VINO: REVISITING 3D REPRESENTATION LEARNING WITH VOLUME TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Self-supervised representation learning has proven highly effective in 2D, but comparable success has yet to be achieved for 3D scenes. Motivated by this, we identify key limitations in current approaches to 3D representation learning and revisit their design choices. We first replace the commonly used hierarchical PTv3 backbone with the Volume Transformer, a ViT-like architecture for 3D scenes. We then redesign two core objectives of modern approaches built around teacher–student networks: matching representations across views and predicting masked content from visible context. For the former, we separate the roles of teacher and student: the teacher provides clean, consistent targets while the student learns invariance to stronger perturbations. For the latter, we prevent the student from accessing the masked geometry and instead require it to infer missing representations from visible context only. Finally, we redesign 2D-to-3D knowledge distillation by replacing direct regression of view-dependent image features with more view-consistent supervision. Together, these changes form VINO, a representation learning framework that learns semantic representations of 3D scenes without labels. VINO substantially improves dense linear probing across indoor and outdoor benchmarks and transfers strongly to out-of-domain settings.

1 INTRODUCTION

Learning general-purpose representations that transfer across diverse downstream tasks is a central goal of representation learning. Self-supervised learning (SSL) has made remarkable progress toward this goal in computer vision, giving rise to vision foundation models (VFMs) (Zhou et al., 2022; Oquab et al., 2024; Siméoni et al., 2025) that have advanced the state of the art across downstream tasks (Lin et al., 2026; Kerssies et al., 2025; Siméoni et al., 2025). A key ingredient behind this progress is *self-distillation*, in which a student network is trained to predict targets produced by a teacher, which itself is updated as an exponential moving average (EMA) of the student.

Recent SSL methods for 3D scenes (Wu et al., 2025; Abdelsamad et al., 2026; Yang et al., 2026) adapt this self-distillation paradigm to 3D point clouds, but the resulting representations do not yet have the quality observed for 2D images. In 2D, frozen SSL encoders equipped with a linear prediction head can approach the performance of fully supervised models of comparable size (Oquab et al., 2024; Siméoni et al., 2025). In contrast, recent 3D SSL methods still substantially underperform supervised models, despite training on much more data. Motivated by this gap, we first identify and address several limitations in existing 3D self-distillation designs (Fig. 1), and then complement this with redesigned 2D–3D knowledge distillation. Together, this leads to *VINO*, a new framework for *volumetric distillation with no labels* that learns improved semantic 3D representations.

We first reconsider the 3D backbone architecture. Modern self-distillation methods are developed around Vision Transformers (ViTs) (Dosovitskiy et al., 2021), which operate on sequences of image tokens, whereas 3D SSL methods predominantly rely on the hierarchical Point Transformer V3 (PTv3) encoder (Wu et al., 2024), which operates on multi-resolution features. We therefore replace PTv3 with the state-of-the-art *Volume Transformer* (Volt) (Yilmaz et al., 2026), a ViT-like architecture operating directly on volumetric tokens. This change removes several architecture-specific mechanisms required for compatibility with PTv3 and already yields a substantial improvement. Building on this stronger foundation, we next revisit the two core objectives of self-distillation and examine how they should be formulated for sparse 3D scenes.

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.

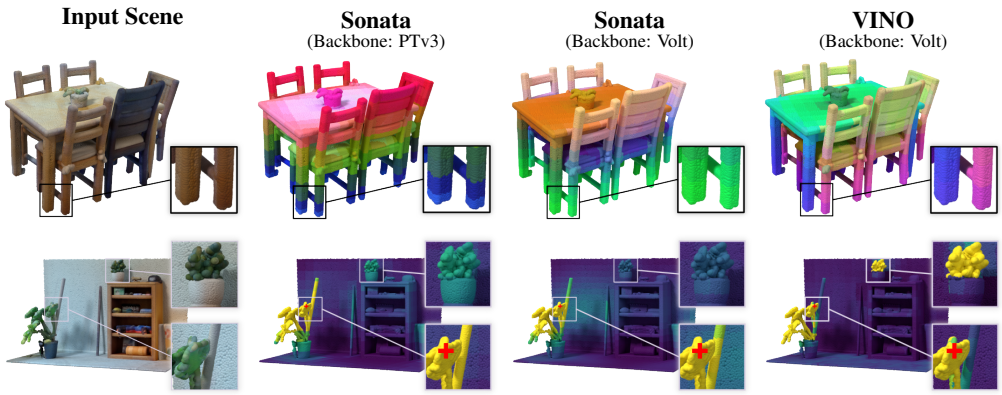

Figure 1: **Self-supervised 3D Representations.** We propose VINO, a framework for learning semantic representations of 3D scenes with no labels. Compared with Sonata (Wu et al., 2025), VINO representations clearly reflect semantic similarity, in both PCA projections *(top)* and cosine-similarity maps *(bottom)*. Sonata yields qualitatively similar representations across backbones, whereas VINO better distinguishes semantically different but geometrically similar regions, for example chair- and table-legs, and captures semantic similarity across geometrically different instances, as illustrated by one plant strongly activating the other.

The first objective trains the student encoder, through a projection head, to match the teacher's output across differently augmented views of the same scene. Existing approaches typically apply the same set of augmentations to both the teacher and student views. However, this exposes the teacher to the same strong perturbations as the student, potentially degrading the quality of its targets. We instead introduce an asymmetric formulation in which the teacher receives clean global views while the student processes strongly augmented local views (see Fig. 2). This asymmetry also changes how the teacher should construct its targets. Conventionally, the teacher's projection head is obtained as an EMA copy of the student's. Under our formulation, however, teacher and student operate on different view distributions, making this coupling undesirable. Inspired by Darcet et al. (2025), we therefore *decouple target construction from student prediction* and learn the teacher's target head independently by enforcing multi-view consistency between teacher views.

