Open-Vocabulary 3D Part Segmentation with Semantic Propagation Hawkes Process

Kejie Shi[1], **Kun Zhou**[2], **Jieyu Zhao**[1], **Xulun Ye**[1,†]
[1] School of Artificial Intelligence, Ningbo University, Ningbo, China
[2] School of Architecture & Urban Planning, Shenzhen University, Shenzhen, China
2411100290@nbu.edu.cn,zhoukun@szu.edu.cn,{zhao_jieyu,yexulun}@nbu.edu.cn

Abstract

Existing open-vocabulary 3D part segmentation methods typically rely on point–text matching, where semantic organization is shaped implicitly by the training loss, and both training and inference lack an explicit forward semantic optimization mechanism. We address this limitation with a framework of *forward semantic dynamics*, implemented with *coupled semantic-dynamics blocks* that organize part semantics within a single forward pass. Specifically, differentiable forward low-rank semantic shaping constrains point features to a prompt-conditioned semantic subspace, suppressing semantic drift and improving intra-part consistency. Built on this shaped state, a finite-depth spatial Hawkes-style propagation models high-confidence prompt responses as sparse semantic events and propagates them across multiple steps to provide reliable evidence for ambiguous regions. The propagated context is further fed back into later feature shaping, coupling semantic organization and evidence propagation without test-time iterative optimization or additional post-processing. Extensive experiments on multiple open-vocabulary 3D part segmentation benchmarks show that our method establishes a new state of the art, outperforming the previous comparable SOTA method on all 16 reported evaluation slices by 1.88 to 5.31 mIoU points (3.84 on average) and achieving the overall best results on 14 of them. Project Page:https://github.com/weatheraim/Open-Vocabulary-3D-Part-Segmentation-with-Semantic-Propagation-Hawkes-Process.

1 Introduction

Open-vocabulary 3D part segmentation aims to assign fine-grained part labels to 3D points from free-form language prompts [1–9]. Unlike closed-set part segmentation and geometry-centric point understanding pipelines [10–13], this task requires more than recognizing local geometric patterns: it must organize stable part semantics under language guidance over the entire point cloud. This is particularly challenging in 3D part understanding, where repeated structures, sparse observations, and irregular sampling often separate points belonging to the same part, while locally similar regions may correspond to different semantic parts [1, 2, 6, 8, 9]. The ambiguity becomes even stronger under object-conditioned prompts such as *leg of a chair*, where local geometry alone is insufficient to determine the correct semantics [3, 6, 9]. The core problem of open-vocabulary 3D part segmentation is therefore not merely local point–text matching, but how to explicitly organize fine-grained part semantics during inference and provide reliable contextual evidence for locally ambiguous regions.

Existing open-vocabulary 3D part segmentation methods typically rely on one-shot point–text similarity matching [14, 2–9]. Related open-vocabulary 3D scene segmentation pipelines likewise rely heavily on one-shot text–region alignment [15–19]. Many of these methods are further built on CLIP-style or tri-modal representation learning pipelines [20–24]. They often enhance inter-point interactions via attention or context aggregation [10, 11, 25, 26, 12, 13]. However, these approaches

†Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

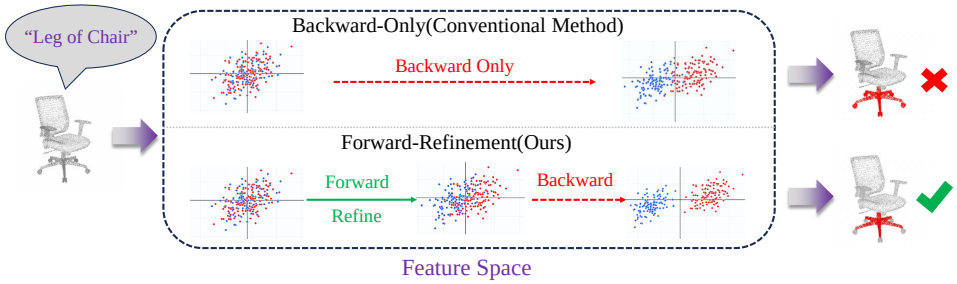


Figure 1: Comparison between conventional one-shot forward matching and the proposed forward semantic dynamics. Our model explicitly refines intermediate semantic states during forward propagation, yielding more reliable part organization under ambiguous prompts.

still suffer from two key limitations. First, high-dimensional 3D point features are prone to being contaminated by geometry-irrelevant perturbations and training noise, which amplify intra-part feature variance. Meanwhile, the final loss—optimized through backpropagation—can only constrain the outputs indirectly, making it difficult to explicitly preserve intra-part compactness during the forward pass (Fig. 1). Second, existing context aggregation mechanisms typically lack explicit modeling of point-wise semantic confidence, allowing low-confidence ambiguous points to participate in information propagation, which can lead to undirected context mixing and even semantic contamination.

To address these limitations, we propose a framework of *forward semantic dynamics*, implemented through stacked *coupled semantic-dynamics blocks*. Each block combines differentiable low-rank semantic shaping with a finite-depth spatial Hawkes-style propagation. Low-rank semantic shaping constrains feature updates to a prompt-conditioned low-dimensional semantic subspace, thereby suppressing semantically irrelevant variations, reducing intra-part variance, and stabilizing point–text alignment. Built on this shaped state, Hawkes-style propagation models high-confidence prompt responses as sparse semantic events and routes them in a confidence-aware, locality-constrained, and prompt-selective manner to provide reliable evidence for ambiguous regions. These two mechanisms are tightly coupled throughout the network: low-rank semantic shaping organizes a propagation-ready semantic state, while the propagated context in turn refines subsequent shaping. This coupling enables the model to perform semantic organization explicitly during the forward pass, jointly improving intra-part compactness, reliable evidence selection, and ambiguity correction, rather than relying solely on the final loss to implicitly induce such structure. Our distinction from generic context aggregation is therefore not simply the use of more inter-point communication, but the use of forward semantic control that first stabilizes prompt-relevant semantic states and then propagates only reliable semantic carriers. Empirically, our approach establishes a new state of the art on the aligned benchmark suite, outperforming the strongest broadly comparable baseline across all 16 evaluation slices by 1.88–5.31 mIoU points (3.84 on average), and obtaining the highest object mIoU on 14 of them, all without test-time iterative optimization or additional post-processing.
Our main contributions are as follows:

- We propose a forward semantic dynamics framework for open-vocabulary 3D part segmentation, reformulating point–text matching as explicit forward semantic control within a single pass, so that fine-grained part semantics can be directly organized during inference.
- We develop a coupled mechanism of differentiable low-rank semantic shaping and a finite-depth spatial Hawkes-style propagation. The former constrains point features to prompt-conditioned semantic subspaces to suppress semantic drift and improve intra-part consistency, while the latter models high-confidence prompt responses as sparse semantic events and propagates reliable evidence to ambiguous regions.
- We show that, without test-time iterative optimization or additional post-processing, our method establishes a new state of the art on the aligned benchmark suite, with especially strong gains in fine-grained, ambiguity-heavy, and object-conditioned settings.

2 Related Work

Open-vocabulary 3D segmentation and prompt-conditioned part understanding. Recent open-vocabulary 3D understanding is largely driven by vision-language alignment and 3D-language

pre-training. CLIP-style supervision [20] and its 3D extensions, including PointCLIP [21], ULIP [22], ULIP-2 [23], OpenShape [24], and CLIP2Scene [27], improve transferable 3D representations. Built on such features, open-vocabulary 3D scene segmentation methods perform text-guided dense prediction through point-level or region-level matching, as in OpenScene [15], PLA [16], OpenMask3D [17], Open3DIS [18], RegionPLC [28], foundation-model-based open-vocabulary 3D semantic segmentation [29], and Search3D [19]. Prompt-conditioned part understanding extends this line to fine-grained object parts, including PartGlot [1], PointCLIP V2 [14], PartSLIP [2], PartSLIP++ [3], PartDistill [4], ZeroPS [5], PARIS3D [7], Reasoning3D [6], Find3D [8], and CoSMo3D [9]. These methods improve 3D part grounding, but most still resolve inference through one-shot point-text matching, without explicitly regulating intermediate semantic states or separating reliable semantic carriers from ambiguous points during forward inference.

Low-rank structure in deep representation learning. Low-rank modeling has long been used for structured representation learning [30]. In deep networks, it is widely exploited for compression and acceleration [31–33], parameter-efficient adaptation [34, 35], and compact high-order representations such as low-rank bilinear pooling [36]. Related compactness-oriented objectives, including center loss [37] and supervised contrastive learning [38], also encourage more consistent feature geometry. Our setting differs from these uses: we employ low-rank structure not as a global factorization or efficiency trick, but as a blockwise forward-time semantic controller for dense prompt-conditioned prediction.

Hawkes processes and self-exciting dynamics in deep networks. Our propagation branch is motivated by event-history-dependent intensity modeling. Classical Hawkes processes model self-exciting and mutually exciting point processes [39, 40], while neural extensions include RMTPP [41], Neural Hawkes [42], Self-Attentive Hawkes Process [43], and Transformer Hawkes Process [44]. In parallel, dense prediction on point clouds commonly relies on graph propagation, non-local interaction, and neighborhood attention [10, 11, 25, 26, 12, 13]. These operators exchange context effectively, but typically act as generic feature mixing rather than as explicit accumulation of reliable semantic evidence. Our method instead converts high-confidence prompt responses into sparse semantic events and propagates them through a finite-depth, prompt-selective excitation process on a fixed spatial domain, rather than through generic all-point message passing. This separation between *carrier extraction* and *carrier-driven Hawkes-style propagation* is the main distinction between our formulation and prior point-cloud context operators, and it is also the distinction that the matched-control comparison in Sec. 4 is intended to test directly.

3 Method

3.1 Problem Formulation and Framework

Problem formulation. Given a point cloud $\mathcal{P} = \{p_i\}_{i=1}^{N}$, $p_i \in \mathbb{R}^3$, and an active prompt set $P \in \mathbb{R}^{K\times d}$, $P_k \in \mathbb{R}^d$, extracted from a frozen SigLIP text encoder [45], our goal is to assign each point i a prompt-conditioned semantic label from $\{1, \dots, K\}$ under either free-form part queries or object-conditioned part queries. We denote the initial point feature state by $X^{(0)} \in \mathbb{R}^{N\times d}$.

Framework. Our model consists of three stages as shown in Fig. 2: feature extraction, a semantic-dynamics stack with L blocks, and a final prompt-conditioned prediction, where the two semantic-dynamics modules are embedded into the deep network as lightweight intermediate components. Each semantic-dynamics block is defined as follows: the model receives the current feature state $X^{(l)}$, the prompt set P, point coordinates $\mathcal{P}$, and a propagated context summary $h^{(l-1)} \in \mathbb{R}^d$, with $h^{(-1)} = 0$. Throughout the paper, $\mathrm{Pool}(\cdot)$ denotes global average pooling. Each block performs the following computations:

$$\bar{X}^{(l)} = \mathcal{S}^{(l)}(X^{(l)}, P, h^{(l-1)}), \quad \lambda^{(l,0)} = \mathcal{I}^{(l)}(\bar{X}^{(l)}, P), \tag{1}$$

$$C^{(l)}, \lambda^{(l,1:T)} = \mathcal{H}^{(l)}(\lambda^{(l,0)}, P, \mathcal{P}), \quad X^{(l+1)} = \bar{X}^{(l)} + \gamma_l C^{(l)}, \quad h^{(l)} = \mathrm{Pool}(C^{(l)}). \tag{2}$$

Here $\lambda^{(l,1:T)} := \{\lambda^{(l,t)}\}_{t=1}^{T}$ denotes the propagated intensity sequence within block l. The operator $\mathcal{S}^{(l)}$ denotes the differentiable low-rank semantic shaping module (Sec. 3.2), $\mathcal{I}^{(l)}$ produces base prompt-conditioned intensities, and $\mathcal{H}^{(l)}$ denotes finite-depth spatial Hawkes-style propagation (Sec. 3.3), which performs carrier extraction, intensity updates, and feedback construction. Therefore,

