PAGER: PARTIAL-TO-GLOBAL ALIGNMENT VIA GEOMETRIC AND RELATIONAL DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Pretrained 3D encoders are typically developed on globally reconstructed scenes expressed in a consistent world coordinate frame, whereas embodied systems must reason from partial, viewpoint-dependent observations in camera coordinates. We show that this shift from globally learned 3D feature spaces to realistic partial observations exposes a severe representation mismatch, which we find consistently across representative state-of-the-art encoders, including Sonata and Concerto. A frozen Sonata encoder with a global linear probe achieves **72.47 mIoU** on full ScanNet scenes, but **only 2.57 mIoU** on single-frame camera-coordinate inputs. Training-free gravity alignment recovers performance to 41.64 mIoU, showing that coordinate-frame mismatch is a dominant source of degradation but cannot be fully resolved through canonicalization alone. We introduce **PAGER**, a label-free adaptation method that aligns partial-view features with a frozen global 3D semantic space using only paired partial/global geometry. It learns lightweight adaptation modules while keeping the pretrained encoder and global segmentation probe frozen. Matched-point feature alignment anchors partial features to their global counterparts, while relational supervision preserves their similarity structure with respect to the global representation. Global geometry provides supervision only during training. Inference operates directly on the partial observation. Without partial-view labels, PAGER outperforms label-supervised PEFT on both Sonata and Concerto, and in zero-shot ScanNet→ScanNet++ transfer surpasses fully fine-tuned Sonata (53.93 vs. 48.09 mIoU), suggesting that preserving the frozen global representation can improve cross-dataset transfer.

1 INTRODUCTION

Large-scale 3D representation learning has significantly advanced semantic scene understanding, enabling pretrained point-cloud encoders to transfer effectively to downstream tasks such as semantic/instance segmentation, and scene parsing (Qi et al., 2017; Zhao et al., 2021; Wu et al., 2024; Schult et al., 2023; Xie et al., 2020; Wu et al., 2023; Wang et al., 2024a; Wu et al., 2025; Zhang et al., 2025). Most existing models, however, are developed and evaluated on globally reconstructed scenes, where geometry is aggregated across views in a consistent world-coordinate frame (Dai et al., 2017; Yeshwanth et al., 2023). This setting differs from realistic embodied perception, where an agent observes only a partial point cloud from its current viewpoint, with incomplete geometry, occlusions, limited FOV, and camera-coordinate inputs (Fig. 1). Similar viewpoint gaps have been studied in egocentric and ego-exo video representation learning (Grauman et al., 2022; 2023; Luo et al., 2025), but their effect on pretrained 3D point-cloud representations remains underexplored.

We show that this gap is not merely a consequence of missing geometry, but a severe representation mismatch between globally learned 3D feature spaces and view-constrained partial observations. Using Sonata (Wu et al., 2025) as a strong representative pretrained encoder, we find that a frozen encoder with a global linear probe reaches 72.47 mIoU on full ScanNet scenes, but drops to 2.57 mIoU on single-frame camera-coordinate partial views. Notably, when the same partial views remain expressed in world coordinates, performance remains much higher at 67.45 mIoU (Sec. 3). This indicates that camera-coordinate partial inputs can lose compatibility with the semantic feature space learned from globally reconstructed scenes. Recent progress in monocular and multi-view 3D

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

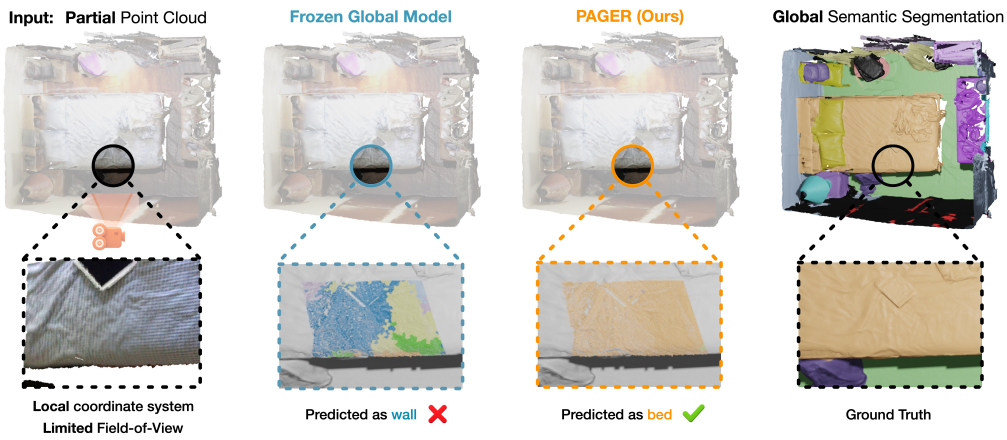


Figure 1: **Problem formulation - Partial Semantic Segmentation.** Given a 3D point cloud obtained from monocular 3D reconstruction with limited field-of-view (FoV)(left), our goal is to predict semantic labels for the **visible** points (top, circle) without relying on additional scene context. We propose PAGER, a framework specifically designed to address this challenging scenario. This setting mirrors real-world challenges in semantic scene understanding from monocular onboard cameras, where models must infer object semantics from partial and incomplete observations, for instance, identifying a bed region in a bedroom despite limited FoV and strong visual similarity to surrounding surfaces (*white, flat*). Here, the globally trained model Wu et al. (2025) misclassifies a bed as wall. *Transparent points denote unobserved regions shown only for reference.*

reconstruction (Wang et al., 2024b; 2025a;b) further motivates this setting, as view-constrained 3D observations are becoming increasingly available from casually captured visual input.

A straightforward solution is to fine-tune the pretrained encoder on labelled camera-view point clouds fully. This means updating the full encoder backbone together with a segmentation head using dense semantic supervision on the target partial-view training split. While this can be effective in-domain, it is computationally and storage expensive since each target setting requires an independently adapted model. It also depends on dense semantic annotations of partial 3D observations, which are costly to obtain, and may reduce cross-dataset generalization when only limited labelled views are available. Parameter-efficient fine-tuning (PEFT) reduces the number of updated parameters (Houlsby et al., 2019; Hu et al., 2022; Li & Liang, 2021; Zaken et al., 2022), and geometry-aware methods such as GEM (Tang et al., 2025) have shown that lightweight spatial adaptation can be effective for 3D scene segmentation. However, existing PEFT methods rely on semantic supervision in the target domain and have not been applied to partial-view segmentation, where label biases and missing scene context further limit their effectiveness.

To address this limitation, we propose PAGER, a label-free partial-to-global adaptation framework that maps partial camera-coordinate observations back into the semantic feature space of a frozen globally pretrained model. Our key premise is that adapting individual features independently is insufficient: a partial observation should recover not only the global feature associated with the same physical point, but also its relationships to the broader global representation.

We therefore derive two complementary forms of supervision entirely from paired partial/global geometry. First, matched-point feature anchoring aligns each partial feature with the global feature of the corresponding physical point. Second, a point-specific relational objective preserves how that feature relates to reference features sampled from the global scene. This transfers structure from the complete-scene representation without requiring semantic annotation of the partial observations. The pretrained encoder and global segmentation probe remain frozen throughout adaptation. Paired global geometry is used only during training. At inference, PAGER operates directly on partial camera-coordinate point clouds and requires neither the complete scene nor camera pose.

We evaluate PAGER under a partial-view semantic segmentation protocol constructed from standard 3D scene datasets, including ScanNet and ScanNet++ (Dai et al., 2017; Yeshwanth et al., 2023). On ScanNet, PAGER reaches 65.46 mIoU, outperforming adapter-based PEFT (Houlsby et al., 2019) and GEM (Tang et al., 2025) without partial-view semantic labels. In cross-dataset transfer from ScanNet to ScanNet++, PAGER reaches 53.93 mIoU and surpasses the fully fine-tuned

ScanNet Sonata baseline, suggesting that aligning to the frozen global representation can preserve transferability better than adapting the backbone to a labelled source domain.

We summarize the contributions as follows:

- We isolate coordinate-frame mismatch, not missing geometry, as the dominant failure mode of globally pretrained 3D encoders on partial views (-67 vs. -5 mIoU).
- We propose PAGER, which repairs this mismatch from paired partial/global geometry alone, keeping the encoder and probe frozen, requiring no global views or camera pose at inference.
- With 1.3M parameters and no labels, PAGER outperforms supervised PEFT in-domain (65.46 mIoU) and fully fine-tuned Sonata under domain shift (53.93 vs. 48.09).