We next revisit *masked modeling*, which trains the student encoder, typically through another projection head, to match the teacher's output at masked locations using only the surrounding visible context. Existing 3D methods commonly replace masked voxel features with learned embeddings, while retaining their locations in the sparse grid, mirroring masked-image modeling. In sparse 3D scenes, however, the occupancy of voxels itself encodes geometry. Retaining masked locations therefore reveals the structure that the student should learn to infer (see Fig. 2). Instead, we remove masked tokens entirely before encoding and use a lightweight cross-attention predictor to reconstruct their representations from visible context. We supervise these predictions with the same decoupled targets introduced above, providing a unified supervision signal across both objectives.

Finally, we redesign 2D-to-3D knowledge distillation from pretrained 2D VFMs, which has become an important source of supervision for 3D representation learning (Knaebel et al., 2026; Zhang et al., 2025; 2026). Existing approaches typically train the 3D encoder to regress dense image features, even though the same 3D region can yield different image embeddings when viewed from different viewpoints. This exposes the 3D encoder to inconsistent targets for the same 3D region, encouraging it to learn view-specific variation rather than a consistent semantic representation. To address this, we construct multi-view-consistent targets in the frozen 2D feature space and train the 3D encoder to predict them. This removes view-specific variation from the supervision signal and further improves representation quality (see Fig. 5).

Combined, these four contributions establish VINO as a strong framework for 3D representation learning. With 3D-only pretraining, VINO substantially improves linear probing over prior work by $+5.2$ mIoU on ScanNet200 (Rozenberszki et al., 2022) and $+3.5$ on ScanNet++ (Yeshwanth et al., 2023). Adding 2D supervision yields a further $+3.9$ on ScanNet200, $+6.2$ on ScanNet++, and $+3.0$ on nuScenes (Caesar et al., 2020), achieving the best results across all six benchmarks. The frozen representations also transfer strongly beyond the pretraining setting to zero-shot point-cloud registration and single-view RGB-D semantic segmentation under linear probing.



Figure 2: **Comparison of Training Views.** Sonata (Wu et al., 2025) geometrically corrupts all views and retains masked tokens in the student encoder. In contrast, VINO keeps the teacher's global views clean, removes masked tokens from the student encoder, and applies geometric corruptions only to local views.

2 Related Work

Self-Distillation in 2D. BYOL (Grill et al., 2020) and DINO (Caron et al., 2021) establish self-distillation as a powerful paradigm for image representation learning, in which an EMA teacher provides targets for a student across differently augmented views of the same image. DINO adopts a multi-crop strategy (Caron et al., 2020), where the teacher processes global views while the student additionally processes local views and matches the teacher's distributions over learned prototypes across views. iBOT (Zhou et al., 2022) combines this objective with masked modeling: the student processes masked global views and predicts the teacher's prototype assignments at masked locations from the remaining visible context. DINOv2 (Oquab et al., 2024) builds on these objectives while substantially scaling the training data and model capacity to learn general-purpose representations.

SSL on 3D Scenes. Early 3D SSL approaches rely on contrastive objectives, enforcing consistency across corresponding points, regions, or augmented scene views (Xie et al., 2020; Hou et al., 2021; Zhang et al., 2021; Chen et al., 2022a; Wang et al., 2024). Subsequent methods complement contrastive learning with reconstruction objectives, such as recovering masked geometry, color, or surface normals (Wu et al., 2023; Xu et al., 2023). More recent approaches adopt self-distillation frameworks inspired by advances in 2D vision (Caron et al., 2021; Zhou et al., 2022; Oquab et al., 2024). Sonata (Wu et al., 2025) distills global teacher representations into local and masked student views, while MSM (Hermosilla et al., 2025) reconstructs teacher features at masked locations through a hierarchical decoder. Subsequent works extend this framework in complementary directions: DOS (Abdelsamad et al., 2026) restricts supervision to observable regions and introduces long-tail-aware prototype assignments, PointINS (Yang et al., 2026) combines self-distillation with instance-aware geometric learning, and LAM3C (Yamada et al., 2026) scales pretraining using video-generated point clouds. In this work, we identify limitations of these current approaches and revisit their core design choices, resulting in substantially stronger representations.

2D–3D Distillation. A parallel line of work transfers knowledge from pretrained VFMs into 3D encoders for representation learning. DITR (Knaebel et al., 2026) distills dense 2D foundation-model features into a 3D backbone, while Concerto (Zhang et al., 2025) combines 3D self-distillation with cross-modal 2D–3D alignment. Utonia (Zhang et al., 2026) further extends this paradigm toward joint pretraining across heterogeneous point-cloud domains. These approaches demonstrate that pretrained image representations provide powerful semantic supervision for 3D representation learning, but overlook the view-dependent information encoded in VFM features. Instead, we construct a view-consistent target space before transferring this supervision to 3D.

3 Method

We revisit the key design choices of self-distillation for sparse 3D scenes: the encoder, local prediction, masked prediction, and 2D–3D distillation. We first establish a stronger baseline by adapting Sonata (Wu et al., 2025) from Point Transformer V3 (Wu et al., 2024) to the Volume Transformer (Volt) (Yilmaz et al., 2026). We then introduce asymmetric local prediction with clean teacher and strongly augmented student views, while decoupling the teacher's target head from the student's prediction head. For masked prediction, we remove masked tokens from the encoder and reconstruct them solely from visible context using a cross-attentive predictor. Fig. 3 provides an overview. Finally, we perform a second stage of 2D–3D distillation using view-consistent targets from a frozen vision foundation model rather than directly regressing its view-dependent features.

