40.4	+17.7
	Partner (Ours)	24.2	27.9	33.2	36.9	39.6	42.5	+18.4

Table 2: Quantitative comparison of zero-shot interactive segmentation refinement. The best scores are emphasized in bold. mIoU@T denotes mean IoU with T refined parts (each 5 prompt points).

Here y_{split} equals 1 if and only if the ground-truth state contains one more part than the input segmentation, that is, when the primary part has to be divided. It is 0 whenever the target can be reached by merging or by boundary refinement alone. Combined with the empty-target convention of Eq. 3, this term enables the model to determine the number of parts on its own during inference, rather than relying on a hand-tuned area threshold.

The final weighted total loss function is defined as:

$$\mathcal{L}_{total} = \alpha \mathcal{L}_{mask} + \beta \mathcal{L}_{split}. \tag{5}$$

5 Experiment

5.1 Implementation Details

We use the same dataset as P^3-SAM for training. For each mesh object, we sample 100,000 points and employ PTv3 (Wu et al., 2024; Zhang et al., 2025) as the backbone to extract point-wise features. For the first stage, we follow the training paradigm as Zhou et al. (2025). In the second stage, we freeze the first-stage model and exclusively train the holistic refinement module. Note that, consistent with the single-part interaction paradigm, we also adopt the multi-round prompt inputs for interactive prediction and sample the point prompts in the same way (see Appendix.A for details). By adopting this strategy, we effectively characterize the evolving user intent as the holistic segmentation results are refined from coarse to fine. Full architectural configurations are summarized in Table 4, and the complete set of hyper-parameters is provided in Appendix.D.

5.2 Zero-shot holistic segmentation Refinement

Evaluation Datasets. Following Zhou et al. (2025); Ma et al. (2025a); Zhu et al. (2025), we evaluated our framework across diverse datasets: PartObj-Tiny, PartNext, PartNet-Mobility, and ScanNet. Specifically, PartObj-Tiny includes instance-level GT for fine-grained components and semantic-level GT for broader categories. PartNeXt features a hierarchical structure tree; we pre-merged leaf nodes (the lowest level) to construct a hierarchy-level GT. To represent real-world demands for varying semantic granularities, we used these different GT levels to guide the adjustment process and evaluated performance using the IoU metric.

Task and Metric. The task is to refine an existing holistic segmentation through user inputs. In each interaction, the part with the lowest overlap compared to the GT is selected as the primary part for adjustment. The initial prompt is placed within the intersection of the primary part and the GT as a positive sample, while subsequent prompts are iteratively generated following the sampling strategy in training. To avoid spending prompts on parts that are already correct, we stop prompting a part once its IoU reaches 0.95, and stop the whole procedure once the least accurate part already exceeds 0.9. Since the number and indexing of predicted masks may differ from the ground truth, we employ a maximal-matching IoU for evaluation. Given the predicted mask set $\hat{\mathcal{S}} = \{\hat{\mathbf{m}}_i\}_{i=1}^{\hat{K}}$ and the ground truth set $\mathcal{S}_{gt} = \{\mathbf{m}_j^{gt}\}_{j=1}^{K_{gt}}$, the overall performance is measured by the mean IoU, calculated by averaging the best-matched IoU for each GT component:

$$mIoU = \frac{1}{K_{gt}} \sum_{j=1}^{K_{gt}} \left(\max_{k \in \{1,\dots,\hat{K}\}} \frac{\left|\hat{\mathbf{m}}_k \cap \mathbf{m}_j^{gt}\right|}{\left|\hat{\mathbf{m}}_k \cup \mathbf{m}_j^{gt}\right|} \right). \tag{6}$$

Here $\cap$, $\cup$ and $|\cdot|$ operate on the sets of points covered by the masks.