2 Related Work

3D representation learning. Point-based and transformer-based architectures have driven rapid progress in 3D semantic segmentation (Neshev et al., 2025; Qi et al., 2017; Zhao et al., 2021; Wu et al., 2024; Schult et al., 2023). In parallel, self-supervised pretraining has become the main route to transferable 3D features (Xiao et al., 2023). PointContrast introduced correspondence-based pretraining on aligned RGB-D observations (Xie et al., 2020). Masked Scene Contrast and GroupContrast scaled scene-level pretraining and improved semantic grouping (Wu et al., 2023; Wang et al., 2024a); and Sonata explicitly addressed geometric shortcut learning to produce stronger frozen representations (Wu et al., 2025). However, these models are trained and evaluated primarily on globally reconstructed scenes. Our focus is the deployment gap that arises when such representations are applied to single-view partial observations.

Egocentric and cross-view understanding. Cross-view learning has been studied extensively in video through ego/exo datasets and view-invariant representation learning (Grauman et al., 2022; 2023; Luo et al., 2025). In contrast to video, where viewpoint invariance is an established objective, the analogous gap in pretrained 3D point-cloud representations has received little attention. At the same time, recent 3D reconstruction systems such as DUSt3R, VGGT, and PI3 (Wang et al., 2024b; 2025a;b) make dense partial observations from monocular capture readily available, making this gap increasingly consequential. Dense per-point semantic segmentation from a single partial camera-coordinate cloud, without multi-frame fusion or reconstruction, remains largely unexplored.

Efficient adaptation and PEFT for 3D. Standard PEFT methods: adapters (Houlsby et al., 2019), LoRA (Hu et al., 2022), prefix tuning (Li & Liang, 2021), and BitFit (Zaken et al., 2022), are effective in language and vision but transfer poorly to point clouds, as they adapt at the per-point level or inject static global tokens, ignoring local spatial structure (Tang et al., 2025). GEM (Tang et al., 2025) addresses this with geometry-aware adapters, achieving performance close to full fine-tuning with a small fraction of trainable parameters, but operates in the fully reconstructed-scene setting and relies on semantic supervision.

In contrast, PAGER targets a more challenging deployment setting: adapting a pretrained global 3D encoder to view-constrained point clouds in camera coordinates, where both scene context and the coordinate frame differ from pretraining. Rather than relying on semantic labels, PAGER learns a lightweight translator from paired partial/global geometry alone. Its self-distillation objective transfers both local and scene-level structure by matching similarity relations to shared global keys, addressing the combined effects of coordinate mismatch and limited context in local views.

3 Representation Gap Analysis

Before introducing PAGER, we quantify the mismatch between globally pretrained 3D encoders and view-constrained partial observations. The goal is to isolate *why* the standard frozen-transfer recipe fails, and what the adaptation module needs to correct.

Evaluation conditions. We construct four input conditions that progressively increase deployment realism and evaluate them in world and camera coordinate frames (Fig. 2). **Global** is the scene-level point cloud accumulated from all frames, corresponding to the conventional training and evaluation setting for 3D scene understanding. **Room Slice** retains one sixth of the global point cloud, obtained by partitioning the scene into six slices from the room center. It introduces coarse geometric

incompleteness while preserving the world coordinate frame. **View Crop** is a spatial crop of the global point cloud centered at a single camera viewpoint, retaining only points within the region a single frame would observe while keeping full geometric fidelity within the crop. **Single Frame** contains the 3D points back-projected from one RGB-D observation, reflecting realistic deployment with limited field of view, missing geometry, and depth noise.

For the **World** variants, all points are represented in the global world coordinate system. For the **Camera** variants, points are instead represented in a camera coordinate system. Importantly, only the Camera—Global variant (Fig. 2, $5^{th}col.$) transforms the entire accumulated scene into the coordinate frame of the first image. This setting is purely a diagnostic experiment. This first-frame transformation is not used in our main method or main experiments. The remaining Camera variants, View Crop and Single Frame (Fig. 2, 6th, $7^{th}col.$), do *not* use the first-frame coordinate system. Instead, each observation is represented in the coordinate frame of its corresponding camera pose.

Two separable sources of degradation. Table 1 reveals an asymmetry. Within world coordinates, performance decreases gradually as inputs become more geometrically incomplete: from 72.47 mIoU on Global down to 67.45 mIoU on Single Frame, a drop of under 5 points across four conditions spanning the full range from complete to single-frame reconstruction. (This 67.45 mIoU value uses ground-truth camera pose to transform Single Frame inputs into world coordinates, an oracle reference not available to PAGER at inference.) The pretrained representation is more robust to missing geometry when the coordinate frame is preserved. The coordinate frame tells a completely different story. Expressing the full Global scene in camera coordinates, with no geometry removed at all, already collapses mIoU from 72.47 to 5.14. Adding single-frame partiality brings it to 2.57. The coordinate frame alone, applied to a geometrically complete scene, causes a 67-point drop. Geometry reduction, by contrast, causes only a 5-point drop. Within the camera-coordinate setting, geometry reduction continues to degrade performance (5.14 → 2.57 mIoU), confirming that both factors contribute, with the coordinate frame being the larger and qualitatively distinct effect.

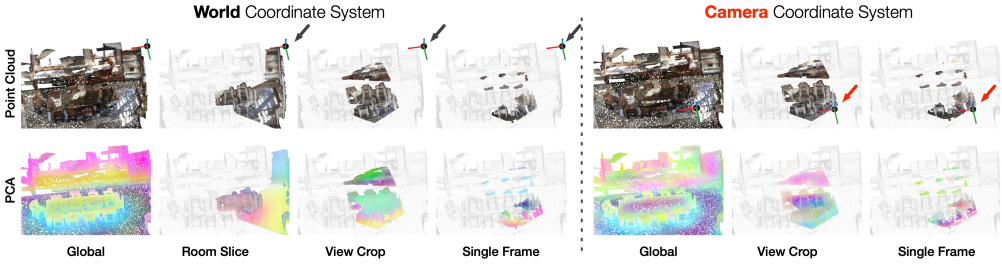


Figure 2: **Two sources of degradation.** *Left:* progressive geometric reduction (Global → Room Slice → View Crop → Single Frame) in world coordinates lowers performance due to missing context. *Right:* the same inputs with camera coordinates causes a larger drop, visible in both segmentation predictions (Tab. 1) and **PCA feature visualizations** (bottom row). Coordinate-frame mismatch is the dominant factor, geometry reduction further compounds the gap. Coordinate system origin shown by black/red arrows.

Why does the coordinate frame matter? The Camera—Global condition applies a single rigid transform to an otherwise unchanged complete scene. The 67-point drop therefore cannot be attributed to missing scene content. The pretrained encoder often relies on world-frame priors (floors below, gravity-aligned axes) that are disrupted under camera coordinates. Standard augmentations could not cover all possible variations introduced by camera frames, leaving this gap unaddressed.

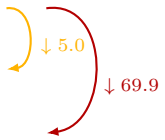
Can training-free canonicalization solve the gap? The strong coordinate-frame effect raises the question of whether the problem can be addressed without learning. We therefore evaluate two training-free canonicalization strategies. PCA-based axis alignment improves the frozen model from 2.57 to 8.38 mIoU. Gravity alignment, using pitch and roll estimated from a single RGB image with Perspective Fields (Jin et al., 2023), provides a substantially stronger recovery to 41.64 mIoU.

However, both remain well below PAGER at 65.46 mIoU. Coarse orientation recovery is therefore helpful but insufficient to restore the frozen global representation. We treat the remaining failure as a representation mismatch and directly adapt partial-view features toward the global feature space.

Although our training supervision is constructed from matched partial/global features, those global targets were themselves produced in the context of the complete reconstructed scene. Aligning

Table 1: **Representation gap on ScanNet.** A frozen Sonata encoder with a global linear probe degrades under geometric reduction in world coordinates (-5 mIoU), and significantly under camera-coordinate shift (-70 mIoU). Training-free canonicalization recovers part, but not all, of this loss.

Coord. frame	Input type	allAcc	mAcc	mIoU
World	Global	89.73	83.27	**72.47**
	Room Slice	88.48	81.01	69.64
	View Crop	87.24	80.79	69.02
	Single Frame	87.36	78.98	**67.45**
Camera	Global	26.42	9.56	**5.14**
	View Crop	19.92	7.52	3.00
	Single Frame	18.30	6.94	**2.57**
	+ PCA align	28.39	14.70	8.38
	+ Gravity align (Jin et al., 2023)	70.85	54.74	41.64



to them can therefore transfer information about the structure learned in globally contextualized representations, which is particularly useful for close-range views where visually similar planar regions may otherwise be ambiguous.