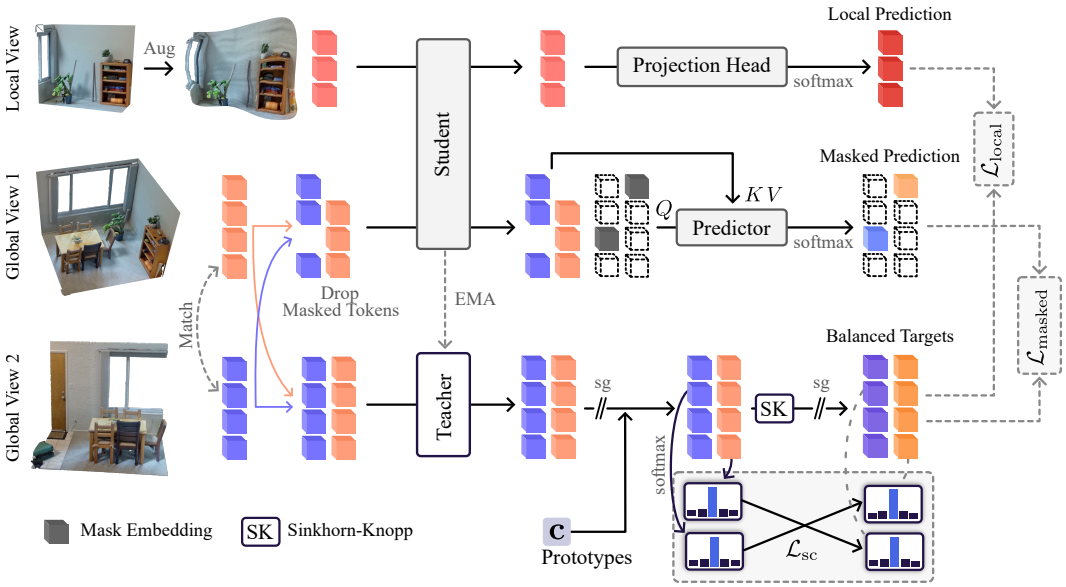

Figure 3: **VINO Overview.** Global and local crops are sampled from a scene. The **Student** encodes masked global and corrupted local views. The **Teacher** is an EMA of the student, encoding unmasked global views. The **Balanced Targets** are balanced assignments of teacher's tokens to K learned prototypes. The **Predictor** predicts the targets at masked positions from visible student tokens. The **Projection Head** predicts the targets of corresponding global-view tokens from local-view tokens.

3.1 Baseline

Sonata (Wu et al., 2025) is an established framework for 3D self-distillation that uses PTv3 (Wu et al., 2024) as the encoder. It trains a student on local and masked views to match targets produced by a teacher from global views. Unlike 2D self-distillation approaches, it performs both local and masked predictions densely. To obtain dense representations from PTv3's multi-resolution encoder, Sonata upsamples features from multiple encoder stages and concatenates them before applying the distillation objective. For masked modeling, it follows the 2D paradigm closely by replacing masked voxel features with a learned mask embedding and predicting the corresponding targets. As in 2D, these objectives train the student to match the teacher's soft assignments over learned prototypes.

We start by adapting Sonata to the current best-performing 3D backbone, Volt (Yilmaz et al., 2026). Unlike PTv3, Volt operates on a single-resolution sequence of volumetric tokens. We can therefore apply dense supervision directly to the encoder's output representations. Replacing PTv3 with Volt substantially improves training stability and enables a simpler optimization recipe. Specifically, following DINOv3 (Siméoni et al., 2025), after a warmup we use a constant learning rate, keep weight decay fixed, and remove all other hyperparameter schedules, allowing training to be extended as long as performance improves (Tab. 1).

	ScanNet200
Sonata† (PTv3)	28.3
Sonata (Volt)	**29.6**
Simpler opt. recipe	29.5

Table 1: **Baseline.** Semantic segmentation linear-probing performance (mIoU). The simpler optimization recipe results in a similar performance. † is reproduced.

3.2 Local Prediction

To isolate the effect of local prediction, we first remove the masked branch from our baseline and find that it has a minor impact on performance (see Tab. 2). We then revisit how the views for local prediction are constructed. In Sonata, both the local student views and global teacher views use the same set of augmentations (see Fig. 2). This exposes the teacher to the same strong geometric perturbations as the student, potentially degrading its targets. Instead, we use an asymmetric augmentation pipeline in

	ScanNet200
No masked loss	28.9
Asymmetric augm.	31.3
Decoupled head	**32.8**

Table 2: **Local Prediction.** The masked branch is removed. Same evaluation setup as in Tab. 1.

which the teacher receives clean global views, while the student processes strongly augmented local views. The student must therefore learn to predict teacher assignments from substantially corrupted inputs. This asymmetry also changes how the teacher should construct its targets. In Sonata, the teacher's prediction head is an EMA copy of the student's head. Under our formulation, however, the student head operates on representations of corrupted views while the teacher head operates on representations of clean views, so coupling them through EMA would force the teacher head to track a student head operating on a different representation space, preventing the teacher and student from specializing properly for their individual objectives. We therefore use an independent teacher prediction head, trained separately to produce targets rather than copied from the student via EMA.

Teacher Target Construction. In modern self-distillation methods, the teacher head maps encoder outputs to soft assignments to K cluster centroids or prototypes that serve as targets for the student. Inspired by CAPI (Darcet et al., 2025), we only equip the teacher with an independently initialized linear layer that learns the prototypes $\{c_k\}_{k=1}^{K}$, with $c_k \in \mathbb{R}^D$. For each teacher encoder output $x_i \in \mathbb{R}^D$, we compute the cosine similarity $\ell_{i,k}$ with each prototype c_k. We then sharpen these similarities with teacher temperature τ_t and apply the Sinkhorn–Knopp (Sinkhorn & Knopp, 1967) (SK) algorithm to obtain balanced soft assignments $P_t \in \mathbb{R}^{N\times K}$:

$$\ell_{i,k} = \frac{c_k^\top x_i}{\|c_k\|_2 \|x_i\|_2}, \qquad P_t = \mathrm{SK}\left(\frac{\ell}{\tau_t}\right). \tag{1}$$

The student predicts distributions over the same K prototypes and is trained with cross-entropy loss against the teacher's assignments. By decoupling target construction from the student's prediction, the teacher can define stable targets from clean encoder representations, while the student's prediction head is free to adapt to the corrupted representations from which the targets must be recovered.