Baselines. As there is currently no existing framework specifically designed for holistic segmentation adjustment, we established a baseline by combining state-of-the-art single-mask models with a flood-fill algorithm. In this baseline paradigm, a single-mask model first iteratively refines the primary mask, which is then directly overlaid onto the initial holistic segmentation result. To resolve subsequent inconsistencies, we apply an area-based heuristic: disconnected regions smaller than a threshold τ_{area} are merged into adjacent components via a point cloud flood-fill algorithm, while unassigned regions exceeding τ_{area} are designated with new part IDs. Please refer to Appendix. E for an ablation study on the choice of τ_{area}. Among these baselines, *Partner (single mask)* is a controlled counterpart of our full model: it uses the same backbone and the same first-stage single-mask predictor, and only replaces the second stage with the heuristic above. Comparing it with *Partner (Ours)* therefore isolates the contribution of the Semantic Coordinator, since the two start from an identical initial segmentation. Multi-object interactive methods such as AGILE3D (Yue et al., 2024) are not included, as they require the key points of all the components to be given together with their component labels in a single pass, and therefore cannot take an automatic holistic segmentation as their starting point.

Results. As shown in Table 2, Partner significantly outperforms the "single-mask model + flood-fill" baseline in zero-shot holistic segmentation adjustment tasks. Due to the inherent differences in initial segmentation granularities between PartNext and PartObj-Tiny, the performance of unadjusted models varies. Throughout the refinement process, Partner not only demonstrates a higher performance ceiling for optimization but also exhibits superior interaction efficiency, enabling users to achieve high-quality results with substantially fewer prompt inputs. Qualitative comparisons between baselines and Partner on PartObj-Tiny and PartNeXt are illustrated in Fig. 4. We further compare the methods on AIGC assets in Appendix. G, where the user intent is expressed by textual descriptions and visual prompt demonstrations instead of being guided by the GT. This setting is closer to real-world usage and allows the behaviour of each method to be compared directly.

Models trained on diverse data distributions typically exhibit varying holistic segmentation patterns. When applying our framework across different domains, the model must handle a wide range of initial segmentations. We performed experiments to assess how different initial segmentations influence the results. As demonstrated in Table 2, Partner exhibits remarkable adaptability across various model outputs. Notably, the final refinement quality improves in tandem with the precision of the initial holistic segmentation. This strong positive correlation, combined with the model's ability to handle suboptimal inputs, fully demonstrates the plug-and-play versatility of our approach.

5.3 Ablation Study

To investigate the effectiveness of our design, we conducted ablation studies on the PartObj-Tiny and PartNeXt datasets, with results summarized in Table 3. We first evaluate the architecture, particularly focusing on the two-stage multi-head framework. Since the Split-Fill Predictor is fundamental to the model's ability to generate new semantic components, it is maintained across all structural variants.

We compare three ablated versions against our full method. The first variant excludes the single-mask predictor from the first stage, relying solely on the second-stage holistic adjustment. The second

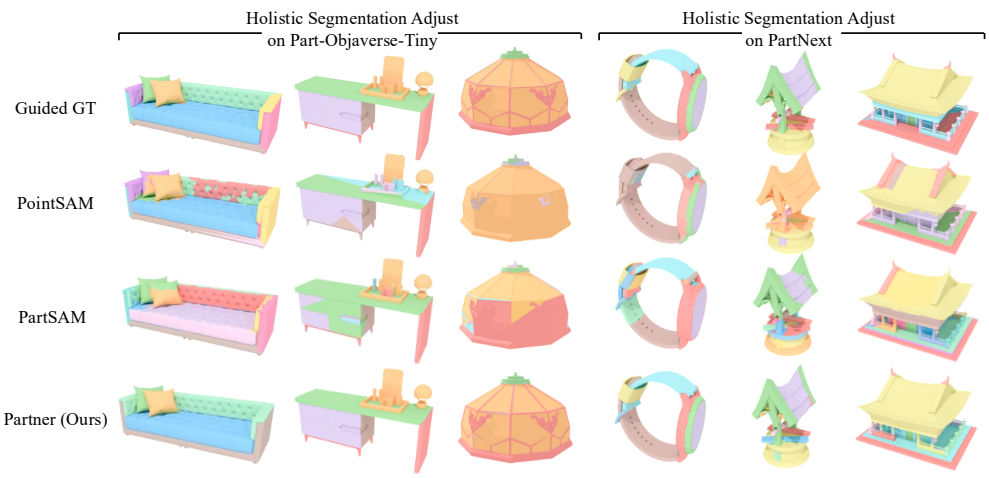


Figure 4: Qualitative comparisons between baselines and Partner on PartObj-Tiny and PartNeXt datasets. For each case, the results are obtained by performing 5 parts refinement guided by GT.