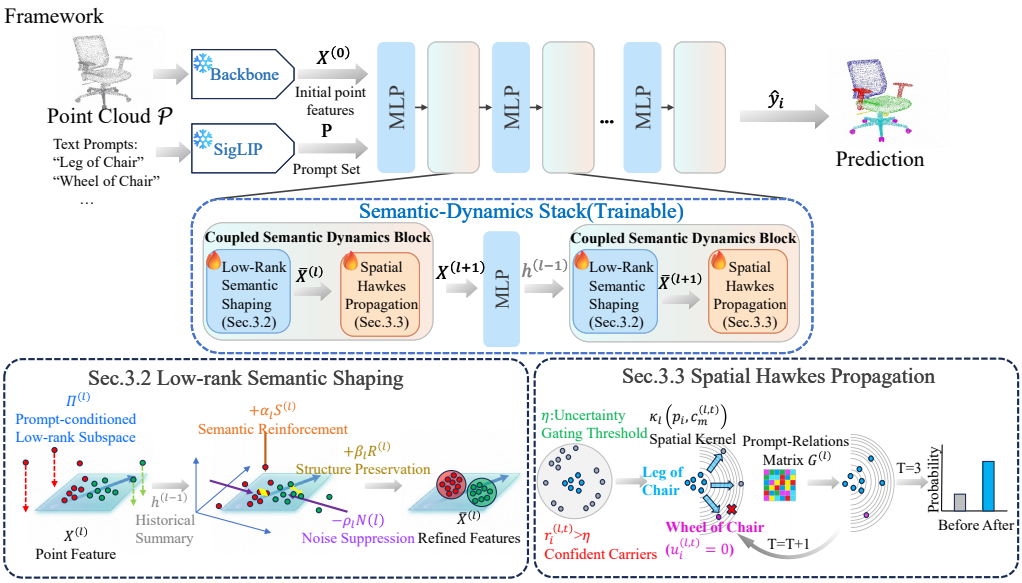


Figure 2: Overview of the proposed framework. Differentiable low-rank semantic shaping (Sec. 3.2) first restricts feature updates to a prompt-conditioned subspace. A finite-depth spatial Hawkes-style propagation (Sec. 3.3) then converts confident activations into semantic events and propagates reliable evidence, which is finally fed back to condition subsequent shaping steps

each block follows a fixed semantic-control sequence: semantic shaping, prompt scoring, carrier extraction, event-driven propagation, and feedback coupling. The overall framework is simply a stack of these blocks on top of frozen point features, making the model effectively a learned forward semantic controller rather than an iterative optimization routine. After L blocks and T propagation steps per block, the final prediction is $\hat{y}_i = \arg\max_k \lambda_{ik}^{(L,T)}$.

3.2 Low-Rank Semantic Shaping

The low-rank semantic shaping branch is responsible for *semantic formation*. Its role is to make same-part features more coherent before any evidence propagation happens, rather than leaving this organization to be induced only by the final loss. Given incoming features $X^{(l)} \in \mathbb{R}^{N\times d}$, we form a block context $z^{(l)} = [\mathrm{Pool}(X^{(l)}); \mathrm{Pool}(P); h^{(l-1)}]$. Two lightweight mappings then predict a prompt-conditioned low-rank basis and block gates:

$$B^{(l)} = \mathrm{Orth}(\psi_l(z^{(l)})), \quad \Pi^{(l)} = B^{(l)}B^{(l)\top}, \quad (\alpha_l, \rho_l, \beta_l, \gamma_l) = \mathrm{Sigmoid}(\delta_l(z^{(l)})). \tag{3}$$

where $B^{(l)} \in \mathbb{R}^{d\times r}$ with $r \ll d$, $\Pi^{(l)}$ is the corresponding rank-r projector, and $\mathrm{Orth}(\cdot)$ is a differentiable orthonormalization map.

Proposition 1 (Projector properties; proof in Appendix, Proposition A.1). If the columns of $B^{(l)}$ are orthonormal, then $\Pi^{(l)}$ is an orthogonal projector:

$$\Pi^{(l)2} = \Pi^{(l)}, \quad \Pi^{(l)\top} = \Pi^{(l)}. \tag{4}$$

This standard result justifies treating $\Pi^{(l)}$ as a valid prompt-conditioned projector in the decomposition and shaping update below. The basis is shared by all points for the current object-prompt instance, so shaping is blockwise and prompt-conditioned rather than pointwise adaptive. This shared basis is a deliberate tradeoff: it is coarser than fully pointwise adaptation, but it is more stable under sparse 3D sampling and more favorable to same-part consistency. Unlike a generic bottleneck adapter, the update in Eq. (6) explicitly separates subspace reinforcement, orthogonal suppression, and detail preservation. Appendix A provides the implementation details of $\mathrm{Orth}(\cdot)$ and the extended structural discussion.

We then decompose the centered features into an on-subspace component, an off-subspace residual, and a geometry-preserving residual:

$$\tilde{X}^{(l)} = X^{(l)} - \mathbf{1}\bar{x}^{(l)\top}, \quad S^{(l)} = \tilde{X}^{(l)}\Pi^{(l)}, \quad N^{(l)} = \tilde{X}^{(l)}(I - \Pi^{(l)}), \quad R^{(l)} = g_l(X^{(l)}), \tag{5}$$

where $\mathbf{1} \in \mathbb{R}^N$ is the all-ones vector, $I \in \mathbb{R}^{d\times d}$ is the identity matrix, $\bar{x}^{(l)} = \mathrm{Pool}(X^{(l)})$, and $g_l(\cdot)$ is a lightweight residual mapping used to preserve local structural detail. The shaped state is then

$$\bar{X}^{(l)} = X^{(l)} + \alpha_l S^{(l)} - \rho_l N^{(l)} + \beta_l R^{(l)}. \tag{6}$$

This update gives the low-rank branch a clear semantic role: $\alpha_l S^{(l)}$ reinforces prompt-consistent directions shared across the object, $-\rho_l N^{(l)}$ suppresses off-subspace drift, and $\beta_l R^{(l)}$ preserves local structural detail that should not be erased by semantic compression. For the structure-only part of the shaping update, define

$$Z^{(l)} = \tilde{X}^{(l)} + \alpha_l S^{(l)} - \rho_l N^{(l)}. \tag{7}$$

Proposition 2 (Rank constraint and residual contraction; proof in Appendix, Proposition A.2). For the construction above,

$$\mathrm{rank}(S^{(l)}) \le r, \tag{8}$$

$$Z^{(l)}(I - \Pi^{(l)}) = (1-\rho_l)N^{(l)}, \tag{9}$$

and, if $\rho_l \in [0,1]$,

$$\left\| Z^{(l)}(I-\Pi^{(l)}) \right\|_F \le \left\| N^{(l)} \right\|_F. \tag{10}$$

Proposition 2 formalizes the two structural claims used in the text: the reinforced semantic variation is confined to an at-most-r-dimensional prompt-conditioned subspace, and the off-subspace residual is contractively controlled by ρ_l. Appendix E further shows that this shaping leads to lower feature dispersion at the part level.

3.3 Semantic Propagation via Finite-Depth Spatial Hawkes-Style Dynamics

The low-rank semantic shaping branch stabilizes *what* the semantic state is; finite-depth spatial Hawkes-style propagation determines *how reliable evidence should spread*. Within each block, it takes the shaped state $\bar{X}^{(l)}$, derives pointwise prompt confidence, extracts reliable carriers, propagates event intensities for a fixed depth T, and stops after the T-th update. Starting from $\bar{X}^{(l)}$, we first compute base prompt intensities

$$\mu_{ik}^{(l)} = \lambda_{ik}^{(l,0)} = \tau_l \langle \bar{X}_i^{(l)}, P_k \rangle, \tag{11}$$

where $\tau_l > 0$ is a learned scale. These intensities measure local prompt compatibility, but do not yet distinguish reliable semantic responses from ambiguous ones.

At propagation step t, the current intensities induce prompt posteriors $\pi_{ik}^{(l,t)} = \exp(\lambda_{ik}^{(l,t)}) / \sum_{k'} \exp(\lambda_{ik'}^{(l,t)})$, winning prompts $q_i^{(l,t)} = \arg\max_k \pi_{ik}^{(l,t)}$, confidences $r_i^{(l,t)} = \max_k \pi_{ik}^{(l,t)}$, and uncertainties $u_i^{(l,t)} = 1 - r_i^{(l,t)}$. Thus, confidence is not predicted by a separate head; it is the maximum posterior implied by the current prompt intensities, while uncertainty is its complement. We keep only the confident subset $r_i^{(l,t)} \ge \eta_l$, where $\eta_l \in (0,1)$ is the block-specific confidence threshold, then build a local radius graph with support radius $r_h^{(l)}$ among points that share the same winning prompt, and treat each connected component $\mathcal{C}_m^{(l,t)}$ as one prompt-conditioned semantic event $e_m^{(l,t)} = (c_m^{(l,t)}, q_m^{(l,t)}, s_m^{(l,t)})$. In practice, $c_m^{(l,t)}$ is the coordinate centroid of $\mathcal{C}_m^{(l,t)}$, $q_m^{(l,t)}$ is its common prompt label, and $s_m^{(l,t)} = \sum_{i\in\mathcal{C}_m^{(l,t)}} r_i^{(l,t)}$ is its aggregated confidence strength. This event construction is the core of Hawkes-style propagation: it converts dense pointwise responses into a sparse set of reliable semantic carriers. Exact implementation details are provided in Appendix A.

To route those events, we define a nonnegative prompt-relation matrix and a locality-constrained spatial kernel,

$$G^{(l)} = \mathrm{softplus}(U^{(l)} V^{(l)\top}) \in \mathbb{R}_+^{K\times K}, \quad \kappa_l(p_i, c_m) = \exp\left(-\frac{\|p_i - c_m\|_2^2}{2\sigma_l^2}\right) \mathbf{1}[\|p_i - c_m\|_2 \le r_h^{(l)}], \tag{12}$$

where $U^{(l)}, V^{(l)} \in \mathbb{R}^{K\times d_g}$, d_g is the relation embedding dimension, $\sigma_l > 0$ is the kernel bandwidth, and $r_h^{(l)} > 0$ is the spatial support radius used both for carrier-graph construction and for the

propagation kernel. Propagation then follows

$$\lambda_{ik}^{(l,t+1)} = \lambda_{ik}^{(l,t)} + u_i^{(l,t)} \sum_{m=1}^{M_{l,t}} s_m^{(l,t)} \kappa_l(p_i, c_m^{(l,t)}) G^{(l)}_{k q_m^{(l,t)}}, \tag{13}$$

where $M_{l,t}$ is the number of extracted events and $G^{(l)}_{k\, q_m^{(l,t)}}$ measures how strongly an event with prompt label $q_m^{(l,t)}$ excites prompt k. At each step, every point therefore receives additive evidence from nearby carriers weighted by its own uncertainty, the carrier confidence strength, spatial proximity, and prompt relation. We use the Hawkes terminology in a precise but restricted sense: this module is a finite-depth spatial analogue of mutually exciting event accumulation rather than a classical stochastic point process. Its difference from ordinary sparse message passing lies in three coupled choices: the propagation source is a dynamically extracted event set rather than all points, the update is intensity-additive across propagation depth, and transfer is jointly modulated by uncertainty, locality, and prompt relations. In other words, the branch first decides *which* regions deserve to become carriers and only then decides *how* those carriers should excite later predictions. Since graph construction is restricted to the confident subset rather than the full point set, each propagation step operates on a sparse active graph instead of dense all-pairs interactions over all N points. The process is finite-depth rather than convergence-based: within each block it recomputes posteriors, carriers, and event strengths after every update, and then terminates exactly after T propagation steps; detailed efficiency statistics are reported in the appendix.

Proposition 3 (Monotone accumulation; proof in Appendix, Proposition A.3). For every block l, step t, point i, and prompt k,

$$\lambda_{ik}^{(l,t+1)} \geq \lambda_{ik}^{(l,t)}. \tag{14}$$

Proposition 4 (Prompt-selective relative sharpening; proof in Appendix, Proposition A.4). Even though the raw increments are nonnegative, the propagation remains discriminative. For any two prompts k_1 and k_2,

$$\left(\lambda_{ik_1}^{(l,t+1)} - \lambda_{ik_2}^{(l,t+1)}\right) - \left(\lambda_{ik_1}^{(l,t)} - \lambda_{ik_2}^{(l,t)}\right) = u_i^{(l,t)} \sum_{m=1}^{M_{l,t}} s_m^{(l,t)} \kappa_l(p_i, c_m^{(l,t)}) \left(G^{(l)}_{k_1 q_m^{(l,t)}} - G^{(l)}_{k_2 q_m^{(l,t)}}\right). \tag{15}$$

Proposition 4 shows that, although every increment is nonnegative, the *relative* preference between prompts is still selectively modulated by the prompt-relation matrix $G^{(l)}$. Therefore, propagation does not degenerate into indiscriminate score inflation. Appendix E further shows that the gains of this branch are concentrated on initially low-confidence points.