Learning the Prototypes. The self-distillation objective trains the student, but cannot update the teacher prototypes as that would couple the target space to its predictions and lead to collapse. We therefore learn the prototypes independently using correspondences between two global teacher views. Let $\mathcal{M}$ denote the set of teacher embeddings of corresponding patches. For each pair $(i, j) \in \mathcal{M}$, the balanced assignment of one embedding, which is used to supervise the student, also serves as a fixed target for the other's distribution over prototypes, where τ_C is the temperature:

$$\mathcal{L}_{\mathrm{sc}} = \frac{1}{|\mathcal{M}|} \sum_{(i,j)\in\mathcal{M}} \left[\mathrm{CE}(P_{t,i}, q_j) + \mathrm{CE}(P_{t,j}, q_i)\right], \qquad q_i = \mathrm{softmax}\left(\frac{\ell_i}{\tau_C}\right). \tag{2}$$

Gradients are stopped through both the teacher encoder and the Sinkhorn-Knopp assignments (Asano et al., 2020), such that the swapped clustering loss $\mathcal{L}_{\mathrm{sc}}$ updates only the prototypes. This produces a target space whose learning signal comes entirely from teacher representations and is independent of the student prediction head. See experimental results in Tab. 2.

3.3 Masked Prediction

Masked modeling trains the student to predict targets constructed by the teacher for hidden regions using the surrounding visible context. In 2D vision, masking is commonly done by replacing selected tokens with a learned mask embedding while retaining them in the sequence. Existing 3D approaches adopt a similar strategy for point clouds (Wu et al., 2023; 2025; Zhang et al., 2025; 2026). In sparse 3D data, however, token occupancy itself carries geometric information. Consequently, retaining mask tokens at their original spatial locations can provide a *geometric shortcut* (Wu et al., 2025; Pang et al., 2022; Knaebel et al., 2023), allowing the student to exploit the geometry of the masked region when predicting its targets rather than relying on the surrounding visible context.

Cross-Attentive Predictor. To avoid this shortcut, we remove masked tokens from the student encoder and process only the remaining tokens. To predict features for the masked tokens as required for masked modeling, we introduce a lightweight, cross-attention-only Transformer decoder, following Fu et al. (2025); Leijenaar & Kasaei (2025). This decoder uses queries that are initialized with a shared learned mask embedding and assigned the 3D position of their corresponding masked tokens. Across decoder layers, queries cross-attend to the encoded visible tokens using the same 3D positional encoding as

	ScanNet200
Local pred.	32.8
+ Masked pred.	35.6
+ Fix artifacts	**35.9**

Table 3: **Masked Prediction.** Same evaluation setup as in Tab. 1.

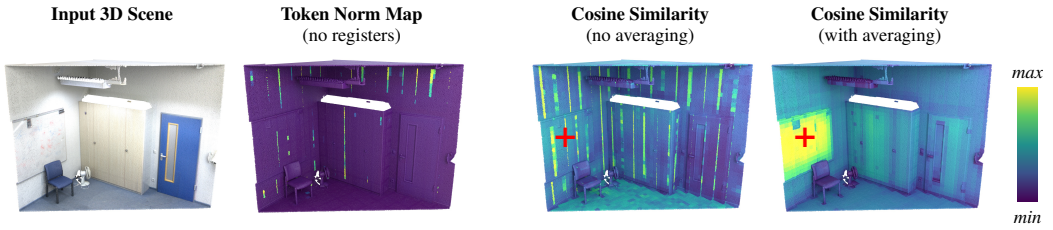

Figure 4: **Representation Artifacts.** We observe high-norm token artifacts in the representations (left). Register tokens largely eliminate these artifacts, but early in training, the learned representations still strongly reflect local voxel occupancy. Averaging voxel features within each patch removes this residual artifact (right).

the encoder. We omit self-attention among queries to prevent them from using the positions of other masked queries for prediction, and rely solely on the encoded visible context. The final decoder features for each masked token are supervised by the corresponding targets produced by the teacher.

Shared Target Space. The separation between the teacher's target head and the student prediction head is also effective for masked prediction, where the student operates on decoder features while the teacher operates on encoder features. We therefore reuse the same targets as for local prediction. For each queried masked location, the complete teacher view provides a target assignment that supervises the decoder prediction.

Preventing Representation Artifacts. During self-supervised pretraining, we observe high-norm token artifacts similar to those previously reported for ViTs (Darcet et al., 2024). Following prior work, we introduce learned register tokens (Darcet et al., 2024), which remove many of these outliers, but not all of them, particularly early in training. We find that the remaining artifacts are associated with the tokenizer. Volt aggregates the voxel features within each patch by a weighted sum. Because patches might contain different numbers of occupied voxels, the magnitude of the resulting token depends on local occupancy. Early in training, before semantic representations emerge, the model can rely on occupancy-dependent token magnitudes as a shortcut for matching the teacher. We remove this shortcut by averaging the patch feature over its occupied voxels rather than summing them. This decouples token magnitude from voxel occupancy and substantially reduces the remaining high-norm artifacts; see Fig. 4. For experimental results, see Tab. 3.

3.4 2D–3D Distillation

Recent 3D representation learning methods supervise 3D encoders with VFM features extracted from aligned posed images (Knaebel et al., 2026; Zhang et al., 2025; 2026).

Two-Stage Training. We first examine when VFM supervision should be introduced and find that applying 2D–3D distillation from the start of training underperforms first learning the 3D representation fully self-supervised and aligning it with the VFM afterwards (Tab. 4). We therefore divide pretraining into two stages. The first stage learns 3D representations fully self-supervised from high-quality point clouds using the objectives introduced above. In the second stage, we additionally distill knowledge from a pretrained VFM (Oquab et al., 2024).

	ScanNet200
Align from start	37.2
Raw embeddings	38.6
Prototypes	**39.8**

Table 4: **2D–3D Distillation.** Linear probing (mIoU) for different 2D–3D distillation objectives.