Dataset	Variant	Init	mIoU@1	mIoU@3	mIoU@5	mIoU@7	mIoU@10	Gain@10
PartObj-Tiny	w/o stage 1	47.6	51.2	56.6	60.1	62.0	63.5	+15.9
	w/o stage 2 prompt	47.6	52.1	57.2	60.5	62.8	64.8	+17.2
	point sampling	47.6	52.7	59.5	63.6	65.9	68.0	+20.3
	w/o erosion aug	47.6	52.7	59.8	64.2	67.2	70.0	+22.3
	Full	47.6	**53.3**	**62.2**	**67.9**	**71.5**	**74.6**	**+27.0**
PartNext	w/o stage 1	54.7	61.5	68.5	71.9	73.7	75.0	+20.3
	w/o stage 2 prompt	54.7	60.2	65.4	68.1	70.0	72.0	+17.2
	point sampling	54.7	62.6	70.2	73.6	75.2	76.6	+21.9
	w/o erosion aug	54.7	62.6	70.5	74.2	76.4	78.3	+23.6
	Full	54.7	**63.9**	**73.4**	**77.7**	**79.9**	**81.7**	**+27.0**

Table 3: Ablation study on model architecture and training strategy. The best scores are emphasized in bold. mIoU@T denotes mean IoU with T refined parts (each 5 prompt points).

variant omits user prompt injection during the second stage, attempting to refine boundaries based only on the first-stage output and the initial holistic context. Our full version outperforms both, clearly demonstrating that the synergy between localized single-mask refinement and prompt-guided holistic coordination is essential for accurate segmentation.

In single-mask segmentation models, point prompts are typically used to represent a component (Zhou et al., 2025; Zhu et al., 2025). Following this paradigm, we implemented a baseline aggregating method by encoding a randomly sampled point from each component. Our experimental results demonstrate that feature-aggregated query outperforms the point-sampling approach in holistic segmentation adjustment. We hypothesize that a single sampled point fails to capture the comprehensive semantics of a component and is highly susceptible to the variability of the sampling location.

Furthermore, we assess our data construction pipeline to validate the importance of specific transformations. The fourth ablated model trained on the dataset only utilized random merging and splitting for data construction, without boundary erosion. The drop in performance indicates that random boundary erosion is necessary for the model to rectify local misalignments and make intelligent decisions regarding semantic splitting.

6 Conclusion

In this paper, we present Partner, a promptable framework for 3D holistic segmentation. Departing from isolated mask predictions, Partner models global correlations via Semantic Coordinator and dynamically adjusts semantic granularities using an adaptive Split-Fill Predictor. This design resolves boundary conflicts and semantic fractures, ensuring global consistency. Experiments show that Partner surpasses heuristic baselines in quality and efficiency, particularly on complex objects. Furthermore, Partner seamlessly integrates with diverse 3D foundation models to enhance their outputs, bridging the gap between automated tools and user intent for high-quality 3D asset modeling.

AI USE STATEMENT

We used generative AI tools only for editing assistance on the manuscript, namely grammar, spelling and word choice. We did not use generative AI tools for research ideation, method design, experiment design, data construction or result analysis, and no part of the reported results was produced by such tools. All AI-assisted edits were reviewed by the authors, and we take responsibility for the final content of this work.

ETHICS STATEMENT

Based on our review, the datasets and visual materials used in this work do not raise identifiable ethical, security, copyright, or privacy concerns. We use these resources solely for research purposes and follow their applicable licenses and terms of use.