4 METHODOLOGY

Let f be a frozen 3D encoder pretrained on globally reconstructed scenes, producing per-point features in $\mathbb{R}^d$, and let $W \in \mathbb{R}^{C\times d}$ be the frozen linear probe trained on world-coordinate global scenes for semantic segmentation.

Our goal is to adapt features extracted from partial camera-coordinate observations so that they become compatible with the global feature space learned by f and the frozen classifier W, without updating the backbone or requiring semantic labels for the target partial views. Adaptation is driven entirely by correspondences to the pretrained global feature space rather than by class-specific supervision. At inference, the model receives only a partial camera-coordinate point cloud and requires neither the complete scene nor camera pose.

Limitations of Supervised Adaptation. Supervised adaptation is effective in-domain but carries costs: (i) full fine-tuning updates the entire encoder per target setting, requiring dense partial-view labels and risking corruption of the global relational structure learned during pretraining (ii) label-supervised PEFT (e.g., GEM) is parameter-efficient but inherits class taxonomy (iii) pointwise feature regression is label-free but constrains each adapted feature only relative to its own target, leaving inter-feature relationships unconstrained. PAGER requires no semantic labels, treats every point without class assumptions, and constrains inter-feature structure rather than only pointwise targets.

4.1 PAGER

We design ϕ as a lightweight geometry-aware adapter model inserted into the frozen backbone (Fig. 3). It refines per-point features through a residual bottleneck whose middle layer ag-

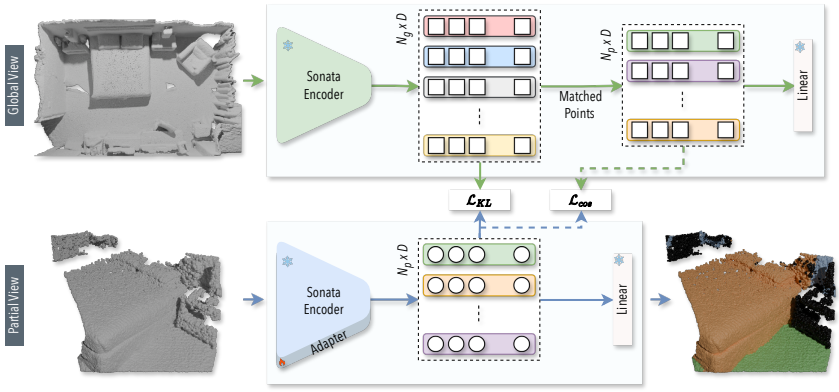


Figure 3: **PAGER architecture.** Only the adaptation layers on the partial branch are trained. For matched points, an anchor $\mathcal{L}_{\cos}$ (Eq. 2) and a relational loss $\mathcal{L}_{\mathrm{KL}}$ (Eq. 4) are used. Encoder and linear probe remain frozen. At inference, only the partial branch is used.

gregates over a local 3D voxel neighbourhood:

$$\phi(x) = x + \sigma\left(\sum_{i\in\mathcal{N}} W^i_{\text{loc}}(xW_{\text{down}})\right) W_{\text{up}}, \tag{1}$$

where $W_{\text{down}} \in \mathbb{R}^{d\times r}, W_{\text{up}} \in \mathbb{R}^{r\times d}$, and $r \ll d$. The local neighbourhood $\mathcal{N}$ is defined in voxel space, allowing the adapter to aggregate spatially nearby information while remaining parameter-efficient. We adopt a GEM-style spatial adapter (Tang et al., 2025) because its local-aggregation prior is well matched to correcting the coordinate-frame errors identified in Sec. 3. We deliberately do *not* include a global-attention module: scene-level information is supplied through the loss, not the architecture. We justify this choice formally in Sec. 4.3 and empirically in Tab. 4.

4.2 Training Objective

Local anchor: cosine alignment at matched points. For each matched pair $(i, j) \in \mathcal{A}$, we align the final partial-view feature z_i^{part} with its corresponding global feature z_j^{glob}:

$$\mathcal{L}_{\cos} = \frac{1}{|\mathcal{A}|}\sum_{(i,j)\in\mathcal{A}} \left(1 - \cos\left(z_i^{\text{part}}, z_j^{\text{glob}}\right)\right). \tag{2}$$

This term provides a direct local alignment signal. It encourages each adapted partial-view feature to match the global representation of the same physical point, which is already compatible with the frozen linear probe.

Relational shaper: point-specific similarity supervision. Let $K \in \mathbb{R}^{M\times d}$ be M keys drawn at random from global scene points. Both branches compute similarity distributions over K:

$$s_i^{\text{part}} = z_i^{\text{part}} K^\top/\tau_s, \qquad s_j^{\text{glob}} = z_j^{\text{glob}} K^\top/\tau_t, \tag{3}$$

with teacher temperature $\tau_t < \tau_s$ so that the teacher is sharper, following the centering-and-sharpening practice of DINO (Caron et al., 2021). The relational objective minimizes the discrepancy between the two distributions. We instantiate it with KL divergence:

$$\mathcal{L}_{\text{KL}} = \frac{1}{|A|}\sum_{(i,j)\in A} D_{\text{KL}}\left(\text{softmax}(s_j^{\text{glob}}) \,\|\, \text{softmax}(s_i^{\text{part}})\right). \tag{4}$$

Direct regression of the sampled similarities is an equally valid parameterization of the same constraint. It attains higher mIoU and allAcc, while the KL formulation attains higher mAcc on both backbones (App. A.4); we report KL as our default and treat the point-specific relational supervision, rather than any particular divergence, as the contribution. The contribution is the point-specific relational supervision itself rather than its specific instantiation. Because keys are random samples across the scene, softmax(s_j^{glob}) is a compact signature of how z_j^{glob} relates to the rest of the scene. Matching it forces the student to reproduce that relational signature from partial observation.

4.3 Why the two losses are complementary

The cosine loss anchors each partial feature to its global target but says nothing about how features relate to each other. The relational loss fills this gap: by matching similarity patterns over shared keys, it constrains inter-feature structure rather than just absolute positions. Tab. 4 (A1) shows the two provide complementary gains.

Why random keys? An alternative is to sample keys from the anchor's neighbourhood (e.g., k-NN in world coordinates), but this is counterproductive: spatially adjacent points lie on similar surfaces and produce highly correlated inner products with the anchor. Random sampling avoids this by drawing keys broadly across the scene, while remaining O(M) and trivially parallelizable, in contrast to alternatives such as farthest-point sampling whose cost scales poorly with scene size. Tab. 4 (A2) confirms the predicted drop when k-NN keys replace random keys.

Table 2: **Semantic segmentation under partial-view adaptation.** Results across two pretrained 3D backbones. Bold and underline denote best and second-best non-upper-bound results within each backbone.

				ScanNet			ScanNet++			ScanNet200		
Method	**Family**	**Superv.**	**Params**	mIoU ↑	mAcc ↑	allAcc ↑	mIoU ↑	mAcc ↑	allAcc ↑	mIoU ↑	mAcc ↑	allAcc ↑
Sonata (Wu et al., 2025)												
Frozen	frozen	none	–	2.57	6.94	18.30	0.69	1.92	9.78	0.14	0.37	3.11
Linear	fine-tune	sem.	<0.1M	36.17	52.42	68.64	15.73	26.11	67.66	8.13	12.98	61.43
Late blocks + linear	fine-tune	sem.	21.0M	48.28	59.13	76.86	17.77	25.33	72.98	11.43	16.56	63.55
Full + linear†	fine-tune	sem.	108.5M	69.26	77.11	87.88	35.99	44.75	85.95	24.58	32.13	77.86
Adapter (Houlsby et al., 2019)	adapt	sem.	6.0M	41.49	65.68	74.22	13.96	28.97	65.58	10.15	16.17	64.49
GEM (Tang et al., 2025)	adapt	sem.	1.8M	<u>59.81</u>	<u>73.92</u>	<u>83.90</u>	**33.27**	**42.57**	**84.22**	**21.97**	**32.52**	**75.45**
PAGER **(ours)**	adapt	**geom.**	1.3M	**65.46**	**80.50**	**85.93**	<u>27.38</u>	<u>39.18</u>	<u>81.43</u>	<u>19.25</u>	<u>27.73</u>	<u>74.19</u>
Concerto (Zhang et al., 2025)												
Frozen	frozen	none	–	4.97	10.83	21.55	1.44	4.26	17.46	0.37	1.64	5.20
Linear	fine-tune	sem.	<0.1M	39.53	56.57	71.10	18.13	26.37	71.69	12.68	19.94	65.89
GEM (Tang et al., 2025)	adapt	sem.	1.8M	<u>69.90</u>	<u>80.96</u>	<u>88.64</u>	<u>32.86</u>	<u>45.28</u>	<u>84.12</u>	<u>25.06</u>	<u>35.10</u>	**77.83**
PAGER **(ours)**	adapt	**geom.**	1.3M	**71.67**	**87.77**	**89.96**	**33.97**	**46.28**	**84.92**	**26.73**	**50.19**	<u>76.12</u>

† Fully supervised full-backbone upper bound, excluded from bold/underline ranking.
none: frozen encoder with global linear probe. *sem.*: partial-view semantic labels. *geom.*: paired partial/global geometry only, no semantic labels.