Clustering VFM Embedding Space. We next examine how VFM features should supervise the 3D encoder. Previous approaches typically linearly project 3D features into the VFM embedding space and align them with corresponding 2D features using a cosine-similarity objective. However, VFM representations encode not only semantic content but also view-dependent information, such as image-grid position (Cuttano et al., 2026). Consequently, directly regressing these embeddings can assign different targets to the same 3D region when observed from different viewpoints, yielding inconsistent supervision for a view-independent 3D model. To reduce this view dependence, we construct the distillation targets in a separate prototype space learned in the embedding space of the frozen VFM. Using camera poses and depth, we establish correspondences between 2D tokens in overlapping views and optimize the prototypes with the same swapped clustering loss $\mathcal{L}_{\text{sc}}$ used for the 3D teacher prototypes. Because corresponding tokens across views are optimized to share consistent assignments, the resulting prototype space captures components of the

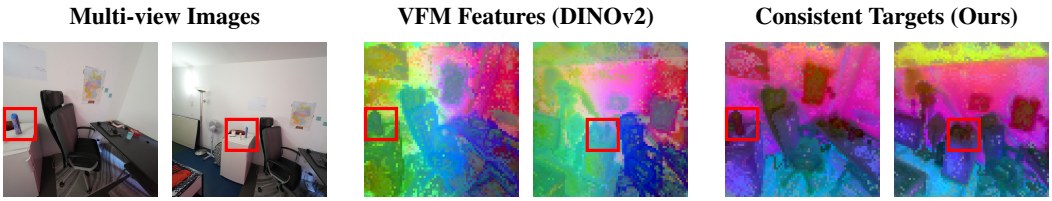

Figure 5: **Consistent 2D Targets.** VFM (Oquab et al., 2024) features of the same 3D region vary across views. We learn prototypes in the frozen VFM feature space, optimized such that corresponding tokens across views produce matching soft assignments. These serve as our targets for 2D–3D distillation.

VFM representation that are consistent across viewpoints while reducing sensitivity to view-specific variation. We then use the resulting soft prototype assignments as distillation targets, training each 3D token to predict the assignment distribution of its corresponding 2D token.

Robustness. To improve robustness across diverse point clouds, where color may be unavailable and scenes may not be gravity-aligned, we extend the local-view augmentations during the second training stage. Specifically, we apply random color dropout and unrestricted $\mathrm{SO}(3)$ rotations to local views. We introduce these transformations only in the second stage, as they enforce invariances that are desirable for downstream tasks but are not required for the 3D-only stage.

4 Experiments

Pretraining Data. For indoor scenes, the first, 3D-only stage uses the real-scene subset of Sonata's pretraining data (Wu et al., 2025), comprising ScanNet (Dai et al., 2017), ScanNet++ (Yeshwanth et al., 2023), S3DIS (Armeni et al., 2016), ARKitScenes (Baruch et al., 2021), and HM3D (Ramakrishnan et al., 2021), totaling approximately 18,000 scenes. In the second stage, we additionally include RealEstate10K (Zhou et al., 2018), increasing the indoor pretraining set to approximately 90,000 scenes. For outdoor experiments, both stages use the same point clouds from nuScenes (Caesar et al., 2020), SemanticKITTI (Behley et al., 2019), and Waymo (Sun et al., 2020).

Implementation Details. We train separate models for indoor and outdoor scenes. Following previous work (Wu et al., 2025), we sample two global crops with ratios 0.4–1.0 and four local crops with ratios 0.1–0.4. Student global views are block-masked with a 55% masking ratio and a block size of 8 tokens. During the first stage, we train the model with AdamW for 200k steps using a batch size of 32. For the second stage, we initialize from the corresponding SSL checkpoint and continue to train for 30k steps, using a frozen DINOv2-g/14 (Oquab et al., 2024) image encoder following prior work (Zhang et al., 2025; 2026). Additional details are in the appendix.

4.1 Semantic Segmentation

Linear Probing. We first evaluate the quality of the representations by freezing the pretrained 3D encoder and training only a linear head for semantic segmentation. Tab. 5 (*top*) compares methods using only 3D pretraining. Here, VINO achieves the best linear-probing performance on five of the six benchmarks. The gains are particularly pronounced on indoor scenes, improving over the strongest prior result by $+5.2$ mIoU on ScanNet200, $+2.2$ on ScanNet, and $+3.5$ on ScanNet++. On outdoor scenes, VINO improves nuScenes to 76.2 mIoU and SemanticKITTI to 68.5 mIoU, while remaining 1.7 mIoU below DOS on Waymo. Notably, VINO uses an 85.7M-parameter Volt-B encoder, compared with the larger 108.4M-parameter PTv3-B backbone used by most prior methods.

Adding 2D–3D distillation further improves the learned representation (Tab. 5, *bottom*). Compared with our 3D-only model, it improves performance by $+3.9$ mIoU on ScanNet200, $+6.2$ on ScanNet++, $+3.0$ on nuScenes, and $+3.2$ on Waymo. As a result, VINO achieves the best performance across all six benchmarks, outperforming Concerto by $+2.4$ mIoU on ScanNet200 and $+3.0$ on ScanNet++, and Utonia by $+3.7$ on nuScenes and $+4.7$ on Waymo. Together, these results demonstrate the effectiveness of the design choices revisited in VINO.

Fine-tuning. We next evaluate the representations' adaptability to downstream tasks by jointly fine-tuning the pretrained encoder and a lightweight task-specific decoder on each benchmark. As shown in Tab. 6, VINO achieves the best performance on five of the six benchmarks across indoor

	Method	Backbone	Indoor Datasets			Outdoor Datasets		
			ScanNet200	ScanNet	ScanNet++	nuScenes	Sem.KITTI	Waymo
	Supervised†	Volt-B (87.7 M)	36.2	77.7	51.7	81.3	70.6	71.5
3D only	LAM3C (Yamada et al., 2026)	PTv3-B (108.4 M)	25.3	66.0	34.2	–	–	–
3D only	MSM (Hermosilla et al., 2025)	HUNet (67.6 M)	26.8	68.7	–	–	–	–
3D only	Sonata (Wu et al., 2025)	PTv3-B (108.4 M)	29.3	72.5	38.9	66.1	62.0	60.5
3D only	PointINS (Yang et al., 2026)	PTv3-B (108.4 M)	29.6	72.4	–	74.6	68.0	–
3D only	DOS (Abdelsamad et al., 2026)	PTv3-B (108.4 M)	30.7	73.9	–	74.8	68.3	**67.0**
3D only	**VINO** (Ours)	Volt-B (85.7 M)	**35.9**	**76.1**	**42.4**	**76.2**	**68.5**	65.3
with 2D	LargeAD (Kong et al., 2025)	MinkUNet (37.9 M)	–	–	–	47.8	35.8	36.1
with 2D	ScaLR (Puy et al., 2024)	WaffleIron (59.4 M)	–	–	–	67.8	55.8	–
with 2D	Concerto (Zhang et al., 2025)	PTv3-B (108.4 M)	37.4	77.3	45.6	74.2	66.6	62.2
with 2D	Utonia (Zhang et al., 2026)	PTv3-B (137.3 M)	36.4	77.7	44.7	75.5	67.7	63.8
with 2D	**VINO** (Ours)	Volt-B (85.7 M)	**39.8**	**77.9**	**48.6**	**79.2**	**69.8**	**68.5**