REPRODUCIBILITY STATEMENT

The task setting and the notation are defined in 4.1, and the two stages of Partner are described in 4.2. The training objective is given in 4.4, and the construction of the training data is described in 4.3. The complete set of hyper-parameters, including the optimizer, the learning rate, the training schedule and the point-to-mesh mapping, is provided in Table 4 of Appendix. D. The evaluation metric is defined in Eq. 6, together with the protocol for selecting the primary part and the stopping criteria. The prompt sampling procedure used in both training and evaluation is specified in Appendix. A, and the aggregation of point labels into part masks in Appendix. B. The datasets used for evaluation are all publicly available.

REFERENCES

Ahmed Abdelreheem, Ivan Skorokhodov, Maks Ovsjanikov, and Peter Wonka. Satr: Zero-shot semantic segmentation of 3d shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15166–15179, 2023.

Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1534–1543, 2016.

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*, pp. 5828–5839, 2017.

Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. *arXiv preprint arXiv:2212.08051*, 2022.

Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, Eli VanderBilt, Aniruddha Kembhavi, Carl Vondrick, Georgia Gkioxari, Kiana Ehsani, Ludwig Schmidt, and Ali Farhadi. Objaverse-xl: A universe of 10m+ 3d objects. *arXiv preprint arXiv:2307.05663*, 2023.

Google. Gemini-2.5-pro, 2025. `https://deepmind.google/technologies/gemini/pro/`.

Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9224–9232, 2018.

Li Jin, Weikai Chen, Yujie Wang, Yingda Yin, Zeyu Hu, Runze Zhang, Keyang Luo, Shengju Qian, Xin Wang, and Xueying Qin. Cosmo3d: Open-world promptable 3d semantic part segmentation through llm-guided canonical spatial modeling. *arXiv preprint arXiv:2603.01205*, 2026.

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*, pp. 4015–4026, 2023.

Theodora Kontogianni, Ekin Celikkan, Siyu Tang, and Konrad Schindler. Interactive object segmentation in 3d point clouds. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2891–2897. IEEE, 2023.

Zeqiang Lai, Yunfei Zhao, Zibo Zhao, Xin Yang, Xin Huang, Jingwei Huang, Xiangyu Yue, and Chunchao Guo. Natex: Seamless texture generation as latent color diffusion. *arXiv preprint arXiv:2511.16317*, 2025. URL https://arxiv.org/abs/2511.16317.

Liunian Harold Li*, Pengchuan Zhang*, Haotian Zhang*, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, Kai-Wei Chang, and Jianfeng Gao. Grounded language-image pre-training. In *CVPR*, 2022.

Minghua Liu, Yinhao Zhu, Hong Cai, Shizhong Han, Zhan Ling, Fatih Porikli, and Hao Su. Partslip: Low-shot part segmentation for 3d point clouds via pretrained image-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 21736–21746, 2023.

Minghua Liu, Mikaela Angelina Uy, Donglai Xiang, Hao Su, Sanja Fidler, Nicholas Sharp, and Jun Gao. Partfield: Learning 3d feature fields for part segmentation and beyond. *arXiv preprint arXiv:2504.11451*, 2025. URL https://arxiv.org/abs/2504.11451.

Changfeng Ma, Yang Li, Xinhao Yan, Jiachen Xu, Yunhan Yang, Chunshi Wang, Zibo Zhao, Yanwen Guo, Zhuo Chen, and Chunchao Guo. P3-sam: Native 3d part segmentation. *arXiv preprint arXiv:2509.06784*, 2025a. URL https://arxiv.org/abs/2509.06784.

Ziqi Ma, Yisong Yue, and Georgia Gkioxari. Find any part in 3d. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7818–7827, 2025b.

Kaichun Mo, Shilin Zhu, Angel X Chang, Li Yi, Subarna Tripathi, Leonidas J Guibas, and Hao Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 909–918, 2019.

OpenAI. Gpt-4o, 2025. https://openai.com/index/introducing-4o-image-generation/.