5 EXPERIMENTS

We report results on two pretrained encoders, Sonata (Wu et al., 2025) and Concerto (Zhang et al., 2025), and use Sonata for the ablation studies. An additional frozen-transfer comparison with PTv3 (Wu et al., 2024) is in App. A.5.

Datasets. We use ScanNet (Dai et al., 2017), ScanNet200 (Rozenberszki et al., 2022), and ScanNet++ (Yeshwanth et al., 2023). ScanNet contains 1513 RGB-D scans with 20 and 200 class point-level annotations. Each frame is back-projected using its captured camera pose, so the evaluation set spans ScanNet's natural pose distribution(App. A.1). We sample every 100th frame for training ($\sim$0.3 FPS, $\approx$19K frames) and every 30^{th} frame for evaluation ($\sim$1 FPS, $\approx$18K frames). ScanNet++ provides high-fidelity scans with 100 classes. We sample every 60^{th} frame for evaluation ($\sim$1 FPS, $\approx$7K frames). All methods use the same subsets for fair comparison. Full statistics in App. C.3.

Baselines. We compare against methods with increasing label and parameter cost. We evaluate two PEFT baselines: *Adapter* (Houlsby et al., 2019), which insert small trainable modules before the classifier, and GEM (Tang et al., 2025), a geometry-aware PEFT method that adds spatial and context adapters to the frozen encoder and trains them with supervised cross-entropy. We also include *Sonata (linear)*, which trains a linear classifier on frozen partial-view features, the standard protocol for evaluating pretrained 3D representations (Wu et al., 2025). *Sonata (late blocks + linear)*, which additionally unfreezes the final encoder blocks and *Sonata (full + linear)*, which updates the entire encoder. All baselines require semantic labels, while PAGER does not.

Metrics. We report mean IoU, mean accuracy, and overall accuracy.
Additional implementation details are provided in App. B.2.

5.1 RESULTS

In-domain adaptation. On ScanNet, PAGER outperforms label-supervised PEFT on both encoders without partial-view labels (Tab. 2), reaching 65.46 mIoU with Sonata and 71.67 with Concerto versus 59.81 and 69.90 for GEM (Tang et al., 2025). Linear probing and adapter-based PEFT (Houlsby et al., 2019) fail to recover (36.17–41.49 mIoU). The gains are larger in mAcc (80.50 vs. 73.92 on Sonata, 87.77 vs. 80.96 on Concerto), and on Sonata PAGER closes most of the remaining gap to full fine-tuning (65.46 vs. 69.26 mIoU) with only 1.2% of the trainable parameters. On the finer-grained taxonomies, PAGER outperforms GEM with Concerto on both ScanNet++ (33.97 vs. 32.86) and ScanNet200 (26.73 vs. 25.06). With Sonata it trails GEM (27.38 vs. 33.27 and 19.25 vs. 21.97) but remains above all other supervised PEFT baselines. Qualitative results are shown in Fig. 4 and per-class results in App. A.3.

Backbone generality. Sonata (Wu et al., 2025) and Concerto (Zhang et al., 2025) differ substantially in pretraining: point-only self-supervision versus joint 2D-3D self-supervision. Yet both collapse on partial camera-coordinate input (2.57 and 4.97 mIoU, Tab. 2), indicating that the representation gap identified in Sec. 3 is a property of global pretraining rather than of a particular encoder. PAGER uses the same adapter and objectives for both and requires only dense per-point features, with no encoder-specific components. That the stronger Concerto representation yields consistent gains over

supervised GEM across all three benchmarks suggests PAGER benefits directly from improvements in the underlying pretrained encoder.

Cross-domain evaluation. To assess robustness to distribution shifts, we train PAGER and all baselines exclusively on ScanNet, using only ScanNet partial/global geometry pairs for PAGER's self-distillation, and evaluate on ScanNet++ without further adaptation. No ScanNet++ scenes, geometry, or labels are seen during training. We map predictions to the 20 classes used for training. As shown in Tab. 3, PAGER learns features that are more resilient to domain changes, achieving higher performance when training and testing data come from different distributions. Full fine-tuning is commonly treated as an upper bound, and on ScanNet Partial it is. That bound, however, is local to the source domain: it costs 108.5M updated parameters and dense partial-view labels per setting. Under distribution shift to ScanNet++, this specialisation becomes a liability, the fully fine-tuned model loses 21 mIoU relative to its in-domain score and falls behind PAGER (53.93 vs. 48.09 mIoU), which never modified the backbone.

Table 3: **Cross-domain evaluation.** Models are trained on ScanNet partial views and evaluated on ScanNet++ partial views. PAGER outperforms all baselines, including *Sonata (full + linear)*, indicating that stronger in-domain adaptation does not necessarily yield better cross-dataset transfer.

Train: ScanNet-Partial (train)→ **Eval:** ScanNet++-Partial (val)			
Method	**mIoU**	**mAcc**	**allAcc**
Sonata (Wu et al., 2025) (full + linear)	48.09	55.10	85.57
Sonata (+ linear)	26.41	35.10	72.32
Sonata (late blocks + linear)	34.15	42.64	78.78
Adapter (Houlsby et al., 2019)	30.69	47.95	74.93
GEM (Tang et al., 2025) (+ linear)	<u>46.10</u>	<u>56.17</u>	<u>84.19</u>
PAGER **(ours)**	**53.93**	**66.64**	**89.22**

5.2 Analysis Experiments

A1: The two losses are complementary. The cosine anchor and relational distillation target distinct aspects of the partial-to-global gap (Sec. 4.3). Tab. 4 (A1) confirms this empirically: $\mathcal{L}_{\text{cos}}$ alone reaches 63.31 mIoU on ScanNet Partial, already strong, since matched-point supervision transfers local class identity well, while $\mathcal{L}_{\text{KL}}$ alone reaches 55.58 mIoU, consistent with the prediction that relational signatures are insufficient when the absolute class direction is unconstrained. Combining the two lifts performance to 65.46 mIoU ($+2.15$ over cosine-only, $+9.88$ over KL-only), confirming the two terms capture different aspects of the alignment.

A2: Random keys outperform k-NN keys. Tab. 4 (A2) shows the predicted direction: replacing random sampling with k-NN-based key selection causes a substantial drop from 65.46 to 49.71 mIoU on ScanNet Partial, in line with the intuition that broadly sampled keys provide a more informative target distribution than spatially clustered ones.

A3: Number of keys. The key set size M controls the breadth of the relational reference: too few keys cannot span the scene, too many are redundant. Performance is stable across two orders of magnitude, varying by under 1.5 mIoU, and we adopt $M = 4096$ as the best value found in this sweep rather than a sharply defined optimum. Since keys are sampled at random, the cost is linear in M and independent of scene size.

Comparison with D-DITR. We additionally compare with Distilled DINO In The Room (D-DITR) (Knaebel et al., 2026), which distills 2D vision foundation model features into a 3D backbone for semantic segmentation (Tab. 5). We apply publicly available D-DITR checkpoints trained on ScanNet-20 and ScanNet-200 directly to partial-view inputs, using both original ScanNet partial scenes and their RGB-D reconstructed counterparts (2.5D). Since D-DITR was not designed for camera-coordinate partial-view inference, the large gap in (Tab. 5) primarily reflects this distribution shift rather than a direct algorithmic comparison. We include it to show that 2D-to-3D feature distillation in global coordinates does not transfer trivially to our partial-view setting.

5.3 Limitations.

PAGER requires paired partial/global observations during training, which are available for RGB-D datasets with camera poses but not universally. Still, it performs well in zero-shot scenarios. It inherits the frozen probe's label taxonomy, so new vocabularies require label remapping or a new

Table 4: **Ablation studies.** mIoU on ScanNet Partial (in-domain) and ScanNet++ Partial (zero-shot cross-eval). All models are trained on ScanNet Partial. Defaults in bold. M values correspond to 1.4%, 2.8%, and 5.6% of global scene points.