3.4 Feedback Coupling and End-to-End Training

The propagated intensities are not used only for the final prediction; they are returned to the feature stream so that later shaping is conditioned on accumulated evidence. From the terminal intensities $\lambda^{(l,T)}$, we compute

$$\pi_{ik}^{(l,T)} = \frac{\exp(\lambda_{ik}^{(l,T)})}{\sum_{k'=1}^{K} \exp(\lambda_{ik'}^{(l,T)})}, \quad C^{(l)} = \pi^{(l,T)} P W_h^{(l)}, \tag{16}$$

where $W_h^{(l)} \in \mathbb{R}^{d \times d}$. This feedback is injected through Eq. (2) and summarized as $h^{(l)} = \mathrm{Pool}(C^{(l)})$, which in turn conditions the next block context. The resulting loop is central to our design: low-rank shaping organizes a propagation-ready state, Hawkes-style propagation accumulates reliable evidence over that state, and the propagated context modifies the next shaping step.

The discrete routing identities induced by $\arg\max$, thresholding, and connected-component extraction are treated as stop-gradient decisions. Gradients do not backpropagate through the carrier assignment itself, but the trainable parameters are still optimized continuously through the base intensities, uncertainty weights, event strengths, and relation weights. This semi-discrete design remains stable early in training because the model can first improve the dense base intensities $\mu_{ik}^{(l)}$ before relying heavily on sparse event routing; when the state is still noisy, u_i remains high and the

propagation branch behaves conservatively rather than locking into brittle early carriers. Empirically, we did not observe routing collapse or unstable carrier extraction once the base prompt-conditioned state became moderately discriminative. The entire model is trained end-to-end using the final segmentation loss

$$p_{ik} = \frac{\exp(\lambda_{ik}^{(L,T)})}{\sum_{k'=1}^{K} \exp(\lambda_{ik'}^{(L,T)})}, \quad \mathcal{L} = -\frac{1}{|\Omega|} \sum_{i \in \Omega} \log p_{iy_i}, \tag{17}$$

where Ω is the supervised point set and $y_i \in \{1, \ldots, K\}$ is the ground-truth prompt-conditioned label of point i. We do not introduce a separate compactness loss or point-process likelihood; both branches are learned jointly through the final dense prediction objective.

Table 1: Main benchmark comparison (object mIoU, %, higher is better). Part denotes free-form part prompts, and Obj-Part denotes object-conditioned prompts of the form *part of object*. Find3D reports the original results; Find3D* is its aligned-suite reproduction. PartSLIP++† denotes the fine-tuned canonical-view PartNetE setting. Best and second-best results are in bold and underlined.

I: 3DCompat200

Method	Venue	3DCompat200-Coarse				3DCompat200-Fine			
		Canonical		Rotated		Canonical		Rotated	
		Part	Obj-Part	Part	Obj-Part	Part	Obj-Part	Part	Obj-Part
PointCLIPV2	ICCV'23	14.16	14.09	13.39	13.17	7.18	7.25	7.20	7.43
PartSLIP	CVPR'23	6.54	1.27	8.03	1.74	1.56	0.23	4.44	0.71
PartSLIP++	arXiv'23	17.62	6.12	15.82	5.89	5.68	3.79	4.76	3.41
Find3D	ICCV'25	25.36	31.72	26.23	32.47	12.57	12.69	12.79	12.73
Find3D*	ICCV'25	45.00	36.89	46.02	38.14	24.97	17.72	26.35	18.57
CoSMo3D	CVPR'26	<u>54.52</u>	<u>47.51</u>	<u>54.55</u>	<u>47.74</u>	<u>31.29</u>	<u>24.48</u>	<u>30.97</u>	<u>24.46</u>
Ours	–	**57.94**	**52.82**	**57.69**	**52.65**	**35.90**	**29.63**	**35.35**	**29.16**

II: ShapeNetPart and PartNetE

Method	Venue	ShapeNetPart				PartNetE			
		Canonical		Rotated		Canonical		Rotated	
		Part	Obj-Part	Part	Obj-Part	Part	Obj-Part	Part	Obj-Part
OpenMask3D	NeurIPS'23	10.37	8.94	14.56	6.75	11.24	12.54	11.67	11.93
PartSLIP++†	arXiv'23	6.46	1.43	6.03	0.94	**32.71**	5.12	**23.03**	3.87
PointCLIPV2	ICCV'23	20.22	16.91	18.19	16.88	9.70	11.28	10.22	10.32
Find3D	ICCV'25	24.09	28.39	23.71	29.64	16.38	16.86	17.16	17.62
CoSMo3D	CVPR'26	<u>33.31</u>	<u>36.16</u>	<u>32.84</u>	<u>34.20</u>	17.19	<u>17.59</u>	18.17	<u>18.48</u>
Ours	–	**36.13**	**38.04**	**34.86**	**37.98**	<u>20.28</u>	**21.83**	<u>21.61</u>	**23.00**

4 Experiments

4.1 Experimental Setup

We evaluate the proposed semantic-dynamics framework on the aligned comparison suite released with 3DCompat200 and adopted by CoSMo3D [46, 9], using four benchmarks that stress complementary aspects of open-vocabulary 3D part segmentation: 3DCompat200-Coarse and 3DCompat200-Fine [46], ShapeNetPart [47], and PartNetE [2, 48]. For each benchmark, we report object mIoU (%, higher is better) under two prompt regimes: *Part* for free-form part prompts and *Obj-Part* for object-conditioned part prompts. We compare our method against PointCLIPV2 [14], OpenMask3D [17], PartSLIP [2], PartSLIP++[3], Find3D [8], and CoSMo3D [9].

All prompt templates, splits, and canonical/rotated settings follow the released aligned protocol [46, 9]. Following CoSMo3D [9], we train only on the 3DCompat200 aligned training split and evaluate on its test split and on the unseen ShapeNetPart and PartNetE datasets. We use the same frozen geometric feature extractor and SigLIP text encoder as CoSMo3D, training only the added semantic-dynamics heads. PointCLIPV2, OpenMask3D, PartSLIP, Find3D*, and CoSMo3D use aligned-suite reruns or reproductions; the original Find3D results are also shown. PartSLIP++† is separately marked for its fine-tuned canonical-view PartNetE setting. The aligned comparisons use the same prompts, splits, views, and object-level mIoU; the daggered result is kept separate. Our method uses no test-time iterative optimization or post-processing. Matched controls use the

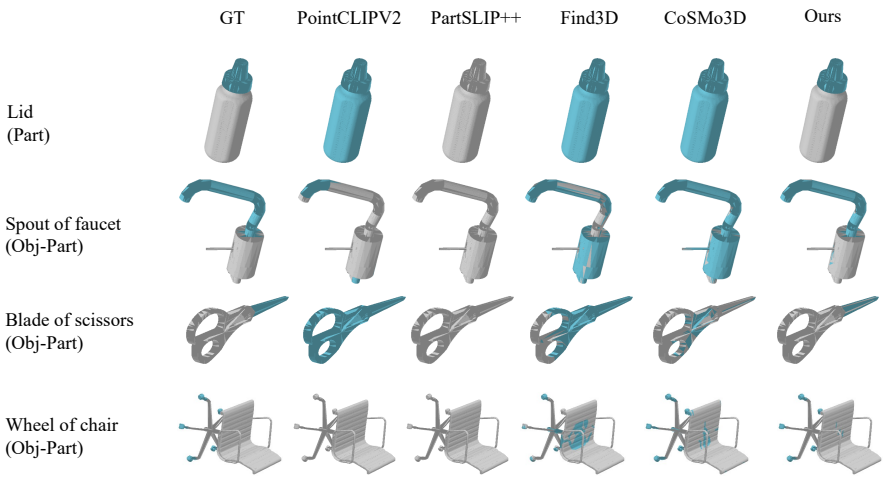


Figure 3: Qualitative comparison. Our model produces more complete and semantically consistent predictions, especially on repeated structures and locally ambiguous regions.

same block locations and frozen-backbone protocol; 'Extra Params' counts only new trainable heads beyond the frozen encoders. Further details are in Appendix A–E.

4.2 Main Results

Consistent gains are observed across the aligned benchmark suite (Table 1). Relative to the strongest broadly comparable baseline, CoSMo3D, our method improves all 16 evaluation slices and achieves the best overall result on 14 of them. The largest gains appear on 3DCompat200, especially on Fine and Obj-Part settings, where ambiguity and object-conditioned reasoning are most pronounced. On 3DCompat200-Fine, the gains over CoSMo3D reach $+4.61$ and $+5.15$ mIoU in the canonical Part and Obj-Part settings, and $+4.38$ and $+4.70$ under rotation. These trends match the intended role split of the semantic-dynamics framework. The low-rank semantic shaping branch improves semantic formation broadly by reducing off-subspace drift, whereas Hawkes-style propagation is most valuable when local prompt compatibility is insufficient and reliable evidence must be routed across space. Gains remain positive across all four benchmarks and both prompt regimes, suggesting that the method is not over-specialized to a single dataset or prompt format. The only two non-best slices are the canonical Part and rotated Part settings of PartNetE, where PartSLIP++† remains higher under its specialized fine-tuned canonical-view protocol. Because the geometric encoder is fixed, the improvements are more naturally attributed to the semantic-dynamics stack, i.e., low-rank semantic shaping and Hawkes-style propagation, than to stronger geometric encoding.

The benchmark-wise margins further clarify this transfer pattern. Relative to CoSMo3D, our gains on 3DCompat200-Coarse range from $+3.14$ to $+5.31$ mIoU across the four prompt/view slices, while the gains on 3DCompat200-Fine range from $+4.38$ to $+5.15$. On the evaluation-only datasets, the corresponding gains range from $+1.88$ to $+3.78$ on ShapeNetPart and from $+3.09$ to $+4.52$ on PartNetE. Thus, the improvements remain positive across the training benchmark and both datasets held out for evaluation, including rotated views. Since neither target dataset is used to train or adapt our model, these results test transfer of the learned semantic-dynamics heads with the frozen feature pipeline. Representative qualitative examples further support the quantitative pattern (Fig. 3). Predictions from our framework are generally more complete and semantically consistent.

4.3 Ablation Studies

Core component ablations. Core ablations clarify the role split between the two branches (Table 3). Removing low-rank semantic shaping causes the largest degradation ($36.57 \rightarrow 33.09$ in All Avg.), identifying it as the main stabilizer of semantic feature formation. Removing Hawkes-style propagation also lowers performance ($36.57 \rightarrow 35.28$), with a larger drop on object-conditioned slices than on part-level slices, consistent with its ambiguity-resolution role. Increasing the propagation depth from $T{=}1$ to $T{=}3$ yields monotonic gains, and removing Hawkes-to-feature feedback reduces All Avg. to 36.24, confirming that propagated evidence is most useful when it is coupled back into later

semantic shaping rather than used only for the final prediction. Appendix E extends these results with feature-dispersion and confidence-stratified analyses.

We additionally reran the core ablations with five independent random seeds, varying only the initialization of the trainable semantic-dynamics heads while keeping the frozen encoders, data split, and evaluation protocol fixed. The resulting All Avg. means and standard deviations are reported in Table 2. The main component effects are substantially larger than the observed seed variation: removing low-rank shaping and propagation reduces the mean by 3.48 and 1.29 points, respectively. The feedback and depth effects are smaller but consistent across paired runs, with gains of 0.33 ± 0.07 and 0.47 ± 0.04, respectively. Over the 16 Table 1 slices, the five-seed mean of the full model is 36.54 ± 0.09, compared with 32.72 for CoSMo3D. The mean paired slice gain is $+3.83$ mIoU, with slice gains from $+1.86$ to $+5.29$. Thus, the strongest conclusions concern low-rank shaping and event-driven propagation, while feedback and depth provide supporting, accuracy–cost-sensitive improvements. Across the 16 individual Table 1 evaluation slices, the standard deviation across the five seeds ranges from 0.12 to 0.19. The value 36.54 ± 0.09 is the mean and standard deviation of the per-seed average over those 16 slices, rather than the standard deviation of an individual slice.