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.

Soumava Paul, Prakhar Kaushik, Ankit Vaidya, Anand Bhattad, and Alan Yuille. Name that part: 3d part segmentation and naming. *arXiv preprint arXiv:2512.18003*, 2025.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017a.

Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017b.

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*, pp. 8748–8763. PMLR, 2021.

George Tang, William Zhao, Logan Ford, David Benhaim, and Paul Zhang. Segment any mesh. *arXiv preprint arXiv:2408.13679*, 2025. URL https://arxiv.org/abs/2408.13679.

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*, pp. 6411–6420, 2019.

Penghao Wang, Yiyang He, Xin Lv, Yukai Zhou, Lan Xu, Jingyi Yu, and Jiayuan Gu. Partnext: A next-generation dataset for fine-grained and hierarchical 3d part understanding. *arXiv preprint arXiv:2510.20155*, 2026. URL https://arxiv.org/abs/2510.20155.

Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pp. 9621–9630, 2019.

Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler, faster, stronger. In *CVPR*, 2024.

Mutian Xu, Xingyilang Yin, Lingteng Qiu, Yang Liu, Xin Tong, and Xiaoguang Han. Sampro3d: Locating sam prompts in 3d for zero-shot instance segmentation. *arXiv preprint arXiv:2311.17707*, 2025. URL https://arxiv.org/abs/2311.17707.

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

Yunhan Yang, Yukun Huang, Yuan-Chen Guo, Liangjun Lu, Xiaoyang Wu, Edmund Y Lam, Yan-Pei Cao, and Xihui Liu. Sampart3d: Segment any part in 3d objects. *arXiv preprint arXiv:2411.07184*, 2024.

Yuanwen Yue, Sabarinath Mahadevan, Jonas Schult, Francis Engelmann, Bastian Leibe, Konrad Schindler, and Theodora Kontogianni. AGILE3D: Attention Guided Interactive Multi-object 3D Segmentation. In *International Conference on Learning Representations (ICLR)*, 2024.

Yujia Zhang, Xiaoyang Wu, Yixing Lao, Chengyao Wang, Zhuotao Tian, Naiyan Wang, and Hengshuang Zhao. Concerto: Joint 2d-3d self-supervised learning emerges spatial representations. In *NeurIPS*, 2025.

Yuchen Zhou, Jiayuan Gu, Xuanlin Li, Minghua Liu, Yunhao Fang, and Hao Su. Partslip++: Enhancing low-shot 3d part segmentation via multi-view instance segmentation and maximum likelihood estimation. *arXiv preprint arXiv:2312.03015*, 2023.

Yuchen Zhou, Jiayuan Gu, Tung Yen Chiang, Fanbo Xiang, and Hao Su. Point-SAM: Promptable 3d segmentation model for point clouds. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=yXCTDhZDh6.

Zhe Zhu, Le Wan, Rui Xu, Yiheng Zhang, Honghua Chen, Zhiyang Dou, Cheng Lin, Yuan Liu, and Mingqiang Wei. Partsam: A scalable promptable part segmentation model trained on native 3d data. *arXiv preprint arXiv:2509.21965*, 2025.

A GT Guided Iterative Prompt Sampling

To simulate authentic user interaction during training, we adopt an iterative training scheme following Zhou et al. (2025). The sampling process begins by selecting the initial prompt $p^{(1)}$ from the intersection of the primary part and the ground truth to establish a positive starting point. For subsequent steps $t > 1$, a new prompt point $p^{(t)}$ is sampled based on the discrepancy between the prediction from the previous iteration $\mathbf{m}_a^{(t-1)}$ and the ground truth $\mathbf{m}_a^{gt}$:

$$p^{(1)} \sim \text{Sample}(\mathbf{m}_a^{init} \cap \mathbf{m}_a^{gt}), \quad p^{(t)} \sim \text{Sample}\left((\mathbf{m}_a^{gt} \setminus \mathbf{m}_a^{(t-1)}) \cup (\mathbf{m}_a^{(t-1)} \setminus \mathbf{m}_a^{gt})\right). \tag{7}$$

Specifically, positive prompts are sampled from the false negative region $\mathbf{m}_a^{gt} \setminus \mathbf{m}_a^{(t-1)}$, representing areas missed by the current prediction. In contrast, negative prompts are drawn from the false positive region $\mathbf{m}_a^{(t-1)} \setminus \mathbf{m}_a^{gt}$, which contains over-assigned points. This iterative feedback loop ensures that the model develops the capacity to handle adjustments across various semantic granularities.