A1 Loss terms			**A2** Key sampling			**A3** Number of keys M		
	ScanNet	ScanNet++		ScanNet	ScanNet++		ScanNet	ScanNet++
Local (Eq. 2)	63.31	51.50	k-NN keys	49.71	35.76	2048	64.01	50.80
Relational (Eq. 4)	55.58	44.73	**random**	**65.46**	**53.93**	**4096**	**65.46**	**53.93**
Both	**65.46**	**53.93**				8192	64.63	53.19

Table 5: **Evaluation with D-DITR.** We evaluate both ScanNet Partial and its RGB-D reconstruction (ScanNet Partial-2.5D), using D-DITR (Knaebel et al., 2026). We report results using two publicly available D-DITR variants trained on ScanNet-20 and ScanNet-200 labels respectively.

Method	**ScanNet-20** (val)		**ScanNet-200** (val)	
	Partial mIoU	**2.5D** mIoU	**Partial** mIoU	**2.5D** mIoU
D-DITR (Knaebel et al., 2026)	1.70	1.66	0.29	0.24
PAGER	(+63.76) **65.46**	(+62.22) **63.88**	(+18.96) **19.25**	(+16.88) **17.12**

probe, although the adapted encoder can be reused once new labels are available. And it remains below in-domain *Sonata (full + linear)* (65.46 vs. 69.26 mIoU on ScanNet), but closes most of the gap with only 1.2% of the trainable parameters and outperforms in cross-dataset transfer.

6 Conclusion

We presented PAGER, a label-free framework that aligns partial camera-coordinate 3D features with a frozen global semantic space using only paired geometry and no semantic labels. Our analysis isolates coordinate-frame mismatch as the dominant source of degradation in frozen 3D transfer, and show that geometric self-distillation, combining a matched-point cosine anchor with a point-specific relational objective, sufficiently repairs this mismatch without modifying the backbone. PAGER outperforms label-supervised PEFT in-domain and surpasses fully fine-tuned Sonata in zero-shot cross-dataset transfer, using only 1.3M trainable parameters across two distinct pretrained encoders.

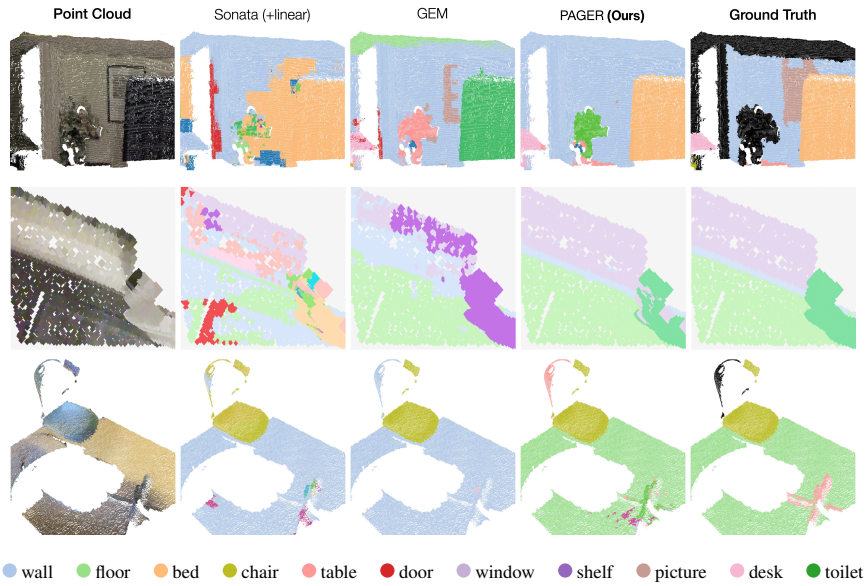


Figure 4: **Qualitative results on ScanNet Partial.** PAGER produces predictions closer to ground truth than the frozen Sonata baseline with a trained linear probe and label-supervised GEM, especially under severe FoV limitations and camera-coordinate inputs. Baselines often confuse visually similar planar surfaces: the mattress (1st *row*) is misclassified as toilet rather than bed, while the floor (3rd *row*) is mistaken for wall. PAGER better maintains object semantics through alignment to the global feature space and also produces plausible labels for unannotated points shown in black.

REPRODUCIBILITY STATEMENT

To ensure reproducibility, we will release our code upon publication. Our experiments use publicly available datasets and resources. Detailed descriptions of the datasets, preprocessing, training settings are provided in App. B and App. C. All evaluation benchmarks utilized in this work are established and widely used within the field. We meticulously detail every aspect of our experimental methodology to ensure faithful reproduction of our work.

AUTHOR STATEMENT ON THE USE OF LLMS

In this work, we used generative AI tools for writing refinement and coding assistance. They were not used for research ideation, experimental design, analysis, or scientific content. All AI-generated code was reviewed and verified by the authors before use. All research contributions are solely the responsibility of the authors.

REFERENCES

Gwangbin Bae and Andrew J. Davison. Rethinking inductive biases for surface normal estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Kristen Grauman, Alex Westbury, Jitendra Malik, Christoph Feichtenhofer, Kris Kitani, Deva Ramanan, Lorenzo Torresani, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Kristen Grauman, Kris Kitani, Deva Ramanan, Christoph Feichtenhofer, Lorenzo Torresani, et al. Ego-exo4d: Understanding skilled human activity from first- and third-person perspectives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*. PMLR, 2019.

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.

Linyi Jin, Jianming Zhang, Yannick Hold-Geoffroy, Oliver Wang, Kevin Blackburn-Matzen, Matthew Sticha, and David F. Fouhey. Perspective fields for single image camera calibration. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17307–17316, 2023. doi: 10.1109/CVPR52729.2023.01660.

Karim Knaebel, Kadir Yilmaz, Daan de Geus, Alexander Hermans, David Adrian, Timm Linder, and Bastian Leibe. DINO in the room: Leveraging 2D foundation models for 3D segmentation. In *2026 International Conference on 3D Vision (3DV)*, 2026.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4582–4597, 2021.

Romy M. Luo, Sherry Z. Xue, Alex G. Dimakis, and Kristen Grauman. Viewpoint rosetta stone: Unlocking unpaired ego-exo videos for view-invariant representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Slavcho Neshev, Krasimir Tonchev, Agata Manolova, and Vladimir Poulkov. 3d scene segmentation: A comprehensive survey and open problems. *IEEE Access*, 13:110457–110496, 2025. doi: 10.1109/ACCESS.2025.3583136.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2017.

David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3d semantic segmentation in the wild. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

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

Liyao Tang, Zhe Chen, and Dacheng Tao. On geometry-enhanced parameter-efficient fine-tuning for 3d scene segmentation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2025.

Chengyao Wang, Li Jiang, Xiaoyang Wu, Zhuotao Tian, Bohao Peng, Hengshuang Zhao, and Jiaya Jia. Groupcontrast: Semantic-aware self-supervised representation learning for 3d understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024a.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025a.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024b.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. Permutation-equivariant visual geometry learning. In *arXiv preprint arXiv:2507.13347*, 2025b.

Xiaoyang Wu, Xin Wen, Xihui Liu, and Hengshuang Zhao. Masked scene contrast: A scalable framework for unsupervised 3d representation learning. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition (CVPR)*, 2023.

Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler faster stronger. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2024.

Xiaoyang Wu, Daniel DeTone, Duncan Frost, Tianwei Shen, Chris Xie, Nan Yang, Jakob Engel, Richard Newcombe, Hengshuang Zhao, and Julian Straub. Sonata: Self-supervised learning of reliable point representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2025.

Aoran Xiao, Jiaxing Huang, Dayan Guan, Xiaoqin Zhang, Shijian Lu, and Ling Shao. Unsupervised point cloud representation learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):11321–11339, 2023. doi: 10.1109/TPAMI.2023.3262786.

Saining Xie, Jiatao Gu, Demi Guo, Charles R Qi, Leonidas Guibas, and Or Litany. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In *European conference on computer vision (ECCV)*, 2020.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2023.

Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 1–9, 2022.

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

Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision (CVPR)*, 2021.

Technical appendices and supplementary material

A Additional Results

A.1 Robustness across camera pose

Every frame in our main experiments is back-projected using its own captured camera-to-world pose, so the 19K training and 18K evaluation frames already span ScanNet's natural distribution of handheld pitch, roll, and height. (The single-transform-to-first-frame setting appears only as the Camera—Global diagnostic in Tab. 1 and is not used in any main experiment.)