Table 2: Five-seed robustness of the core ablations. Only the initialization of the trainable semantic-dynamics heads changes; all frozen encoders, splits, and evaluation settings are fixed. All Avg. denotes object mIoU (%) averaged over the canonical evaluation slices used in Table 3; entries report the mean and sample standard deviation over five independent initializations.

Model variant	Mean (↑)	Std. (↓)	Drop vs. full (↓)
Full model, $T = 3$	36.57	0.09	–
w/o low-rank shaping	33.09	0.10	3.48
w/o propagation	35.28	0.13	1.29
w/o feedback	36.24	0.12	0.33
$T = 1$ propagation	35.73	0.12	0.84
$T = 2$ propagation	36.10	0.10	0.47

Table 3: Core ablations of the coupled semantic-dynamics framework. All values are object mIoU (%) averaged over the canonical evaluation slices.

Variant	Low-rank	Propagation	Feedback	Steps	Part Avg.	Obj. Avg.	All Avg.
w/o propagation	✓			–	36.61	33.95	35.28
w/o low-rank shaping		✓	✓	3	34.22	31.96	33.09
w/o feedback	✓	✓		3	37.29	35.19	36.24
1-step propagation	✓	✓	✓	1	36.90	34.56	35.73
2-step propagation	✓	✓	✓	2	37.18	35.01	36.10
Full model	✓	✓	✓	3	**37.56**	**35.58**	**36.57**

Table 4: Matched-control and parameter-overhead summary. Generic controls are inserted at the same block locations and trained under the same protocol as the full model. 'Extra Params' counts only the added trainable heads beyond the frozen CoSMo3D backbone and frozen SigLIP text encoder.

Variant	3DCompat200-Fine Obj-Part (Canon.)	All Avg.	Extra Params
Full model	29.63	36.57	+0.82M
Replace low-rank with bottleneck adapter	28.23	35.22	+0.79M
Replace Hawkes with sparse GAT propagation	28.74	35.79	+0.88M

Matched controls and parameter summary. Matched controls address the alternative explanation that the gains arise from adding any lightweight refinement module or any sparse propagation operator (Table 4). Replacing low-rank semantic shaping with a parameter-matched bottleneck adapter lowers performance from 29.63 to 28.23 on the representative 3DCompat200-Fine Obj-Part slice and from 36.57 to 35.22 in All Avg., despite nearly identical parameter overhead. Replacing Hawkes-style propagation with sparse GAT propagation also underperforms the full model (28.74 on the representative slice and 35.79 in All Avg.) while using a comparable number of added parameters. The comparison is therefore not explained by a larger learnable budget: the specific low-rank semantic shaping form and the carrier-driven Hawkes-style propagation mechanism still matter under matched insertion locations and frozen backbone features.

To separate the event-driven design from generic propagation, we further compare propagation operators under the same insertion locations, frozen-backbone protocol, prompts, splits, and metric

(Table 5). Neighbor-based controls propagate features or logits from graph nodes, whereas our module first aggregates confident points into connected semantic events and then updates prompt intensities using target uncertainty, carrier strength, locality, and prompt relations. The confidence-gated message-passing control is stronger than ungated generic propagation, but remains below the full event-driven formulation.

Table 5: Propagation-operator controls under the same aligned protocol. Only the propagation operator changes; the low-rank shaping branch, insertion locations, frozen encoders, and training/evaluation settings are fixed. All Avg. includes mean±sample standard deviation where five-seed results are available; diffusion-style and confidence-gated message passing are single-run means.

Propagation operator	Source	Updated quantity	3DCompat200-Fine Obj-Part (↑)	All Avg. (↑)
w/o propagation	None	Local prompt intensities	27.84	35.28 ± 0.13
Sparse kNN	Neighbor points	Features/logits	28.46	35.47 ± 0.12
Diffusion-style	Neighbor points	Logits	28.52	35.56
Sparse GAT	Graph nodes	Features	28.74	35.79 ± 0.13
Confidence-gated MP	Confident points	Features/logits	28.88	35.91
Ours	Connected events	Prompt intensities	**29.63**	$\mathbf{36.57 \pm 0.09}$

Among the generic propagation controls, confidence-gated message passing is the strongest alternative, reaching 35.91 All Avg. and 28.88 on the representative Fine Obj-Part slice. The full model is higher by 0.66 and 0.75 points, respectively; compared with sparse GAT, the margins are 0.78 and 0.89 points. These controlled comparisons indicate that confidence filtering alone does not account for the full result: event construction and prompt-intensity updates also contribute. Diffusion-style and confidence-gated results are single-run means, so these margins are descriptive comparisons rather than estimates of seed-level significance.

Hyperparameter Sensitivity Analysis.

The sensitivity analysis reveals that the model's performance remains stable across moderate ranges of the key hyperparameters: the low-rank dimension r, propagation depth T, confidence threshold η, and spatial kernel radius r_h (Fig. 4). Performance saturates once the shaping subspace is sufficiently expressive ($r \approx 32$). Increasing propagation depth beyond one step gives diminishing returns; the extended sweep in Appendix D shows that $T = 3$ is a practical accuracy–cost default rather than a unique optimum. Intermediate routing settings balance selective evidence transfer and spatial continuity, while extreme values either admit noisy carriers or remove useful evidence.

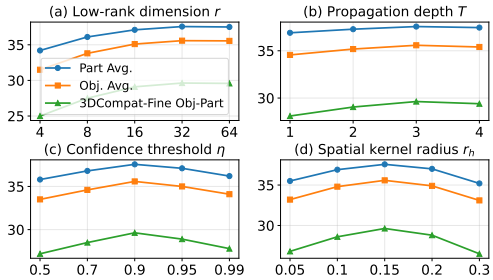


Figure 4: Sensitivity to r, T, η, and r_h.

The extended depth sweep makes this trade-off explicit. Increasing T from 3 to 4 and 5 raises All Avg. by only 0.04 and 0.05 points, while relative inference cost rises to $1.08\times$ and $1.16\times$, respectively. At $T = 6$, the mean returns to 36.57; at $T = 8$ and 10, it falls to 36.37 and 36.16 as cost continues to grow. These results support $T = 3$ as a practical default on the measured accuracy–cost curve.

5 Conclusion

We present a semantic-dynamics framework for open-vocabulary 3D part segmentation. Each coupled semantic-dynamics block combines low-rank semantic shaping with a finite-depth spatial Hawkes-style propagation, allowing the model to stabilize prompt-relevant semantic states and route reliable evidence to ambiguous regions within the forward pass. Empirically, this design yields the strongest gains in fine-grained and object-conditioned settings, where one-shot local matching is least reliable.

Limitations. More broadly, our method still faces a structural gap between continuous semantics and discrete, sparse point clouds: language is compositional and smooth, while 3D data are irregular, incomplete, and locally ambiguous. An incorrect high-confidence carrier can locally reinforce a wrong label, especially near boundaries, for overlapping Obj-Part prompts, or in sparse/noisy regions. Prompt wording and the carrier threshold also introduce sensitivity, and finite-depth propagation adds inference cost. Bridging open-vocabulary semantics with discrete geometry remains an open challenge for future research.

Acknowledgments and Disclosures

Funding. This work was supported by the National Natural Science Foundation of China under Grant 62606327, 62471266, 62006131, 62071260; the National Natural Science Foundation of Zhejiang Province under Grant LQ24F020001; and the Ningbo Major Research and Development Plan Project (No.2023Z224).

Competing interests. The authors declare no competing interests.

References

[1] Juil Koo, Ian Huang, Panos Achlioptas, Leonidas J. Guibas, and Minhyuk Sung. PartGlot: Learning shape part segmentation from language reference games. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16505–16514, 2022.

[2] 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 (CVPR)*, pages 21736–21746, 2023.

[3] 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.

[4] Ardian Umam, Cheng-Kun Yang, Min-Hung Chen, Jen-Hui Chuang, and Yen-Yu Lin. PartDistill: 3D shape part segmentation by vision-language model distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3470–3479, 2024.

[5] Yuheng Xue, Nenglun Chen, Jun Liu, and Wenyun Sun. ZeroPS: High-quality cross-modal knowledge transfer for zero-shot 3D part segmentation. In *International Conference on 3D Vision (3DV)*, pages 1328–1339, 2025.

[6] Tianrun Chen, Chunan Yu, Jing Li, Jianqi Zhang, Lanyun Zhu, Deyi Ji, Yong Zhang, Ying Zang, Zejian Li, and Lingyun Sun. Reasoning3D – grounding and reasoning in 3D: Fine-grained zero-shot open-vocabulary 3D reasoning part segmentation via large vision-language models. *arXiv preprint arXiv:2405.19326*, 2024.

[7] Amrin Kareem, Jean Lahoud, and Hisham Cholakkal. PARIS3D: Reasoning-based 3D part segmentation using large multimodal models. In *European Conference on Computer Vision (ECCV)*, pages 466–482, 2024.

[8] Ziqi Ma, Yisong Yue, and Georgia Gkioxari. Find any part in 3D. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7818–7827, 2025.

[9] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

[10] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems*, 2017.

[11] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic Graph CNN for learning on point clouds. *ACM Transactions on Graphics*, 38(5):146:1–146:12, 2019.

[12] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip H. S. Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16259–16268, 2021.

[13] Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. Point transformer V2: Grouped vector attention and partition-based pooling. In *Advances in Neural Information Processing Systems*, 2022.

[14] Xiangyang Zhu, Renrui Zhang, Bowei He, Ziyu Guo, Ziyao Zeng, Zipeng Qin, Shanghang Zhang, and Peng Gao. PointCLIP V2: Prompting CLIP and GPT for powerful 3D open-world learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2639–2650, 2023.

[15] Songyou Peng, Kyle Genova, Chiyu Max Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. OpenScene: 3D scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–824, 2023.

[16] Runyu Ding, Jihan Yang, Chuhui Xue, Wenqing Zhang, Song Bai, and Xiaojuan Qi. PLA: Language-driven open-vocabulary 3D scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7010–7019, 2023.

[17] Ayca Takmaz, Elisabetta Fedele, Robert W. Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. OpenMask3D: Open-vocabulary 3D instance segmentation. In *Advances in Neural Information Processing Systems*, 2023.

[18] Phuc D. A. 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 (CVPR)*, pages 4018–4028, 2024.

[19] Ayca Takmaz, Alexandros Delitzas, Robert W. Sumner, Francis Engelmann, Johanna Wald, and Federico Tombari. Search3D: Hierarchical open-vocabulary 3D segmentation. *arXiv preprint arXiv:2409.18431*, 2024.

[20] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763, 2021.

[21] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. PointCLIP: Point cloud understanding by CLIP. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[22] Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. ULIP: Learning a unified representation of language, images, and point clouds for 3D understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[23] Le Xue, Ning Yu, Shu Zhang, Artemis Panagopoulou, Junnan Li, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. ULIP-2: Towards scalable multimodal pre-training for 3D understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[24] Minghua Liu, Ruoxi Shi, Kaiming Kuang, Yinhao Zhu, Xuanlin Li, Shizhong Han, Hong Cai, Fatih Porikli, and Hao Su. OpenShape: Scaling up 3D shape representation towards open-world understanding. In *Advances in Neural Information Processing Systems*, 2023.

[25] Lei Wang, Yuchun Huang, Yaolin Hou, Shenman Zhang, and Jie Shan. Graph attention convolution for point cloud semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10296–10305, 2019.

[26] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7794–7803, 2018.

[27] 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 (CVPR)*, 2023.

[28] Jihan Yang, Runyu Ding, Weipeng Deng, Zhe Wang, and Xiaojuan Qi. RegionPLC: Regional point-language contrastive learning for open-world 3D scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19823–19832, 2024.

[29] Li Jiang, Shaoshuai Shi, and Bernt Schiele. Open-vocabulary 3D semantic segmentation with foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21284–21294, 2024.

[30] Guangcan Liu, Zhouchen Lin, Shuicheng Yan, Ju Sun, Yong Yu, and Yi Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):171–184, 2013.

[31] Emily L. Denton, Wojciech Zaremba, Joan Bruna, Yann LeCun, and Rob Fergus. Exploiting linear structure within convolutional networks for efficient evaluation. In *Advances in Neural Information Processing Systems*, 2014.