B Aggregating Point Labels to Part Mask

Based on the point-wise similarity calculated in Eq. 2, we assign a semantic label l_i for each point i via an argmax operation. To derive the optimized holistic segmentation $\mathcal{S}_{opt} = \{\mathbf{m}_i^{opt}\}_{i=1}^{K}$, we first denote the set of unique labels $\mathcal{L} = \{l^1, l^2, \ldots, l^K\}$. Each individual mask $\mathbf{m}_k^{opt}$ is then constructed by grouping points with the same label:

$$\mathbf{m}_{k,i}^{opt} = \mathbb{I}(l_i = l^k), \quad i \in 1, \ldots, N, \tag{8}$$

where $\mathbf{m}_{k,i}^{opt}$ denotes the i-th element of the binary mask $\mathbf{m}_k^{opt}$

C Limitations

In the data construction pipeline, we implement random part merging to simulate the interactive intents. While enhancing the robustness against diverse and arbitrary user prompts, it inherently introduces a trade-off by compromising the prior in semantic granularity. Consequently, for common semantic categories, the model may necessitate more interaction iterations to achieve the desired mask as shown in Fig. 5. To mitigate this, future work could integrate VLMs to adaptively filter out semantically implausible merges.

Another limitation lies in the evaluation protocol. All reported results are obtained with prompts simulated from ground-truth annotations, which makes the comparison reproducible but does not measure the interaction cost perceived by real users. A formal user study is left for future work.



Figure 5: The model exhibits ambiguous semantic granularity when provided with limited user prompts.

D More Training Details

Since aynamic data annotation construction alter the number of parts and hence renumber the labels, we keep track of a correspondence π during the construction, which maps each part index of $\mathcal{S}_{init}$ to the index of its counterpart in $\mathcal{S}_{gt}$. In the over-segmentation case, where a single ground-truth part is divided into two parts of $\mathcal{S}_{init}$, one of them is given an empty target, so that the model learns to reassign its points to the other.

The training process was conducted in two stages. The single mask interactive model in stage 1 was trained on 16 NVIDIA H20 GPUs for 10 epochs (approximately 5 days), while the semantic coordinator in stage 2 utilized 8 NVIDIA H20 GPUs for 5 epochs (approximately 2 days). We apply Concerto (Zhang et al., 2025) as the pre-trained weights for the PointTransformerV3 (Wu et al., 2024) encoder. The backbone produces 512-dimensional point-wise features, which are compressed to 256 dimensions for the Semantic Coordinator. We set the Dice weight λ_{dice} to 2, and both loss factors α and β to 1. The threshold of split probability τ is set to 0.5. We employ the Adam optimizer with a learning rate of 1e-5 and set the batch size to 1 per GPU. For each step, we iteratively sample 9 point prompt for interactive prediction. The complete set of configurations is listed in Table 4.

One detail concerns how part indices are encoded. Since instance labels are category-agnostic, the index of a part carries no semantic meaning, and the model should not associate a particular index with a particular kind of part. During training we therefore draw a fresh random permutation of indices for every sample and use it only when encoding $\mathcal{S}_{init}$, while the part queries are still pooled with the original labels. This keeps the correspondence between queries and parts intact and prevents the model from overfitting to any specific numbering.

E Ablation on Area Threshold τ_{area}

In the heuristic flood-fill baseline, unassigned or disconnected regions are either merged into neighboring components or designated as newly emerged parts based on the area threshold τ_{area}. As detailed in Table 5, the choice of τ_{area} substantially influences the final refinement quality.