To characterize behaviour across this distribution, we bin evaluation frames by camera pitch, computed from each frame's known rotation, and report PAGER against supervised GEM (Tang et al., 2025) per bin (Tab. 6).

Table 6: **Partial-view segmentation by camera pitch.** Frames are binned by pitch computed from each frame's captured pose. The first three bins account for 98.9% of the evaluation set.

Pitch range	Frames	PAGER	GEM
Steep down ($< -30^\circ$)	5,567	61.31	52.52
Down (-30° to -10°)	9,875	65.82	60.89
Level (-10° to 10°)	2,183	61.44	52.45
Up (10° to 30°)	166	34.63	24.55
Steep up ($> 30^\circ$)	30	18.38	18.95

PAGER maintains 61–66 mIoU across the steep-down, downward, and level bins, which together constitute 98.9% of the evaluation set, and outperforms supervised GEM by 5–9 mIoU in each. It also improves substantially over GEM on upward views between 10° and 30°. Both methods degrade on the steep-up bin, which contains only 30 frames; the shared degradation suggests this reflects ScanNet's limited coverage of such viewpoints rather than a failure specific to our adaptation. Targeted augmentation for underrepresented orientations is a natural direction for future work.

A.2 Application: In-the-wild segmentation.

Reconstructed input without camera intrinsics. A practical use case for our local semantic segmentation framework is compatibility with feed-forward 3D reconstruction models such as (Wang et al., 2025a;b), which infer 3D scene attributes, including camera parameters, point maps, depth maps, and tracks, from a single RGB image. Here, we use only the inferred point maps from ScanNet images as input. As shown in Fig. 5 (left), PAGER correctly identifies the *door* from this limited context, without access to ground-truth depth. By contrast, the *Sonata (linear)* and GEM baselines rely mainly on local planar geometry and misclassify the region as *wall*.

Stereo input on unseen scenes. We also evaluate stereo imagery from a mobile Husky platform in unseen environments. Depth from stereo pairs is projected into the robot coordinate frame using known camera intrinsics. There is no ground truth but PAGER assigns plausible semantic labels to these out-of-distribution partial point clouds, producing coherent segmentation masks (Fig. 5 (right)). Qualitative results were consistent across the 183 recorded frames. **Implementations Details** in the Appendix C.2

A.3 Per class analysis.

1. Per class semantic segmentation on ScanNet-Partial (val). Tab. 7 reports per-class IoU on ScanNet-Partial. PAGER improves over GEM on most classes, especially shower curtain, curtain, bathtub, sink, and door, while GEM remains better on sofa and picture.

2. Per class cross evaluation on ScanNet++-Partial (val). Tab. 8 reports per-class IoU for models trained on ScanNet-Partial and evaluated on ScanNet++-Partial. PAGER improves over GEM on most classes, with large gains on bed, door, refrigerator, bookshelf, curtain, sink, and other furniture. GEM remains better on toilet, picture, and window. Overall, the results show that label-free

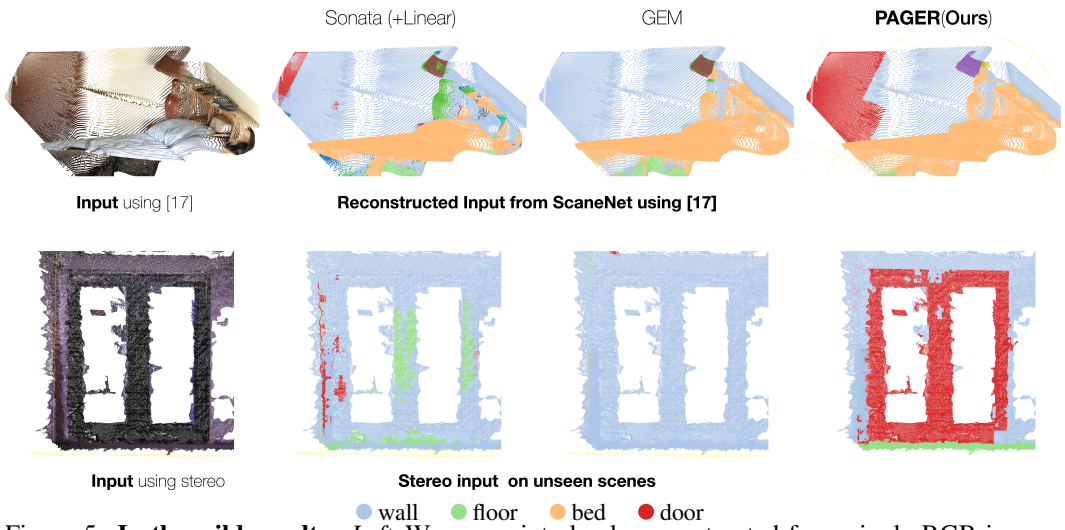


Figure 5: **In-the-wild results.** *Left:* We use point clouds reconstructed from single RGB images with (Wang et al., 2025a) as input. *Sonata (linear)* and GEM mislabel flat regions as wall, while PAGER, recovers the door. *Right:* On stereo point clouds from a mobile robot in unseen environments, PAGER produces coherent segmentations despite the distribution shift, correctly recognizing the door.

Table 7: **Per class IoU on ScanNet-Partial (val)** for PAGER trained on ScanNet-Partial. We report IoU for 20 semantic classes of ScanNet.

Model	wall	floor	cabinet	bed	chair	sofa	table	door	window	bookshelf	picture	counter	desk	curtain	refrigerator	shower curtain	toilet	sink	bathtub	otherfurniture
Sonata (full + linear)	80.52	**94.47**	**60.73**	**81.46**	**86.28**	**79.75**	**75.21**	59.03	52.28	**70.75**	32.08	**59.79**	**65.80**	68.85	**55.96**	60.83	**94.18**	**70.28**	**89.73**	**47.14**
Sonata (late blocks + linear)	66.05	84.67	43.32	64.19	68.22	62.79	62.21	33.72	29.78	50.01	18.52	36.21	44.87	37.95	34.81	38.94	67.46	42.73	61.04	18.03
Sonata (+linear)	62.15	83.24	28.38	45.68	52.16	42.67	49.71	12.33	25.71	36.28	24.85	27.43	29.09	33.64	19.74	34.36	44.71	30.75	30.79	9.68
GEM	76.82	93.19	51.45	72.95	79.54	72.86	66.44	51.06	49.59	64.47	**36.47**	53.66	49.72	57.94	48.31	48.48	74.34	49.39	64.25	35.32
PAGER	**80.94**	93.79	55.32	75.45	81.17	71.90	70.37	**62.07**	**57.58**	64.83	32.93	55.69	55.72	**74.58**	49.96	**68.89**	82.35	61.02	76.78	37.85
Δ PAGER vs GEM	+4.12	+0.60	+3.87	+2.50	+1.63	-0.96	+3.93	+11.01	+7.99	+0.36	-3.54	+2.03	+6.00	+16.64	+1.65	+20.41	+8.01	+11.63	+12.53	+2.53

partial-to-global alignment also improves per-class transfer under dataset shift. Overall, PAGER outperforms GEM in almost all classes, with gains of up to +28.11 IoU points.

Table 8: **Per class cross evaluation (IoU) on ScanNet++-Partial (val).** for PAGER trained on ScanNet-Partial for our cross-domain setup. We report IoU for 20 semantic classes of ScanNet.

Model	wall	floor	cabinet	bed	chair	sofa	table	door	window	bookshelf	picture	counter	desk	curtain	refrigerator	shower curtain	toilet	sink	bathtub	otherfurniture
Sonata (full + linear)	82.84	93.72	48.28	59.02	**85.62**	69.63	59.73	45.26	41.06	33.85	29.68	**53.54**	0.00	66.97	35.09	0.00	**78.61**	46.20	0.00	32.82
Sonata (late blocks + linear)	75.57	87.73	31.41	34.75	70.10	54.37	48.25	28.07	28.13	25.90	14.00	29.95	0.00	27.31	22.98	0.00	54.22	34.95	0.00	15.24
Sonata (+linear)	68.12	78.55	19.28	16.33	46.71	35.30	43.07	7.75	22.44	12.23	23.55	21.32	0.00	33.92	9.34	0.00	40.71	30.98	0.00	18.54
GEM	82.00	91.44	51.69	39.25	73.95	64.86	58.21	40.98	**48.08**	43.46	**36.25**	44.84	0.00	63.86	26.84	0.00	74.34	54.59	0.00	27.29
PAGER	**88.32**	**96.14**	**58.84**	**67.36**	83.48	**72.71**	**60.08**	**67.61**	46.18	**58.48**	33.12	47.41	0.00	**77.75**	**43.75**	0.00	69.34	**65.62**	0.00	**42.33**
Δ PAGER vs GEM	+6.32	+4.70	+7.15	+28.11	+9.53	+7.85	+1.87	+26.63	-1.90	+15.02	-3.13	+2.57	+0.00	+13.89	+16.91	+0.00	-5.00	+11.03	+0.00	+15.04

3. Per class semantic segmentation on ScanNet200-Partial (val). Tab. 9 reports representative per-class IoU on ScanNet200-Partial. For readability, we show the 10 classes with the largest PAGER gains over GEM and the 10 classes with the largest GEM gains over PAGER. PAGER performs better on several structural and mid-sized categories, while GEM is stronger on some fine-grained or rare object classes.