[32] Max Jaderberg, Andrea Vedaldi, and Andrew Zisserman. Speeding up convolutional neural networks with low rank expansions. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2014.

[33] Cheng Tai, Tong Xiao, Yi Zhang, Xiaogang Wang, and Weinan E. Convolutional neural networks with low-rank regularization. In *International Conference on Learning Representations (ICLR)*, 2016.

[34] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

[35] Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. AdaLoRA: Adaptive budget allocation for parameter-efficient fine-tuning. In *International Conference on Learning Representations (ICLR)*, 2023.

[36] Shu Kong and Charless Fowlkes. Low-rank bilinear pooling for fine-grained classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 365–374, 2017.

[37] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision (ECCV)*, pages 499–515, 2016.

[38] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, 2020.

[39] Alan G. Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1):83–90, 1971.

[40] Yosihiko Ogata. Statistical models for earthquake occurrences and residual analysis for point processes. *Journal of the American Statistical Association*, 83(401):9–27, 1988.

[41] Nan Du, Hanjun Dai, Rakshit Trivedi, Utkarsh Upadhyay, Manuel Gomez-Rodriguez, and Le Song. Recurrent marked temporal point processes: Embedding event history to vector. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1555–1564, 2016.

[42] Hongyuan Mei and Jason M. Eisner. The neural hawkes process: A neurally self-modulating multivariate point process. In *Advances in Neural Information Processing Systems*, 2017.

[43] Qiang Zhang, Aldo Lipani, Omer Kirnap, and Emine Yilmaz. Self-attentive hawkes process. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 2020.

[44] Simiao Zuo, Haoming Jiang, Zichong Li, Tuo Zhao, and Hongyuan Zha. Transformer hawkes process. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 2020.

[45] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11975–11986, 2023.

[46] Mahmoud Ahmed, Xiang Li, Arpit Prajapati, and Mohamed Elhoseiny. 3DCoMPaT200: Language grounded large-scale 3D vision dataset for compositional recognition. In *Advances in Neural Information Processing Systems*, 2024. Datasets and Benchmarks Track.

[47] Li Yi, Vladimir G. Kim, Duygu Ceylan, I-Chao Shen, Mengyan Yan, Hao Su, Cewu Lu, Qixing Huang, Alla Sheffer, and Leonidas Guibas. A scalable active framework for region annotation in 3D shape collections. *ACM Transactions on Graphics*, 35(6):210:1–210:12, 2016.

[48] 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 (CVPR)*, pages 909–918, 2019.

Appendix

A Additional Method Details and Structural Derivations

Unless otherwise stated, notation follows the main text.

A.1 Differentiable Orthonormalization and Projector Construction

The main text defines the prompt-conditioned low-rank basis through

$$B^{(l)} = \mathrm{Orth}\big(\psi_l(z^{(l)})\big), \qquad \Pi^{(l)} = B^{(l)}B^{(l)\top}, \tag{18}$$

where $B^{(l)} \in \mathbb{R}^{d\times r}$ and $r \ll d$. A standard differentiable realization of $\mathrm{Orth}(\cdot)$ is thin QR orthonormalization applied to the raw basis proposal $\psi_l(z^{(l)})$, followed by retaining the first r orthonormal columns. The derivations below require only that the resulting columns are orthonormal.

Proposition A.1 (Projector properties). If the columns of $B^{(l)}$ are orthonormal, then $\Pi^{(l)}$ is an orthogonal projector:

$$\Pi^{(l)2} = \Pi^{(l)}, \qquad \Pi^{(l)\top} = \Pi^{(l)}. \tag{19}$$

Proof. Since $B^{(l)\top}B^{(l)} = I_r$,

$$\Pi^{(l)2} = B^{(l)}B^{(l)\top}B^{(l)}B^{(l)\top} = B^{(l)}I_rB^{(l)\top} = \Pi^{(l)}. \tag{20}$$

Symmetry follows from

$$\Pi^{(l)\top} = \left(B^{(l)}B^{(l)\top}\right)^\top = B^{(l)}B^{(l)\top}. \tag{21}$$

A.2 Low-Rank Semantic Shaping: Structural Derivations

This subsection corresponds directly to Eq. (6) in the main text. For convenience, we restate the key quantities:

$$\tilde{X}^{(l)} = X^{(l)} - \mathbf{1}\,\bar{x}^{(l)\top}, \qquad S^{(l)} = \tilde{X}^{(l)}\Pi^{(l)}, \qquad N^{(l)} = \tilde{X}^{(l)}(I - \Pi^{(l)}), \tag{22}$$

and

$$\bar{X}^{(l)} = X^{(l)} + \alpha_l S^{(l)} - \rho_l N^{(l)} + \beta_l R^{(l)}. \tag{23}$$

Define the structure-only part of the update by

$$Z^{(l)} = \tilde{X}^{(l)} + \alpha_l S^{(l)} - \rho_l N^{(l)}. \tag{24}$$

Proposition A.2 (Rank constraint and residual contraction). For the construction above,

$$\mathrm{rank}(S^{(l)}) \le r, \tag{25}$$

$$Z^{(l)}(I - \Pi^{(l)}) = (1 - \rho_l)N^{(l)}, \tag{26}$$

and, if $\rho_l \in [0, 1]$,

$$\left\|Z^{(l)}(I - \Pi^{(l)})\right\|_F \le \left\|N^{(l)}\right\|_F. \tag{27}$$

Proof. For Eq. (25),

$$\mathrm{rank}(S^{(l)}) = \mathrm{rank}(\tilde{X}^{(l)}\Pi^{(l)}) \le \min\big(\mathrm{rank}(\tilde{X}^{(l)}), \mathrm{rank}(\Pi^{(l)})\big) \le r. \tag{28}$$

For Eq. (26),

$$Z^{(l)}(I - \Pi^{(l)}) = \tilde{X}^{(l)}(I - \Pi^{(l)}) + \alpha_l S^{(l)}(I - \Pi^{(l)}) - \rho_l N^{(l)}(I - \Pi^{(l)}) \tag{29}$$

$$= N^{(l)} + \alpha_l \tilde{X}^{(l)}\Pi^{(l)}(I - \Pi^{(l)}) - \rho_l \tilde{X}^{(l)}(I - \Pi^{(l)})^2. \tag{30}$$

Since $\Pi^{(l)}(I - \Pi^{(l)}) = 0$ and $(I - \Pi^{(l)})^2 = (I - \Pi^{(l)})$,

$$Z^{(l)}(I - \Pi^{(l)}) = N^{(l)} - \rho_l N^{(l)} = (1 - \rho_l)N^{(l)}. \tag{31}$$

Equation (27) then follows from

$$\left\| Z^{(l)}(I - \Pi^{(l)}) \right\|_F = |1 - \rho_l| \, \|N^{(l)}\|_F \leq \|N^{(l)}\|_F \tag{32}$$

whenever $\rho_l \in [0, 1]$.

Interpretation. These identities justify the structural role assigned to low-rank semantic shaping in the main text. The update explicitly confines reinforced variation to an at-most-r-dimensional prompt-conditioned subspace while preventing growth of the orthogonal residual. The claim that this improves semantic compactness is empirical and is evaluated in Appendix E.

A.3 Finite-Depth Spatial Hawkes-Style Propagation: Structural Interpretation

This subsection corresponds to Eq. (13) in the main text. The propagation update is

$$\lambda_{ik}^{(l,t+1)} = \lambda_{ik}^{(l,t)} + u_i^{(l,t)} \sum_{m=1}^{M_{l,t}} s_m^{(l,t)} \kappa_l\big(p_i, c_m^{(l,t)}\big) G^{(l)}_{k\, q_m^{(l,t)}}, \tag{33}$$

with

$$u_i^{(l,t)} \in [0, 1], \qquad s_m^{(l,t)} \geq 0, \qquad \kappa_l \geq 0, \qquad G^{(l)} \geq 0. \tag{34}$$

Proposition A.3 (Monotone accumulation). For every block l, step t, point i, and prompt k,

$$\lambda_{ik}^{(l,t+1)} \geq \lambda_{ik}^{(l,t)}. \tag{35}$$

Proof. Define the propagation increment

$$\Delta_{ik}^{(l,t)} = u_i^{(l,t)} \sum_{m=1}^{M_{l,t}} s_m^{(l,t)} \kappa_l\big(p_i, c_m^{(l,t)}\big) G^{(l)}_{k\, q_m^{(l,t)}}. \tag{36}$$

Every factor is nonnegative by Eq. (34), so $\Delta_{ik}^{(l,t)} \geq 0$. Therefore,

$$\lambda_{ik}^{(l,t+1)} = \lambda_{ik}^{(l,t)} + \Delta_{ik}^{(l,t)} \geq \lambda_{ik}^{(l,t)}. \tag{37}$$

Proposition A.4 (Prompt-selective relative sharpening). Even though the raw increments are nonnegative, the propagation remains discriminative. For any two prompts k_1 and k_2,

$$\begin{aligned} &\left(\lambda_{ik_1}^{(l,t+1)} - \lambda_{ik_2}^{(l,t+1)}\right) - \left(\lambda_{ik_1}^{(l,t)} - \lambda_{ik_2}^{(l,t)}\right) \\ &= u_i^{(l,t)} \sum_{m=1}^{M_{l,t}} s_m^{(l,t)} \kappa_l\big(p_i, c_m^{(l,t)}\big) \left(G^{(l)}_{k_1\, q_m^{(l,t)}} - G^{(l)}_{k_2\, q_m^{(l,t)}}\right). \end{aligned} \tag{38}$$

Proof. Applying Eq. (33) to prompts k_1 and k_2 gives

$$\lambda_{ik_1}^{(l,t+1)} = \lambda_{ik_1}^{(l,t)} + u_i^{(l,t)} \sum_{m=1}^{M_{l,t}} s_m^{(l,t)} \kappa_l\big(p_i, c_m^{(l,t)}\big) G^{(l)}_{k_1\, q_m^{(l,t)}}, \tag{39}$$

$$\lambda_{ik_2}^{(l,t+1)} = \lambda_{ik_2}^{(l,t)} + u_i^{(l,t)} \sum_{m=1}^{M_{l,t}} s_m^{(l,t)} \kappa_l\big(p_i, c_m^{(l,t)}\big) G^{(l)}_{k_2\, q_m^{(l,t)}}. \tag{40}$$

Subtracting the second equation from the first yields

$$\begin{aligned} \lambda_{ik_1}^{(l,t+1)} - \lambda_{ik_2}^{(l,t+1)} = \lambda_{ik_1}^{(l,t)} - \lambda_{ik_2}^{(l,t)} \\ + u_i^{(l,t)} \sum_{m=1}^{M_{l,t}} s_m^{(l,t)} \kappa_l\big(p_i, c_m^{(l,t)}\big) \left(G^{(l)}_{k_1\, q_m^{(l,t)}} - G^{(l)}_{k_2\, q_m^{(l,t)}}\right). \end{aligned} \tag{41}$$

Moving the previous-step difference to the left-hand side gives Eq. (38).

Interpretation. Equation (38) shows that propagation changes the *relative* prompt preference through the prompt-relation matrix $G^{(l)}$. Nonnegative raw increments therefore do not imply indiscriminate amplification: compatible prompts can still receive stronger excitation than incompatible prompts, and the final prediction remains governed by softmax competition over the resulting prompt intensities. These propositions describe operator-level properties of the propagation update; they do not by themselves guarantee segmentation gains. The empirical analyses below test whether the corresponding semantic organization and evidence routing improve the task metric.

A.4 Clarification of the Hawkes Terminology

The main paper uses the Hawkes terminology in a restricted sense. We do *not* claim that the propagation branch is a classical continuous-time stochastic Hawkes process. Rather, it is a finite-depth spatial excitation module whose future intensities depend additively on an event set extracted from the current semantic state. This event-driven additive accumulation is the Hawkes-style property preserved by our formulation. We prefer this terminology over sparse graph attention because the propagation source is a dynamically extracted semantic-event set rather than the full point set, and the update acts directly on prompt-conditioned intensities rather than on generic feature mixing.