Specifically, setting $\tau_{area} = 0.005$ yields the optimal performance for the baseline method (e.g., reaching an mIoU@7 of 62.3 and mIoU@10 of 62.9). Smaller values of τ_{area} improperly trigger the creation of redundant new part, whereas larger thresholds fail to effectively merge noisy over-segmented fragments, thereby degrading overall segmentation fidelity.

Table 5: Ablation study on the area threshold τ_{area} in the heuristic flood-fill baseline.

τ_{area}	**Init**	**mIoU@1**	**mIoU@3**	**mIoU@5**	**mIoU@7**	**mIoU@10**	**Gain@10**
0.003	47.6	50.7	54.9	56.7	57.1	57.7	10.1
0.005	**47.6**	**52.7**	**59.0**	**61.7**	**62.3**	**62.9**	**15.2**
0.010	47.6	46.5	49.1	49.9	50.2	50.4	2.72
0.020	47.6	43.3	44.8	45.2	45.4	45.5	-2.18
0.030	47.6	37.6	38.3	38.5	38.5	38.4	-9.23

F Inference Efficiency.

Benefiting from our lightweight network design, Partner enables real-time interaction. Table 6 summarizes the inference latency for each stage of our pipeline. Since feature extraction backbone only needs a single forward pass after the object is loaded, its latency is excluded from the single interaction time.

	Feature Extraction	First Stage	Second Stage	Single Interaction
Time	250 (ms)	20 (ms)	30 (ms)	50 (ms)

Table 6: Inference Latency.

Category	Item	Value
Backbone	Pre-trained weights	Concerto
	Output dimension	1728
	Projection MLP	$1728 \to 512 \to 512 \to 512$
	Feature dimension fd	512
Semantic Coordinator	Working dimension $ddim$	256
	Compression MLP	$512 \to 256 \to 256 \to 256$
	Two-Way Transformer blocks	2
	Final cross-attention	1
	Attention heads / head dimension	8 / 32
	Drop path ratio	0.0
Encoding	Point positional encoding	Fourier, 6 frequencies
	Prompt encoding	Fourier + positive/negative embedding
	Mask and label encoding	Sinusoidal scalar embedding + 2-layer MLP
	Holistic context fusion	Concatenation + MLP ($768 \to 512 \to 256$)
Query	Part query construction	Masked mean pooling
	Auxiliary queries	1 split query $\mathbf{q}_{split}$ + 1 decision token $\mathbf{q}_{dec}$
	Maximum number of parts	60
Split-Fill	Predictor	3-layer MLP ($256 \to 256 \to 256 \to 1$)
	Threshold τ	0.5
Loss	Mask loss $\mathcal{L}_{seg}$	Focal loss + λ_{dice}· Dice loss
	Dice weight λ_{dice}	2
	Mask term weight α	1
	Split term weight β	1
Training	Optimizer	Adam
	Learning rate	1e-5
	Batch size per GPU	1
	Sampled points per object	100,000
	Voxel grid size	0.01
	Maximum parts per training object	25
	Prompt iterations per step	9
	Boundary erosion probability	0.9
	Random merge probability	0.5
Evaluation	Prompt points per part	5
	Maximum refined parts	10
	Point-to-mesh mapping	5-NN majority voting on face centroids
	Flood-fill area thresholds	0.005

Table 4: Complete configurations of Partner.

G EXTENSIVE COMPARISONS ON AIGC ASSETS

We conducted extensive comparative experiments on assets generated by 3D AIGC Models as shown in Fig. 6. To simulate real-world usage, we employ a combination of textual descriptions and visual prompt demonstrations to represent user intent, instead of the GT-guided schema used in our quantitative analysis. Visual comparisons reveal that heuristic-based baselines struggle to refine specific boundaries. Their operations often lead to regions overlapping into unrelated components. In contrast, Partner accurately executes the user's intent within the target regions and prevents unexpected changes. It maintains the integrity of surrounding parts, ensuring the stability and consistency in the proceedings of interactive iteration.