4. Per class semantic segmentation on ScanNet++-Partial (val). Tab. 10 reports representative per-class IoU on ScanNet++-Partial. For readability, we show the 10 classes with the largest PAGER gains over GEM and the 10 classes with the largest GEM gains over PAGER. PAGER improves several object and furniture categories, including coat hanger, blanket, refrigerator, bed, shower wall, towel, kitchen cabinet, picture, printer and poster. GEM remains stronger on some fine-grained object categories, such as tap, book, whiteboard eraser, shoes, mouse, speaker, laptop, computer tower, keyboard and smoke detector.

Table 9: **Per class IoU on ScanNet200-Partial (val).** We report 20 representative semantic classes: 10 where PAGER outperforms GEM and 10 where GEM outperforms PAGER.

Model	fireplace	light	decoration	clothes dryer	stairs	windowsill	hair dryer	tv stand	curtain	end table	backpack	paper cutter	dishwasher	coffee maker	coffee kettle	ceiling	shoe	scale	dish rack	blackboard
Sonata (full + linear)	10.67	6.17	0.81	**36.62**	48.93	9.24	0.00	25.92	61.96	**33.03**	44.14	**28.01**	**50.87**	**58.68**	**52.89**	**77.95**	34.57	**47.94**	**66.60**	**71.74**
Sonata (late blocks + linear)	0.01	2.11	0.73	1.39	21.48	6.10	0.00	25.10	37.06	11.13	13.38	3.32	5.97	13.53	0.00	43.95	8.01	0.80	18.15	13.52
Sonata (+ linear)	0.00	1.11	3.95	0.03	8.61	9.92	0.00	14.21	30.50	0.88	19.72	10.60	0.35	12.50	5.31	49.95	23.46	0.00	2.76	7.63
GEM	3.54	8.19	9.25	5.99	37.74	9.52	1.17	33.01	60.00	16.27	**44.24**	27.62	25.64	34.30	28.82	73.73	**44.89**	34.21	50.37	62.84
PAGER	**52.86**	**36.76**	**32.86**	25.21	**54.66**	**25.64**	**14.24**	**44.86**	**71.16**	26.18	26.36	8.71	4.88	11.43	2.44	47.25	18.12	5.02	18.76	25.54
Δ PAGER vs GEM	+49.32	+28.57	+23.61	+19.22	+16.92	+16.12	+13.07	+11.85	+11.16	+9.91	-17.88	-18.91	-20.76	-22.87	-26.38	-26.48	-26.77	-29.19	-31.61	-37.30

Table 10: **Per class IoU on ScanNet++-Partial (val).** We report IoU for 20 out of 100 semantic classes for visualisation purposes. 10 where PAGER outperforms GEM and 10 where GEM outperforms PAGER

Model	coat hanger	blanket	refrigerator	bed	shower wall	towel	kitchen cabinet	picture	printer	poster	tap	book	whiteboard eraser	shoes	mouse	speaker	laptop	computer tower	keyboard	smoke detector
Sonata (full + linear)	2.64	33.21	35.83	44.03	**46.10**	31.91	45.91	24.98	**36.35**	0.51	31.31	**43.11**	**39.96**	**50.27**	**44.76**	20.22	**50.35**	**78.00**	**73.77**	**52.61**
Sonata (late blocks + linear)	1.66	29.41	9.63	32.58	18.97	13.40	32.36	13.36	14.49	1.36	8.62	18.15	8.56	21.96	5.62	2.26	2.10	37.62	25.29	7.21
Sonata (+ linear)	3.02	37.07	2.02	20.71	4.87	13.70	16.31	19.78	9.97	5.01	16.28	23.45	7.34	16.25	12.39	3.51	8.60	21.74	34.20	17.15
GEM	3.83	37.24	30.36	39.57	32.70	32.61	45.10	32.70	25.31	3.47	**36.06**	38.71	28.70	47.72	35.80	**33.99**	42.90	64.61	65.13	47.56
PAGER	**21.40**	**52.40**	**43.06**	**51.71**	43.31	**42.37**	**53.30**	**40.72**	33.15	**10.73**	9.16	11.02	0.82	19.43	4.86	0.93	9.60	27.25	24.35	0.25
Δ PAGER vs GEM	+17.57	+15.16	+12.70	+12.14	+10.61	+9.76	+8.20	+8.02	+7.84	+7.26	-26.90	-27.69	-27.88	-28.29	-30.94	-33.06	-33.30	-37.36	-40.78	-47.31

Table 11: **Performance comparison across different models and variants.** Results shown for Sonata and Concerto architectures using KL and Sampled Gram methods.

Model	Variant	mIoU	mAcc	AllAcc
Sonata	KL	65.46	**80.50**	85.93
	Sampled Gram	**67.27**	77.75	**87.62**
Concerto	KL	71.67	**87.77**	89.96
	Sampled Gram	**74.14**	85.26	**90.66**

A.4 Relational loss parameterization

The relational objective constrains each partial feature's similarity pattern to a set of global reference keys (Sec. 4.2). Tab. 11 compares two parameterizations of this constraint: KL divergence between softmax distributions, and direct regression of the sampled similarities.

Both improve substantially over the cosine anchor alone, and the choice between them is a trade-off rather than a ranking. Gram regression gives higher mIoU ($+1.81$ on Sonata, $+2.47$ on Concerto) and higher allAcc, while KL gives higher mAcc ($+2.75$ and $+2.51$ respectively). Since mAcc weights all classes equally regardless of frequency, the KL formulation is preferable when per-class balance matters, and Gram regression when overall segmentation quality is the target. We use KL throughout the main paper as our originally validated configuration. That the pattern is consistent across two architecturally distinct encoders supports our claim that the benefit derives from point-specific relational supervision rather than from a specific divergence.

A.5 Choice of encoder

Additionally to our experiments with Sonata and Concerto encoders, we compare the Sonata encoder with PTv3 (Wu et al., 2024), another widely used 3D backbone. Sonata is the strongest pretrained representation in our setting, as shown in Tab. 12. Since PTv3 uses a different decoder, we also evaluate Sonata with the PTv3 decoder for a controlled comparison. Sonata performs best in our *Sonata (frozen)* setup, so we use it for all main experiments.

B Model Details

B.1 Visual Representation of the loss.

For a graphical visualization of the PAGER losses please see Fig.6.

Table 12: **Globally trained** PTv3 vs. Sonata (Decoder) on the partial setup.

Method	mIoU	mAcc	allAcc
PTv3	1.30	5.03	14.28
Sonata (Decoder)	**2.60**	**6.20**	**24.14**

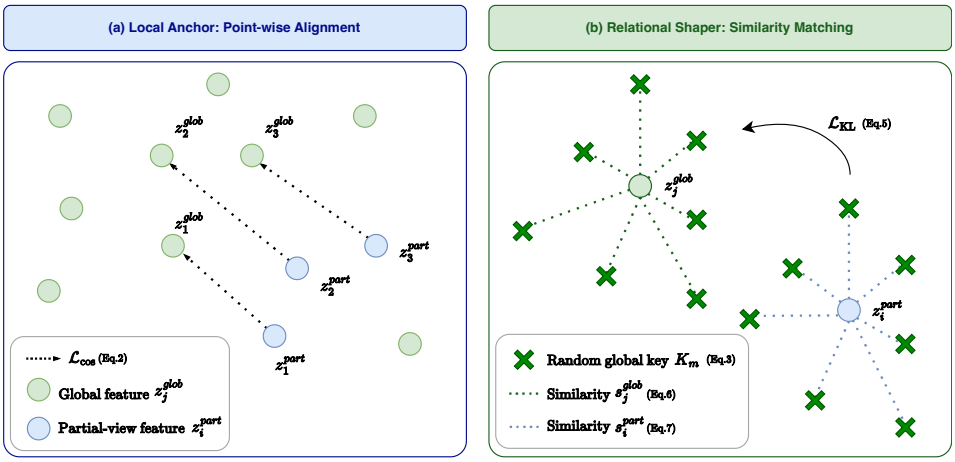


Figure 6: PAGER losses visualization.