A.5 Exact Event Construction and Routing

At propagation step t, define the confident set

$$\mathcal{V}^{(l,t)} = \left\{ i \;\middle|\; r_i^{(l,t)} \geq \eta_l \right\}. \tag{42}$$

We build a radius graph on $\mathcal{V}^{(l,t)}$, restricted to points with the same winning prompt:

$$(i,j) \in \mathcal{E}^{(l,t)} \iff i,j \in \mathcal{V}^{(l,t)}, \quad q_i^{(l,t)} = q_j^{(l,t)}, \quad \|p_i - p_j\|_2 \leq r_h^{(l)}. \tag{43}$$

Let $\{\mathcal{C}_m^{(l,t)}\}_{m=1}^{M_{l,t}}$ denote the connected components of this graph. For each component, we define

$$q_m^{(l,t)} = q_i^{(l,t)} \quad \text{for any } i \in \mathcal{C}_m^{(l,t)}, \tag{44}$$

$$c_m^{(l,t)} = \frac{1}{|\mathcal{C}_m^{(l,t)}|} \sum_{i \in \mathcal{C}_m^{(l,t)}} p_i, \tag{45}$$

and

$$s_m^{(l,t)} = \sum_{i \in \mathcal{C}_m^{(l,t)}} r_i^{(l,t)}. \tag{46}$$

These definitions align with the main text: the event center is the coordinate centroid of the connected component, and the event strength is its aggregated confidence mass.

A.6 Differentiability, Routing, and Early-Training Dynamics

The Hawkes branch contains three discrete operations:

- winning-prompt assignment by $\arg\max$,
- threshold-based confident-point selection,
- connected-component extraction on the induced radius graph.

These are treated as routing decisions rather than differentiable latent variables. Gradients are not propagated through carrier membership itself, but the continuous quantities conditioned on that routing remain differentiable, including base intensities $\lambda^{(l,0)}$, uncertainty weights $u_i^{(l,t)}$, event strengths $s_m^{(l,t)}$, prompt-relation parameters $U^{(l)}, V^{(l)}$, terminal posteriors $\pi^{(l,T)}$, and feedback embeddings $C^{(l)}$.

Why early training does not collapse. Before the prompt-conditioned feature state becomes moderately discriminative, few points satisfy the confidence criterion strongly enough to act as dominant carriers. Optimization is therefore driven mainly by the dense feature-shaping and base-scoring pathways. As those pathways improve, the extracted event topology becomes more stable, after which the propagation branch contributes increasingly meaningful corrections.

A.7 Complexity and Implementation Notes

Low-rank semantic shaping. The projector $\Pi^{(l)} = B^{(l)} B^{(l)\top}$ need not be formed explicitly. The low-rank component can be computed as

$$S^{(l)} = \left(\tilde{X}^{(l)} B^{(l)}\right) B^{(l)\top}, \tag{47}$$

which costs

$$\mathcal{O}(Ndr) \tag{48}$$

per block, up to the lightweight basis generator.

Hawkes-style propagation. At propagation step t, let $M_{l,t}$ be the number of extracted events and let $\bar{n}_{l,t}$ be the average number of points within the support of an event kernel. The propagation update then scales as

$$\mathcal{O}\left(\sum_{t=0}^{T-1} M_{l,t}\, \bar{n}_{l,t}\, K\right), \tag{49}$$

up to the sparse graph construction cost. Because only confident points participate in the graph, event extraction remains sparse in practice.

Overall overhead. The dominant extra cost of one coupled block is therefore

$$\mathcal{O}(Ndr) + \mathcal{O}\left(\sum_{t=0}^{T-1} M_{l,t}\, \bar{n}_{l,t}\, K\right) + \mathcal{O}(NKd), \tag{50}$$

where the last term comes from prompt-conditioned posteriors and feedback embeddings.

A.8 Pseudocode

Algorithms 1 and 2 summarize the forward computation and the event-extraction subroutine in a form aligned with the main text.

Algorithm 1 Forward Pass of Coupled Semantic Dynamics

Input: point cloud $\mathcal{P} = \{p_i\}_{i=1}^{N}$, prompt set $P \in \mathbb{R}^{K \times d}$, initial features $X^{(0)}$, number of blocks L, propagation depth T
Output: point-wise semantic predictions $\hat{y}$

Initialize propagated summary $h^{(-1)} \leftarrow 0$
for $l = 0, \ldots, L-1$ **do**
  Form block context $z^{(l)} = [\mathrm{Pool}(X^{(l)}); \mathrm{Pool}(P); h^{(l-1)}]$
  Predict $B^{(l)}$, $\Pi^{(l)}$, and gates $(\alpha_l, \rho_l, \beta_l, \gamma_l)$
  Compute low-rank semantic shaping via Eq. (6)
  Compute base intensities $\lambda^{(l,0)}$ via Eq. (11)
  for $t = 0, \ldots, T-1$ **do**
    Compute posteriors $\pi^{(l,t)}$ and uncertainties $u^{(l,t)}$
    Extract sparse events $\mathcal{E}^{(l,t)} \leftarrow \textsc{ExtractEvents}(\mathcal{P}, \pi^{(l,t)}, \eta_l, r_h^{(l)})$
    Update intensities $\lambda^{(l,t+1)}$ via Eq. (13)
  end for
  Compute terminal posteriors $\pi^{(l,T)} = \mathrm{Softmax}(\lambda^{(l,T)})$
  Construct feedback $C^{(l)} = \pi^{(l,T)} P W_h^{(l)}$
  Update $X^{(l+1)} \leftarrow \bar{X}^{(l)} + \gamma_l C^{(l)}$ and $h^{(l)} \leftarrow \mathrm{Pool}(C^{(l)})$
end for
return $\hat{y}_i = \arg\max_k \lambda_{ik}^{(L,T)}$

Algorithm 2 EXTRACTEVENTS Subroutine

Input: points $\mathcal{P}$, prompt posteriors π, confidence threshold η, support radius r_h
Output: event set $\mathcal{E} = \{(c_m, q_m, s_m)\}_{m=1}^{M}$

Compute winning prompts $q_i = \arg\max_k \pi_{ik}$ and confidences $r_i = \max_k \pi_{ik}$
Keep confident points $\mathcal{V} = \{i \mid r_i \geq \eta\}$
Build a radius graph on $\mathcal{V}$ with edges (i, j) if $q_i = q_j$ and $\|p_i - p_j\|_2 \leq r_h$
Extract connected components $\{\mathcal{C}_m\}_{m=1}^{M}$
for $m = 1, \ldots, M$ **do**
    Assign $q_m \leftarrow q_i$ for any $i \in \mathcal{C}_m$
    Compute centroid $c_m \leftarrow \frac{1}{|\mathcal{C}_m|} \sum_{i \in \mathcal{C}_m} p_i$
    Compute strength $s_m \leftarrow \sum_{i \in \mathcal{C}_m} r_i$
end for
return $\mathcal{E} = \{(c_m, q_m, s_m)\}_{m=1}^{M}$

B Additional Experimental Validation

The following analyses examine carrier reliability, feature organization, and sensitivity under the aligned protocol.

B.1 Wrong-Carriers and Sensitivity to Prompts and Thresholds

We define an extracted carrier as correct when the majority ground-truth part label in its connected component agrees with the carrier prompt. At the default threshold $\eta = 0.55$, the early-carrier precision is 88.7%. Wrong carriers are concentrated near boundaries, overlapping Obj-Part prompts, and sparse/noisy regions. The local diagnostic below compares point accuracy before and after propagation within the event support radius.

Table 6: Localized effects of correct and incorrect early carriers. Accuracy is measured for points within the propagation support of the indicated carrier group; lower accuracy near wrong carriers is a concrete failure mode.

Carrier neighborhood	Accuracy before (↑)	Accuracy after (↑)	Change (↑)
Near correct early carriers	42.1	48.4	+6.3
Near wrong early carriers	37.6	36.1	-1.5
Far from extracted carriers	34.3	34.9	+0.6

Among the wrong early carriers, 43.8% occur near boundaries, 33.5% under overlapping Obj-Part prompts, and 22.7% in sparse/noisy regions. Thus, locality, uncertainty weighting, and finite depth limit the spatial extent of an incorrect event, but do not eliminate this failure mode.

For prompt wording, we keep the trained model fixed and change only the evaluation prompts. The method is stable under simple paraphrases but is not prompt-invariant.

Table 7: Sensitivity to prompt wording with fixed model weights. The drop is measured relative to the default prompt template.

Prompt variant	Example prompts	All Avg. (↑)	Drop (↓)	Setting
Default	*leg*; *leg of chair*	36.57	0.00	Reference
Word-order	*chair leg*; *chair back*	36.46	0.11	Paraphrase
Possessive Obj-Part	*chair's leg*; *chair's back*	36.45	0.12	Paraphrase
Attribute-heavy	*thin chair leg*; *rear chair back*	36.20	0.37	Added attributes
Ambiguous synonym	*support of chair*; *frame of chair*	36.00	0.57	Ambiguous wording

For carrier selection, only η changes while the trained weights remain fixed. Lower thresholds increase coverage but admit noisier carriers; higher thresholds improve precision but remove useful evidence for thin or sparse parts. Coverage is the average number of extracted events per block per step.

B.2 Feature Organization and Shaping Variants

The following measurements quantify the feature changes across the semantic-dynamics stack. Compactness is the average cosine distance from a point feature to its ground-truth part centroid;

Table 8: Carrier-threshold sensitivity with fixed model weights. Precision and coverage have opposite operating trends, while All Avg. reports object mIoU.

Carrier threshold η	Precision (↑)	Wrong ratio (↓)	Coverage (↑)	All Avg. (↑)
0.45	84.9	15.1	24.2	36.12
0.50	87.6	12.4	20.8	36.43
0.55	88.7	11.3	18.9	36.57
0.60	91.0	9.0	16.4	36.50
0.65	92.7	7.3	12.9	36.22

separation is the average cosine distance between part centroids; silhouette uses ground-truth part labels in feature space. These are diagnostic metrics and are not additional training losses.

Table 9: Feature organization across the frozen backbone and semantic-dynamics blocks. Values are mean±standard deviation over five seeds. Lower compactness and higher separation, silhouette, and mIoU are better.

Feature checkpoint	Compactness (↓)	Separation (↑)	Silhouette (↑)	All Avg. mIoU (↑)
Frozen Backbone	0.259 ± 0.005	0.407 ± 0.007	0.118 ± 0.009	32.72 ± 0.16
After Block 1	0.244 ± 0.005	0.420 ± 0.007	0.139 ± 0.009	34.46 ± 0.14
After Block 2	0.232 ± 0.004	0.435 ± 0.006	0.156 ± 0.008	35.83 ± 0.12
After Block 3	0.223 ± 0.003	0.442 ± 0.005	0.167 ± 0.007	36.57 ± 0.09

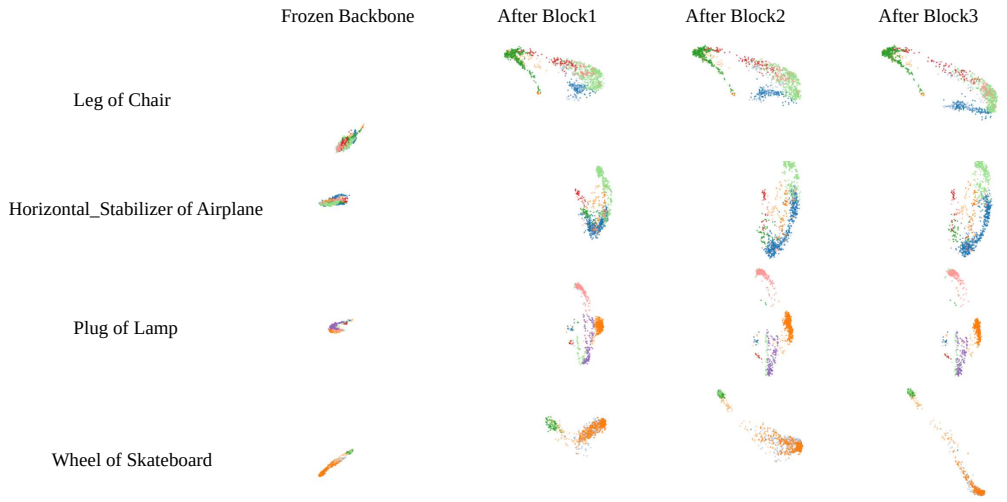


Figure 5: PCA visualization of point features after the frozen backbone and after Blocks 1–3. For each object-prompt instance, PCA is fitted jointly to the four feature snapshots, so all panels in a row share a projection and coordinate range. Colors denote ground-truth part labels. The visualization complements the feature-organization measurements in Table 9.