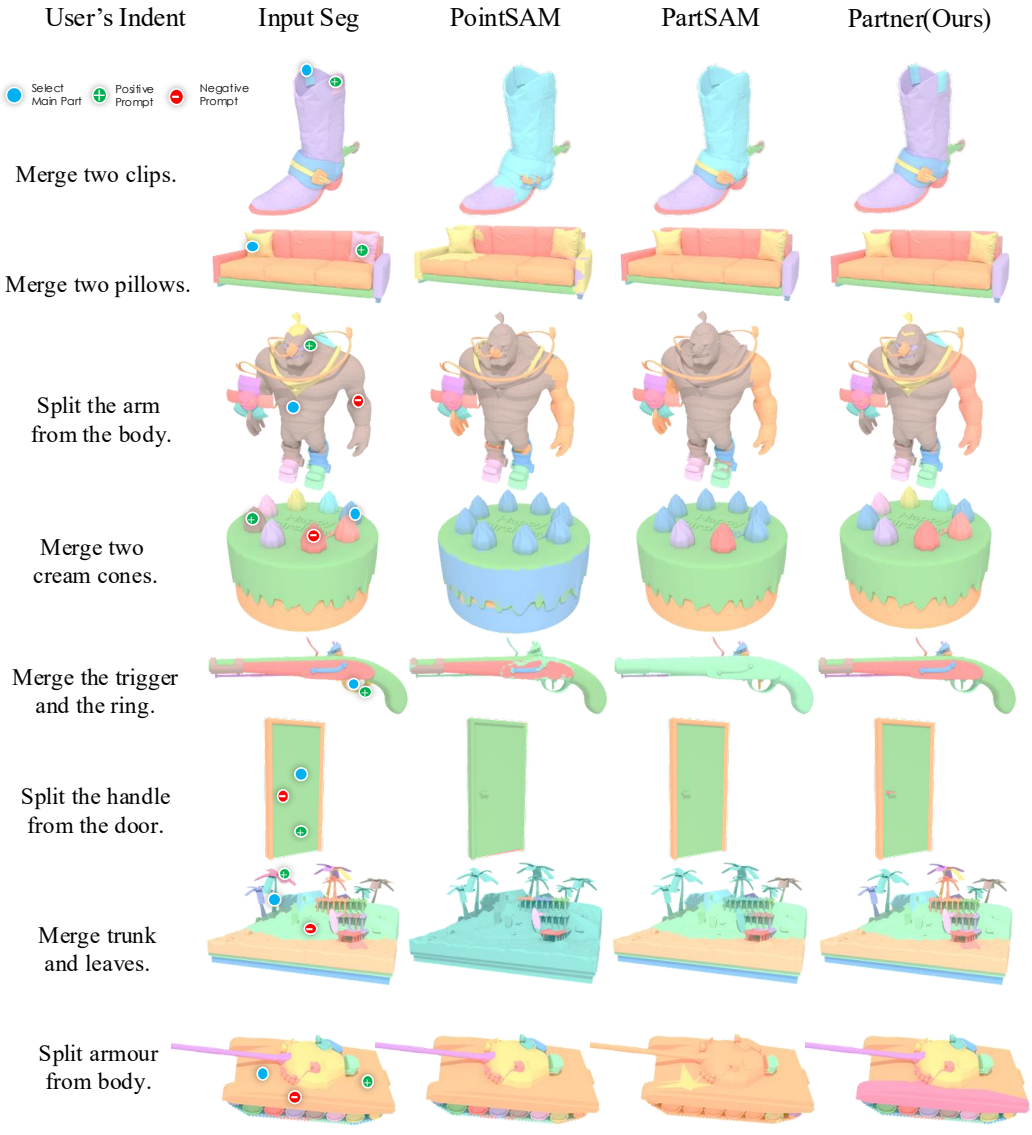


Figure 6: Extensive comparisons conducted on AIGC assets.