Table 5: **Linear Probing 3D Semantic Segmentation (mIoU).** For each dataset, a linear classifier is trained on the training split with the 3D encoder frozen and evaluated on the validation split. Methods are grouped by 3D-only pretraining *(top)* and pretraining with additional 2D supervision *(bottom)*. DOS and PointINS outdoor results use PTv3-S. † is reproduced.

Method	Backbone	Indoor Datasets			Outdoor Datasets		
		ScanNet200	ScanNet	ScanNet++	nuScenes	Sem.KITTI	Waymo
D-DITR (Knaebel et al., 2026)	PTv3-S (46.2 M)	37.7	79.2	–	80.7	69.8	72.1
Sonata (Wu et al., 2025)	PTv3-B (124.8 M)	36.8	79.4	49.3	81.7	72.6	72.9
DOS (Abdelsamad et al., 2026)	PTv3-B (124.8 M)	37.1	79.7	–	81.8	73.5	**73.9**
PointINS (Yang et al., 2026)	PTv3-B (124.8 M)	36.6	79.0	–	81.1	72.7	–
Concerto (Zhang et al., 2025)	PTv3-B (124.8 M)	39.2	80.7	50.7	82.0	71.2	69.2
Utonia (Zhang et al., 2026)	PTv3-B (157.7 M)	39.6	81.1	49.0	82.2	72.0	71.4
VINO (Ours)	Volt-B (94.1 M)	**42.3**	**81.5**	**52.2**	**82.5**	**74.1**	72.4

Table 6: **Fine-tuning 3D Semantic Segmentation (mIoU).** Pretrained encoders and task-specific decoders are jointly fine-tuned on the training split and evaluated on the validation split for each dataset.

and outdoor datasets. The largest gains are observed on indoor scenes, improving the previous best results by +2.7 mIoU on ScanNet200 and +1.5 on ScanNet++. Notably, these results are obtained with a 94.1M-parameter Volt-B backbone, smaller than the PTv3-B backbones used by most competing methods. This shows that the representations learned by VINO are not only readily usable as frozen features but also provide strong initialization for full downstream fine-tuning.

4.2 Out-of-Domain Transfer

We next evaluate whether the learned representations transfer beyond the datasets seen during pretraining in two complementary settings: (i) point-cloud registration, probing geometric correspondence, and (ii) single-view RGB-D semantic segmentation, probing semantic transfer.

Point-Cloud Registration. We form candidate correspondences by nearest-neighbor matching of frozen features between the two point clouds and pass them to a registration backend, SC^2-PCR (Chen et al., 2022b). We evaluate on 3DMatch (Zeng et al., 2017) and its low-overlap counterpart, 3DLoMatch (Huang et al., 2021), and report registration success rate (%) following Pan et al. (2026). As shown in Tab. 7, VINO achieves 89.7% on 3DMatch and 61.1% on 3DLoMatch. This exceeds Utonia by +13.2 and +24.8 points and Concerto by +24.7 and +38.4 points, respectively. VINO also outperforms the hand-crafted FPFH descriptor (Rusu et al., 2009) by +12.2 points on 3DMatch and +24.7 points on 3DLoMatch. Despite using frozen features and without pretraining on 3DMatch, VINO comes within 6.2 points of the state-of-the-art RAP (Pan et al., 2026) model on 3DMatch, which is trained specifically for registration. These results show that the learned representations transfer effectively to geometric correspondence without registration-specific training.

Cross-Modal Semantic Probing. We further compare 2D and 3D representations by evaluating them in two complementary settings outside their native domains (see Tab. 8). On NYUv2 (Sil-

Method	3DMatch	3DLoMatch
FPFH (Rusu et al., 2009)	77.5	36.4
Concerto	65.0	22.7
Utonia	76.5	36.3
VINO (Ours)	89.7	61.1
RAP (Pan et al., 2026)	95.9	78.8

Table 7: **Point-Cloud Registration.** Registration success rate (%) using frozen descriptors. FPFH is a handcrafted descriptor. RAP is a state-of-the-art registration model shown for reference.

Method	Arch.	NYUv2	ScanNet++
DINOv2	ViT-B/14	56.0	34.5
DINOv2	ViT-g/14	59.0	39.3
DINOv3	ViT-B/16	62.0	34.2
Concerto	PTv3-B	54.4	45.6
Utonia	PTv3-B	55.1	44.7
VINO (Ours)	Volt-B	59.7	48.6

Table 8: **Cross-Modal Semantic Probing (mIoU).** For 3D methods, NYUv2 images are lifted to 3D. For 2D methods, their features are aggregated in 3D.

berman et al., 2012), we bring the 3D models to single-view RGB-D by lifting each image into 3D using a monocular geometry estimator (Kong et al., 2026), and we linearly probe the frozen features. Despite being pretrained on full-scene 3D reconstructions, VINO reaches 59.7 mIoU, outperforming both prior 3D representations and DINOv2-g, which also serves as our 2D teacher during pretraining, while approaching DINOv3-B. Conversely, on ScanNet++ (Yeshwanth et al., 2023), we bring the 2D VFMs to 3D by lifting and aggregating their features across posed RGB views before linear probing. Here, VINO reaches 48.6 mIoU, compared with 39.3 for DINOv2-g and 34.2 for DINOv3-B, again outperforming the competing 3D representations. This complementary comparison shows that our representation remains highly competitive in the challenging single-view setting, while its advantage becomes substantially larger when moving to multi-view, full-scene 3D understanding.