B.3 Blockwise Basis and Low-Rank Branch Controls

The shared blockwise basis is one prompt-conditioned basis for the current object-prompt instance; point-specific updates still arise from the projected features and residual mapping. We compare it with a pointwise adaptive basis, a model without low-rank shaping, and a parameter-matched bottleneck adapter under the same insertion and training protocol.

Table 10: Low-rank semantic-shaping controls. Values are mean±sample standard deviation over five seeds. The first group tests the contribution and form of low-rank shaping; the second compares shared blockwise and pointwise adaptive bases. Lower compactness and higher separation and mIoU are better.

Variant	Compactness (↓)	Separation (↑)	Part Avg. (↑)	Obj. Avg. (↑)	All Avg. (↑)
Low-rank shaping controls					
w/o low-rank shaping	0.244 ± 0.004	0.418 ± 0.006	34.22 ± 0.11	31.96 ± 0.13	33.09 ± 0.10
Bottleneck adapter	0.237 ± 0.004	0.426 ± 0.006	36.08 ± 0.12	34.36 ± 0.12	35.22 ± 0.11
Basis granularity					
Pointwise adaptive basis	0.232 ± 0.005	0.433 ± 0.007	36.91 ± 0.16	34.83 ± 0.17	35.87 ± 0.16
Shared blockwise basis (full model)	0.223 ± 0.003	0.442 ± 0.005	37.56 ± 0.10	35.58 ± 0.12	36.57 ± 0.09

C Additional Experimental Details

C.1 Protocol and Key Settings

Frozen backbone and text encoder. We use the same frozen geometric feature extractor as CoSMo3D and the same frozen SigLIP text encoder as the aligned comparison suite. The gains reported in the main paper therefore come from the inserted semantic-dynamics stack rather than from stronger backbone features or different language supervision.

Trainable semantic-dynamics modules. Only the lightweight basis, relation, gating, and feedback heads introduced in Sec. 3 are optimized. The geometric backbone and text encoder remain fixed throughout training and evaluation.

Evaluation scope. Following the aligned protocol, training uses only the aligned-suite training split of 3DCompat200. Evaluation uses the 3DCompat200 test split and the evaluation-only ShapeNetPart and PartNetE benchmarks; our method is not trained or adapted on either of the latter datasets. For each benchmark, we report object mIoU under both Part and Obj-Part prompt regimes, and under both canonical and rotated settings when defined by the benchmark suite.

Inference and matched controls. Our method uses a single forward pass at test time, without iterative optimization or additional post-processing. In the matched-control comparisons, generic alternatives are inserted at the same block locations and trained under the same frozen-backbone protocol so that the comparison isolates the effect of low-rank semantic shaping and Hawkes-style propagation rather than changes in the backbone or evaluation pipeline.

C.2 Baseline Notes and Aligned-Suite Sourcing

The main paper uses aligned-suite results or aligned-suite reproductions for direct comparisons. PointCLIPV2, OpenMask3D, PartSLIP, Find3D*, and CoSMo3D use aligned-suite reruns or reproductions with the same prompts, splits, canonical/rotated settings, and object-level mIoU; the original Find3D results are also included for reference. PartSLIP++† is separately marked for its fine-tuned canonical-view setting used on PartNetE. These distinctions are stated here to clarify the source and protocol of each reported baseline.

C.3 Efficiency and Event Statistics

Table 11 reports the appendix-level efficiency statistics corresponding to the structural complexity discussion in Appendix A.

Table 11: Efficiency and event-statistics summary. Relative latency is measured from inference-only wall-clock time under the same hardware and batch-size setting, and is normalized to the full model. Average events per block per step are reported only for event-driven variants.

Variant	All Avg. (↑)	Relative Latency (↓)	Extra Params (↓)	Avg. Events / Block / Step (↓)
Full model	36.57	1.00×	+0.82M	18.9
Replace low-rank with bottleneck adapter	35.22	0.97×	+0.79M	17.4
Replace Hawkes with sparse GAT propagation	35.79	1.14×	+0.88M	N/A

C.4 Extended Generic Controls

Table 12 expands the generic-control comparison beyond the compact main-text table while staying focused on alternatives that directly test the low-rank semantic shaping and Hawkes-propagation claims under the same protocol.

Table 12: Extended generic-control comparison under the same protocol. All Avg. is reported as mean±sample standard deviation over five seeds; relative latency is normalized to the full model.

Variant	3DCompat200-Fine Obj-Part (↑)	Part Avg. (↑)	Obj. Avg. (↑)	All Avg. (↑)	Extra Params (↓)	Relative Latency (↓)
Residual MLP / bottleneck adapter	28.23	36.08	34.36	35.22 ± 0.11	+0.79M	0.97×
Sparse kNN propagation	28.46	36.06	34.88	35.47 ± 0.12	+0.74M	0.90×
Sparse GAT propagation	28.74	36.56	35.02	35.79 ± 0.13	+0.88M	1.14×
Ours	29.63	37.56	35.58	36.57 ± 0.09	+0.82M	1.00×

For event-driven variants, the average number of events per block per step is computed by averaging the extracted event count $M_{l,t}$ over all blocks, propagation steps, and evaluation samples.

D Extended Experimental Analyses

This section collects the additional analyses reported during review. Model variants in the robustness, propagation-control, and encoder studies are retrained under the aligned 3DCompat200 training protocol with the same frozen-backbone setting unless stated otherwise. Prompt-wording and carrier-threshold studies keep the trained weights fixed and change only the evaluation-time prompt or threshold. The reported diagnostics therefore separate training variation from inference-time sensitivity.

D.1 Multi-seed Depth and Feedback Diagnostics

We varied only the propagation depth T in the full model and normalized inference cost to the default $T = 3$. The extended sweep supports a plateau interpretation: $T = 4$ and $T = 5$ have marginally higher means, but their differences from $T = 3$ are within seed variation and incur additional cost; larger depths return to the $T = 3$ level or degrade.

Table 13: Five-seed propagation-depth sweep. All other model and protocol settings are fixed. Relative cost is normalized to $T = 3$; higher mIoU is better and lower relative cost is better.

Propagation depth T	All Avg. mean ± std. (↑)	Relative cost (↓)	Interpretation
1	35.73 ± 0.10	0.86×	–
2	36.10 ± 0.10	0.93×	–
3	36.57 ± 0.09	1.00×	Default
4	36.61 ± 0.08	1.08×	–
5	36.62 ± 0.10	1.16×	–
6	36.57 ± 0.10	1.24×	–
8	36.37 ± 0.09	1.40×	–
10	36.16 ± 0.10	1.56×	–

The depth sweep is reported separately from the core-ablation rerun; its standard deviations are computed over the five runs of this sweep.

D.2 Propagation Components and Generic Controls

The main paper's Table 5 compares the complete propagation branch with generic operators under the same aligned protocol, changing only the propagation operator. The following appendix table instead removes one event-driven ingredient at a time while keeping the remaining stack unchanged, testing the roles of event aggregation, target uncertainty, prompt relations, and locality.

Table 14: Component controls for the event-driven propagation branch. Each row changes one propagation ingredient under the same aligned protocol.

Propagation variant	Fine Obj-Part (↑)	All Avg. (↑)
Full event-driven propagation	29.63	36.57
Individual confident points instead of connected events	29.18	36.18
Without uncertainty weighting	29.02	36.02
Without prompt-relation matrix	28.95	35.94
Without locality kernel	28.74	35.76

D.3 Task-specific Diagnostic Subsets

To examine the cases emphasized in the review, we evaluate the same trained models on diagnostic subsets covering regular/easier parts, repeated structures, sparse/fragmented parts, locally similar parts, and Obj-Part prompts. Subset membership is fixed from dataset metadata and prompt/object categories before evaluation; no model or evaluation parameter is changed between rows. The larger gains on the latter four groups are consistent with the role of event-driven propagation in ambiguity-heavy cases.

Table 15: Diagnostic subset analysis under the aligned evaluation protocol. Values are object mIoU (%), and all higher-is-better columns are marked with ↑.

Diagnostic subset	CoSMo3D (↑)	w/o Hawkes-style propagation (↑)	Full (↑)	Full–CoSMo3D (↑)	Full–w/o propagation (↑)
Regular/easier	42.18	43.05	43.69	+1.51	+0.64
Repeated structures	28.72	30.18	31.84	+3.12	+1.66
Sparse/fragmented	25.44	26.58	28.39	+2.95	+1.81
Locally similar parts	27.36	28.61	30.26	+2.90	+1.65
Obj-Part prompts	32.72	33.95	35.58	+2.86	+1.63

D.4 Feedback Coupling Diagnostics

We compare the full model with the same architecture trained without feeding propagated context into later shaping blocks. The feedback effect is modest on aggregate and larger for object-conditioned prompts and initially low-confidence points.

Table 16: Feedback-coupling diagnostics. Higher point accuracy and mIoU are better.

Evaluation slice or metric	No feedback (↑)	Full model (↑)	Gain (↑)
All Avg.	36.24	36.57	+0.33
Part Avg.	37.29	37.56	+0.27
Obj. Avg.	35.19	35.58	+0.39
Fine Obj-Part, canonical	29.05	29.63	+0.58
Fine Obj-Part, rotated	28.61	29.16	+0.55
Low-confidence point accuracy	21.91	22.76	+0.85

D.5 Evidence Spread Across Confidence Groups

Here, "spread" means additive prompt-intensity evidence transfer from high-confidence semantic carriers to nearby uncertain points; it does not mean averaging point features. Points are grouped by their base prompt-posterior confidence before propagation. The largest improvement is concentrated in the low-confidence group.

Table 17: Step-wise evidence transfer measured by point-wise accuracy in confidence groups. The same trained model is evaluated after each propagation step.

Propagation state	Low-confidence accuracy (↑)	Medium-confidence accuracy (↑)	High-confidence accuracy (↑)
Before propagation	18.31	44.67	78.42
After 1 step	20.37	45.70	78.55
After 2 steps	21.74	46.54	78.74
After 3 steps	22.76	47.12	78.89

D.6 Encoder and Backbone Sensitivity

We replace one frozen encoder at a time while keeping the other encoder and the aligned evaluation protocol fixed. For each setting, the corresponding semantic-dynamics heads are trained on the same

3DCompat200 training split. The results measure sensitivity to the encoder pair rather than claim encoder independence.

Table 18: Sensitivity to frozen text and point encoders. The baseline disables the semantic-dynamics stack; the full model trains the corresponding heads on the same frozen encoder pair.

Encoder setting	Base w/o dynamics (↑)	Full dynamics (↑)	Gain (↑)
PTv3 + SigLIP	32.72	36.57	+3.85
PTv3 + CLIP	31.84	35.05	+3.21
PTv3 + OpenCLIP	32.05	35.32	+3.27
PTv2 + SigLIP	31.31	33.65	+2.34

Table 19: Component ordering under two frozen point encoders. Only the point encoder changes; SigLIP, prompts, splits, and the training/evaluation protocol are fixed.

Point encoder	No dynamics (↑)	w/o low-rank (↑)	w/o propagation (↑)	Full (↑)
PTv3	32.72	33.09	35.28	36.57
PTv2	31.31	31.54	32.87	33.65

Finally, we compare the frozen-backbone setting used in the main paper with end-to-end point-backbone fine-tuning. This is a three-seed analysis: SigLIP remains frozen, while the point backbone is either frozen or trainable.

Table 20: Frozen versus unfrozen point-backbone training. Memory and time are relative training costs; ShapeNetPart (SNP) statistics are reported to assess transfer stability.

Backbone training setting	Memory (↓)	Time (↓)	Overall (↑)	SNP mean (↑)	SNP std. (↓)
Frozen, no dynamics	16.4 GB	$1.00\times$	32.72	32.02	0.19
Frozen + dynamics	18.6 GB	$1.12\times$	36.54	36.68	0.10
Unfrozen, no dynamics	40.7 GB	$2.18\times$	32.83	31.70	0.44
Unfrozen + dynamics	43.2 GB	$2.35\times$	36.64	36.44	0.46

Unfreezing changes the overall mean only marginally ($36.54 \rightarrow 36.64$) while more than doubling memory and increasing the relative training cost. It also lowers the ShapeNetPart mean and increases its seed variation, supporting the frozen-backbone protocol as the cleaner setting for isolating the semantic-dynamics heads.