B.2 Implementation Details.

Our implementation is based on the open-source Pointcept codebase and follows the official implementations of Sonata (Wu et al., 2025) and GEM (Tang et al., 2025).

For a fair comparison, we train all baselines for 100 epochs, following the GEM training protocol (Tang et al., 2025). We found that all baselines had converged by this point. We also used the same batch size. Details are presented in Table 14.

Software and hardware environment. All experiments used a single NVIDIA H100 PCIe GPU with 80GB memory for training and testing (Tab. 13)

Table 13: **Hardware and software environment** used for all experiments.

Component	**Specification**
Operating system	AlmaLinux release 9.6
CPU	AMD EPYC 9354 32-Core
RAM	768GB
GPU	NVIDIA H100 PCIe 80GB
CUDA version (system)	13.0
Python version	3.12.9
PyTorch version	2.5.1+cu124

C Dataset and Preprocessing Details

For each RGB-D frame, we obtain a depth map D, color image, camera intrinsics K, and camera-to-world pose $T_{\text{cam}\to\text{world}}$. For every valid pixel $p = (u, v)$ with depth D_p, we backproject to 3D in the camera frame,

$$p^{\text{cam}} = D_p \, \mathbf{K}^{-1} \tilde{p}. \tag{5}$$

We discard invalid depths (non-finite, zero, or beyond a maximum range) and transform the remaining points to the global coordinate system using

$$p^{\text{world}} = \mathbf{R}^{\top} \left(D_p \, \mathbf{K}^{-1} \tilde{p} \right) - \mathbf{R}^{\top} \mathbf{t}. \tag{6}$$

The RGB values are sampled at (u, v) from the color image. This produces, for each frame, a dense partial point cloud.

To make partial point clouds consistent with the original ScanNet annotations, we project semantics and normals from the cleaned mesh onto the RGB-D points. For each scene, we load the corresponding mesh and compute vertex normals. Using the dataset's labels we assign each vertex semantic and instance IDs.

We then establish correspondences between mesh vertices and RGB-D points in world coordinates by building a KD-tree over the frame points and querying it with all valid mesh vertices. A mesh vertex and frame point form a pair if their Euclidean distance ≤ 0.01m. For each RGB-D point, we keep only the nearest valid mesh vertex within this threshold. RGB-D points without any valid correspondence are discarded.

For each retained point, we copy the semantic and instance labels as well as the vertex normal from the matched mesh vertex. Normals are rotated into the camera frame as follows

$$n^{\text{cam}} = \mathbf{R}^{\top} n^{\text{world}}. \tag{7}$$

C.1 Correspondences between Partial Point clouds and the Global Mesh

The same world-space coordinates are also use to construct the voxel-level correspondences used in our loss. For each scene, we perform a nearest-neighbor search in the global space and assign each original partial point cloud coordinate to its closest global scene original coordinate.

To ensure these correspondences remain valid after augmentation, we explicitly track how every point moves throughout the entire transform chain. At the beginning of preprocessing, each point is assigned a persistent origin identifier, and after every point-indexing operation we update the mapping that links the current points back to their original positions. During voxelization, we additionally record which original points correspond to each voxel representative, allowing us to propagate origin-level correspondences into voxel-level correspondences that remain aligned with the subsequent augmentations.

C.2 Mobile Robot Data Acquisition

To assess PAGER under real-world conditions, we collect partial point clouds using a Clearpath Husky A200 UGV equipped with a Roboception RC-Visard 160c stereo camera mounted on a rigid front-facing tower. The camera provides synchronized left/right RGB images and onboard SGM-based disparity and depth maps, which we use exclusively.

Camera intrinsics are obtained through Roboception's built-in calibration tool, and the stereo system outputs rectified images, so no additional geometric calibration is required. We back-project the stereo depth into 3D using the calibrated intrinsics to form partial point clouds.

Points beyond 7m are discarded to remove distant background measurements. Surface normals are estimated with (Bae & Davison, 2024).

C.3 Dataset Statistics

Datasets. We employ three widely used indoor 3D datasets: ScanNet (Dai et al., 2017), ScanNet200 (Rozenberszki et al., 2022) and ScanNet++ (Yeshwanth et al., 2023): (1) ScanNet is collected using a lightweight RGB-D scanning setup and comprises 1513 indoor scenes, covering approximately 2M views, with corresponding 3D camera poses, surface reconstructions, and point-level semantic annotations for 20 classes and 200 classes (for the ScanNet200 version respectively). For training all methods we sample every 100th frame from the RGB-D sequences (~ 0.3 FPS) of the training set, resulting in approximately 19K training frames. This higher sampling stride was chosen to ensure training efficiency given the large number of available frames. For evaluation we sample every 30th frame (~ 1 FPS) from the validation set, resulting in about 18K evaluation frames. (2) We also use ScanNet++, which comes with high-fidelity 3D mask annotations, including smaller objects (100 classes). It includes high-resolution 3D scans captured at sub-millimeter precision and annotated comprehensively, covering objects of varying sizes.

Table 14: PAGER training configuration for ScanNet Partial and ScanNet++ Partial.

Configuration	ScanNet-20/200 Partial	ScanNet++ Partial
Optimization		
Optimizer	AdamW	AdamW
Learning rate	2×10^{-5}	2×10^{-5}
LR schedule	Constant	Constant
Weight decay	2×10^{-2}	2×10^{-2}
Gradient clipping	3.0	3.0
Mixed precision	FP16 (AMP)	FP16 (AMP)
Batch size	24	24
Epochs	100	100
Trainable modules	spatial adapter only	spatial adapter only
Backbone, frozen		
Backbone	PT-v3m2, Sonata weights	PT-v3m2, Sonata weights
Input channels	9, xyz + rgb + normal	9, xyz + rgb + normal
Encoder depths	(3, 3, 3, 12, 3)	(3, 3, 3, 12, 3)
Encoder channels	(48, 96, 192, 384, 512)	(48, 96, 192, 384, 512)
Encoder heads	(3, 6, 12, 24, 32)	(3, 6, 12, 24, 32)
Patch size	1024, all stages	1024, all stages
Order	(z, z-trans, hilbert, hilbert-trans)	(z, z-trans, hilbert, hilbert-trans)
Voxel size	0.02 m	0.02 m
Drop path	0.3	0.3
Attention impl.	FlashAttention	FlashAttention
Adapter		
Spatial adapter hidden dim	32	32
Spatial projection	lin-subm3	lin-subm3
Adapter init	LoRA-style	LoRA-style
PAGER loss		
Student temperature τ_s	0.10	0.10
Teacher temperature τ_t	0.04	0.04
Number of relational keys M	4096	16384
Data		
Frame stride, train	100	60
Frame stride, eval	30	60

Table 15: **Dataset statistics.** Number of scenes and average points per scene. Partial variants contain multiple partial reconstructions per scene therefore, we sample /100 for training on ScanNet and /60 for training on ScanNet++. Evaluation follows /30 in ScanNet and /60 in ScanNet++.

Dataset	Src	Scenes			pts/scene		Overall	
		Train	Val	Test	Train	Val	Scenes	Points
ScanNet (Dai et al., 2017)	real	1.2K	312	100	145K	159K	1.6K	242M
ScanNet-Partial	real	19.6K	21.4K	N/A	6.5K	6.6K	40.9K	267M
ScanNet++ (Yeshwanth et al., 2023)	real	856	50	50	1.50M	1.51M	956	1.43B
ScanNet++-Partial	real	38.3K	7.1K	N/A	22.7K	21.6K	12.8K	276M

For evaluation we sample every 60th frame ($\sim$ 1 FPS) from the validation set, resulting in about 7K evaluation frames. All methods are trained and evaluated on the same subsets for fair comparison (see Tab. 15 for details).
For our cross-dataset experiments, we use the ScanNet training protocol, but we evaluate on ScanNet++ validation.

C.4 Broader Impacts

Reliable partial-view 3D segmentation supports applications such as robotics, AR/VR, and assistive navigation, where embodied agents must reason from limited onboard observations. As with any indoor scene understanding system, deployment in human-centred environments raises privacy and monitoring considerations that warrant care in real-world use.