E Mechanism Validation Analyses

This section provides the direct evidence referred to in the main paper: low-rank semantic shaping should improve part-level compactness, and Hawkes-style propagation should contribute most strongly on initially ambiguous points.

E.1 Primary Mechanism Validation

Table 21: Primary mechanism-validation results.

(a) Part-level Feature Compactness

Variant	Part Compactness (↓)	Obj. Compactness (↓)	All Compactness (↓)
w/o low-rank semantic shaping	0.231	0.257	0.244
Full model	0.209	0.236	0.223

(b) Confidence-binned Propagation Gains

Confidence Bin	w/o propagation (↑)	Full model (↑)	Gain (↑)
High	78.42	78.89	+0.47
Medium	44.67	47.12	+2.45
Low	18.31	22.76	+4.45

Notes. In (a), compactness is measured by average intra-part cosine distance to the ground-truth part centroid on the terminal point features; lower is better. In (b), the confidence bins are computed from the final-block base posterior confidence $r_i^{(L,0)}$ before Hawkes-style propagation on the representative ambiguity-heavy slice used in the main text.

E.2 Metric Definitions

For the compactness analysis in Table 21(a), let $f_i \in \mathbb{R}^d$ denote the terminal point feature of point i, and let μ_c be the centroid of the ground-truth part c containing point i:

$$\mu_c = \frac{1}{|\Omega_c|} \sum_{j \in \Omega_c} f_j. \tag{51}$$

The pointwise compactness distance is measured by cosine distance,

$$d_i = 1 - \frac{\langle f_i, \mu_c \rangle}{\|f_i\|_2 \|\mu_c\|_2}, \tag{52}$$

and the reported compactness is the mean of d_i over the relevant evaluation split.

For the confidence-binned analysis in Table 21(b), let $r_i^{(L,0)}$ denote the final-block base confidence before Hawkes-style propagation. We partition points into high, medium, and low confidence groups according to fixed quantile bins:

$$\mathcal{B}_{\text{high}}, \qquad \mathcal{B}_{\text{medium}}, \qquad \mathcal{B}_{\text{low}}. \tag{53}$$

In Table 21(b), the three bins are formed by tertiles of the point-wise base confidence $r_i^{(L,0)}$ on the evaluated split, and accuracy is measured as point-wise top-1 label accuracy within each bin.

F Extended Qualitative Results

This section contains the extra qualitative evidence supporting the main-text teaser and qualitative figure without duplicating them.

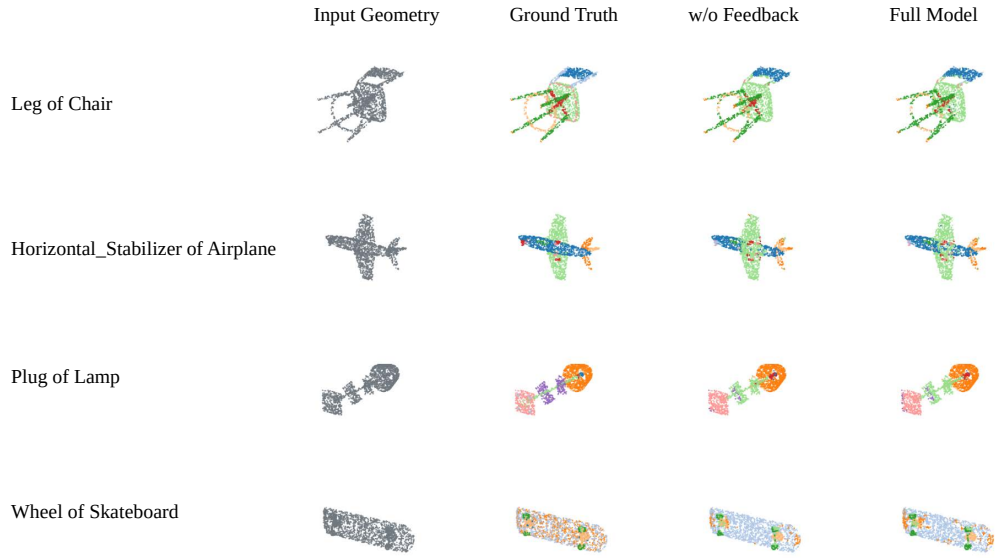


Figure 6: Paired qualitative comparison of the full model and the model trained without feedback coupling on the same objects and prompts. Each example shows the input, ground-truth part, and predictions with and without feedback coupling. The examples illustrate both improvements and regressions; aggregate feedback effects are reported in Table 16.

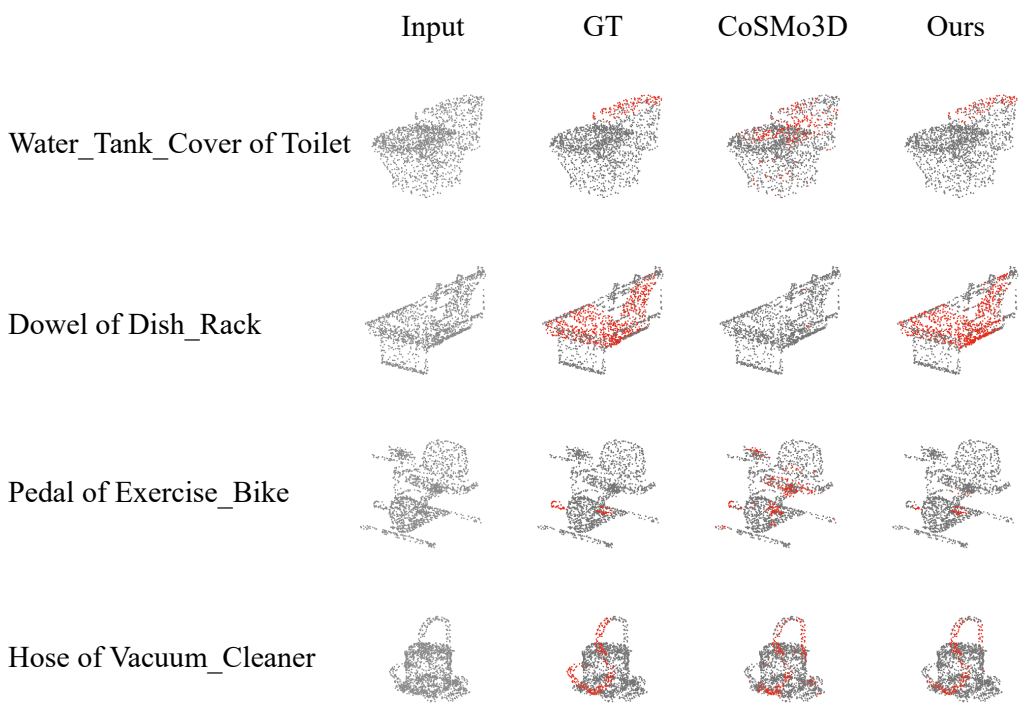


Figure 7: Extended qualitative comparison on additional object-conditioned part queries. Each row shows the input point cloud, ground-truth target part, CoSMo3D prediction, and our prediction. Gray points denote the object context, while colored points denote the queried part.

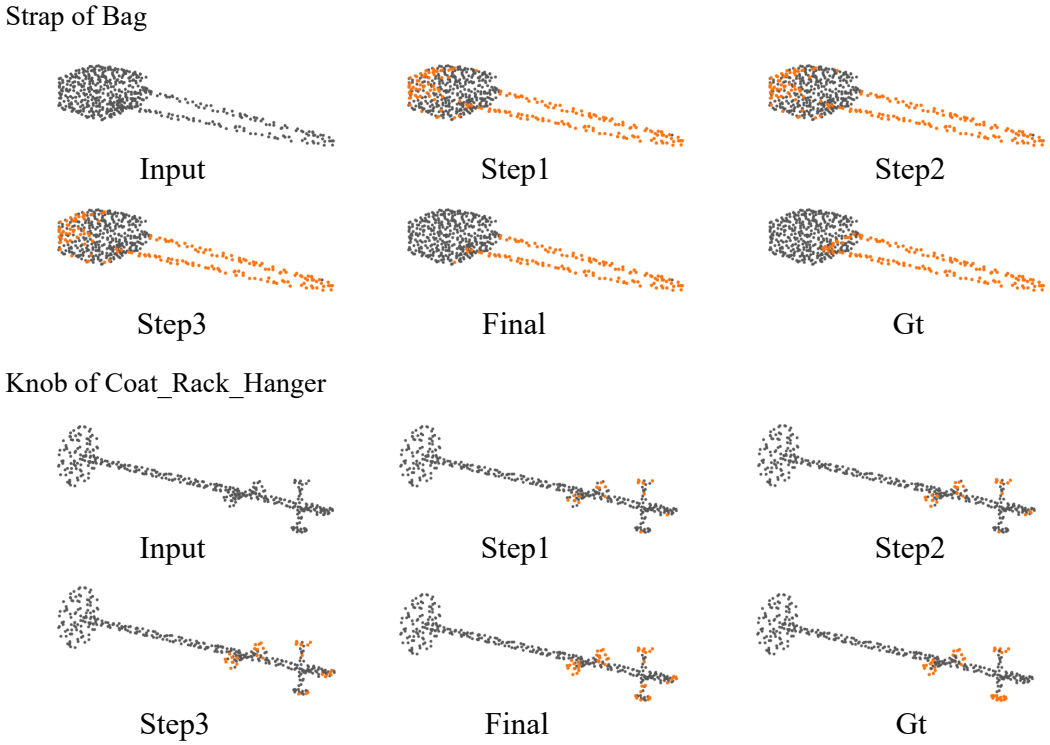


Figure 8: Step-wise visualization of semantic propagation for representative object-conditioned prompts. The finite-depth propagation progressively expands reliable evidence from confident carriers to nearby ambiguous regions, yielding more complete final predictions than early steps.

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: The abstract and introduction claim a semantic-dynamics framework with low-rank semantic shaping and Hawkes-style semantic propagation, and these claims are supported by Secs. 3–4 and the appendix analyses.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: See the **Limitations** paragraph in the conclusion, which discusses sparse sampling, overlapping prompts, and the gap between continuous semantics and discrete point clouds.

   Guidelines:

   - The answer [N/A] 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: The assumptions for Propositions 1–4 are stated in Sec. 3, and the full derivations and proofs are provided in Appendix A.

Guidelines:

- The answer [N/A] 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: Implementation details for the main method, protocol, and ablations are provided in Sec. 4 and Appendix C.

Guidelines:

- The answer [N/A] 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: The submission uses public datasets and public pretrained components, but it does not provide an anonymized code release or a full reproduction package at submission time.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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) necessary to understand the results?

   Answer: [Yes]

   Justification: Training and evaluation settings are described in Sec. 4 and Appendix C.

   Guidelines:

   - The answer [N/A] 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: The experiments report single mIoU values without error bars, confidence intervals, or repeated-run significance tests.

   Guidelines:

   - The answer [N/A] 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: Appendix C reports efficiency-related information including relative latency and event statistics, and the paper discusses the computational overhead of the semantic-dynamics modules.

   Guidelines:

   - The answer [N/A] 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: The paper studies an algorithmic method on public benchmarks and pretrained models, and we are not aware of any aspect of the work that conflicts with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] 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: [No]

Justification: The current submission does not contain a dedicated discussion of broader positive and negative societal impacts.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] 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., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release a high-risk generative model, scraped dataset, or other asset that would require special release safeguards.

Guidelines:

- The answer [N/A] 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 paper credits the datasets, baselines, and pretrained models used throughout the experiments, and these assets are used under their original public release settings and benchmark protocols.

Guidelines:

- The answer [N/A] 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: [N/A]

    Justification: The submission does not introduce or release a new dataset, benchmark, or standalone model asset.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: The paper does not involve crowdsourcing, human annotation studies, or experiments with human subjects.

    Guidelines:

    - The answer [N/A] 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: [N/A]

Justification: The paper does not involve crowdsourcing or human subjects research, so IRB or equivalent review is not applicable.

Guidelines:

- The answer [N/A] 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 rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method does not use an LLM as an important or non-standard component; it uses a frozen SigLIP text encoder rather than an LLM-based reasoning module.

Guidelines:

- The answer [N/A] 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 in the NeurIPS handbook for what should or should not be described.