4.3 Analyses

Predictor Analysis. We analyze the masked predictor design in Tab. 9. Our cross-attention-only decoder performs best at 35.9 mIoU. Adding query-to-query attention reduces performance to 34.2, while a full self-attention decoder causes a further drop to 33.0. Since each query carries the position of its masked region, any interaction among queries exposes the geometry of the masked region and can reintroduce the geometric shortcut we try to avoid. Finally, we find that allowing decoder queries to additionally attend to register tokens, rather than only the 3D patch tokens, reduces performance to 35.3 mIoU.

Predictor Design	ScanNet200
Cross-attention only decoder	**35.9**
Query-to-query attention	34.2
Full self-attention decoder	33.0
Registers as context	35.3

Table 9: **Masked Predictor.** Semantic segmentation linear-probing performance (mIoU) for different predictor designs.

Masking Analysis. We analyze the masking configuration in Tab. 10. A block size of 80 cm performs best at 35.9 mIoU, while 55% masking outperforms both lower and higher ratios. This suggests that moderate spatial masking provides a good balance between prediction difficulty and visible context. We therefore use 80 cm blocks with a 55% masking ratio throughout.

Block Size	ScanNet200	Mask Ratio	ScanNet200
60 cm	35.7	45%	35.2
80 cm	**35.9**	55%	**35.9**
100 cm	34.5	65%	34.8

Table 10: **Masking.** Linear probing performance across block sizes and mask ratios.

5 Conclusion

We introduce VINO, a 3D representation learning framework that revisits the key design choices of current approaches. Building on the Volume Transformer, VINO constructs reliable targets using clean global views and a target head decoupled from the student. For masked modeling, it removes masked tokens from the encoder, preventing the student from exploiting their locations and requiring it to infer their representations from the remaining scene context through a separate predictor. Finally, it distills knowledge from a 2D VFM by learning view-consistent prototype assignments and using them to supervise the corresponding 3D representations. Together, these changes substantially improve frozen representations, provide a strong initialization for fine-tuning, and enable transfer to unseen datasets. We hope this work helps bring 3D representation learning closer to the powerful general-purpose representations that scalable self-supervised pretraining has unlocked in 2D.

AI USE STATEMENT

In this work, we used generative AI tools to implement methods, and clean and reformat datasets. We have not used generative AI tools to help develop theoretical models or conceptual frameworks, propose or refine hypotheses, design or provide feedback on research methodology or experiments, assist with translation, support qualitative and thematic data analysis, or interpret results. Generating synthetic data sets, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs are not applicable to this work. Additionally, we used generative AI tools to create or modify scientific figures or images, create or edit software code, summarize or analyse existing literature, source/search for information, edit a research paper to improve readability, identify relevant literature, and propose a title or keywords for a research paper. We have reviewed all AI-assisted work. Generative AI was used primarily as an assistive tool for programming. All AI-assisted code was manually reviewed and tested by the authors. Suggestions related to figures, literature, and information retrieval were manually verified. References were checked against the original sources. Generative AI outputs were treated as suggestions and were only incorporated after author review and validation. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility and future research, we will publicly release our training and evaluation code as well as pretrained model checkpoints upon acceptance.

REFERENCES

Mohamed Abdelsamad, Michael Ulrich, Bin Yang, Miao Zhang, Yakov Miron, and Abhinav Valada. DOS: Distilling Observable Softmaps of Zipfian Prototypes for Self-Supervised Point Representation. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2026.

Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3D Semantic Parsing of Large-Scale Indoor Spaces. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-Labelling via Simultaneous Clustering and Representation Learning. In *International Conference on Learning Representations (ICLR)*, 2020.

Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. data2vec: A General Framework for Self-supervised Learning in Speech, Vision and Language. In *International Conference on Machine Learning (ICML)*, 2022.

Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEiT: BERT Pre-Training of Image Transformers. In *International Conference on Learning Representations (ICLR)*, 2022.

Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. ARKitScenes - a diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. In *International Conference on Neural Information Processing Systems (NEURIPS)*, 2021.

Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. In *International Conference on Computer Vision (ICCV)*, 2019.

Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A Multimodal Dataset for Autonomous Driving. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. *International Conference on Neural Information Processing Systems (NEURIPS)*, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging Properties in Self-Supervised Vision Transformers. In *International Conference on Computer Vision (ICCV)*, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. In *International Conference on Machine Learning (ICML)*, 2020.

Yujin Chen, Matthias Nießner, and Angela Dai. 4DContrast: Contrastive Learning with Dynamic Correspondences for 3D Scene Understanding. In *European Conference on Computer Vision (ECCV)*, 2022a.

Zhi Chen, Kun Sun, Fan Yang, and Wenbing Tao. SC^2-PCR: A Second Order Spatial Compatibility for Efficient and Robust Point Cloud Registration. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022b.

Claudia Cuttano, Gabriele Trivigno, Christoph Reich, Daniel Cremers, Carlo Masone, and Stefan Roth. INSID3: Training-Free In-Context Segmentation with DINOv3. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision Transformers Need Registers. In *International Conference on Learning Representations (ICLR)*, 2024.

Timothée Darcet, Federico Baldassarre, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Cluster and Predict Latent Patches for Improved Masked Image Modeling. *Transactions on Machine Learning Research (TMLR)*, 2025.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations (ICLR)*, 2021.

Letian Fu, Long Lian, Renhao Wang, Baifeng Shi, Xudong Wang, Adam Yala, Trevor Darrell, Alexei A Efros, and Ken Goldberg. Rethinking Patch Dependence for Masked Autoencoders. *Transactions on Machine Learning Research (TMLR)*, 2025.

Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap Your Own Latent - A New Approach to Self-Supervised Learning. In *International Conference on Neural Information Processing Systems (NEURIPS)*, 2020.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked Autoencoders Are Scalable Vision Learners. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Pedro Hermosilla, Christian Stippel, and Leon Sick. Masked Scene Modeling: Narrowing the Gap Between Supervised and Self-Supervised Learning in 3D Scene Understanding. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Ji Hou, Benjamin Graham, Matthias Nießner, and Saining Xie. Exploring Data-Efficient 3D Scene Understanding with Contrastive Scene Contexts. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Shengyu Huang, Zan Gojcic, Mikhail Usvyatsov, Andreas Wieser, and Konrad Schindler. PREDATOR: Registration of 3D Point Clouds with Low Overlap. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Tommie Kerssies, Niccolò Cavagnero, Alexander Hermans, Narges Norouzi, Giuseppe Averta, Bastian Leibe, Gijs Dubbelman, and Daan de Geus. Your ViT is Secretly an Image Segmentation Model. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Karim Knaebel, Jonas Schult, Alexander Hermans, and Bastian Leibe. Point2Vec for Self-Supervised Representation Learning on Point Clouds. In *German Conference on Pattern Recognition (GCPR)*, 2023.

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 *3DV*, 2026.

Lingdong Kong, Xiang Xu, Youquan Liu, Jun Cen, Runnan Chen, Wenwei Zhang, Liang Pan, Kai Chen, and Ziwei Liu. LargeAD: Large-Scale Cross-Sensor Data Pretraining for Autonomous Driving. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2025.

Lingyu Kong, Ruicheng Li, Ruicheng Wang, Sicheng Xu, Chengtang Yao, Jianfeng Xiang, and Jiaolong Yang. Fine-Detail Monocular Geometry Estimation with Self-Guided Sparse Volumetric Refinement. *arXiv preprint arXiv:2607.17967*, 2026.

Remco F. Leijenaar and Hamidreza Kasaei. Asymmetric Dual Self-Distillation for 3D Self-Supervised Representation Learning. In *International Conference on Neural Information Processing Systems (NEURIPS)*, 2025.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth Anything 3: Recovering the Visual Space from Any Views. In *International Conference on Learning Representations (ICLR)*, 2026.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, et al. DINOv2: Learning Robust Visual Features without Supervision. *Transactions on Machine Learning Research (TMLR)*, 2024.

Yue Pan, Tao Sun, Liyuan Zhu, Lucas Nunes, Iro Armeni, Jens Behley, and Cyrill Stachniss. Register Any Point: Scaling 3D Point Cloud Registration by Flow Matching. In *European Conference on Computer Vision (ECCV)*, 2026.

Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked Autoencoders for Point Cloud Self-supervised Learning. In *European Conference on Computer Vision (ECCV)*, 2022.

Gilles Puy, Spyros Gidaris, Alexandre Boulch, Oriane Siméoni, Corentin Sautier, Patrick Pérez, Andrei Bursuc, and Renaud Marlet. Three Pillars improving Vision Foundation Model Distillation for Lidar. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, Manolis Savva, Yili Zhao, and Dhruv Batra. Habitat-matterport 3d dataset (HM3d): 1000 large-scale 3d environments for embodied AI. In *International Conference on Neural Information Processing Systems (NEURIPS)*, 2021.

David Rozenberszki, Or Litany, and Angela Dai. Language-Grounded Indoor 3D Semantic Segmentation in the Wild. In *European Conference on Computer Vision (ECCV)*, 2022.

Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast Point Feature Histograms (FPFH) for 3D Registration. In *International Conference on Robotics and Automation (ICRA)*, 2009.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor Segmentation and Support Inference from RGBD Images. In *European Conference on Computer Vision (ECCV)*, 2012.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025.

Richard Sinkhorn and Paul Knopp. Concerning Nonnegative Matrices and Doubly Stochastic Matrices. *Pacific Journal of Mathematics*, 21(2):343–348, 1967.

Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke M. Strasdat, Renzo De Nardi, Michael Goesele, Steven Lovegrove, and Richard Newcombe. The Replica Dataset: A Digital Replica of Indoor Spaces. *arXiv preprint arXiv:1906.05797*, 2019.

Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

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 *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Thomas Wimmer, Prune Truong, Marie-Julie Rakotosaona, Michael Oechsle, Federico Tombari, Bernt Schiele, and Jan Eric Lenssen. AnyUp: Universal Feature Upsampling. In *International Conference on Learning Representations (ICLR)*, 2026.

Xiaoyang Wu, Xin Wen, Xihui Liu, and Hengshuang Zhao. Masked Scene Contrast: A Scalable Framework for Unsupervised 3D Representation Learning. In *International 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 *International 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 *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

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.

Mingye Xu, Mutian Xu, Tong He, Wanli Ouyang, Yali Wang, Xiaoguang Han, and Yu Qiao. MM-3DScene: 3D Scene Understanding by Customizing Masked Modeling with Informative-Preserved Reconstruction and Self-Distilled Consistency. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Ryousuke Yamada, Kohsuke Ide, Yoshihiro Fukuhara, Hirokatsu Kataoka, Gilles Puy, Andrei Bursuc, and Yuki M Asano. 3D sans 3D Scans: Scalable Pre-training from Video-Generated Point Clouds. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Bin Yang, Mohamed Abdelsamad, Miao Zhang, and Alexandru Paul Condurache. Towards Foundation Models for 3D Scene Understanding: Instance-Aware Self-Supervised Learning for Point Clouds. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. ScanNet++: A High-Fidelity Dataset of 3D Indoor Scenes. In *International Conference on Computer Vision (ICCV)*, 2023.

Kadir Yilmaz, Adrian Kruse, Tristan Höfer, Daan de Geus, and Bastian Leibe. Volume Transformer: Revisiting Vanilla Transformers for 3D Scene Understanding. *European Conference on Computer Vision (ECCV)*, 2026.

Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-BERT: Pre-Training 3D Point Cloud Transformers With Masked Point Modeling. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

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. *International Conference on Neural Information Processing Systems (NEURIPS)*, 2025.

Yujia Zhang, Xiaoyang Wu, Yunhan Yang, Xianzhe Fan, Han Li, Yuechen Zhang, Zehao Huang, Naiyan Wang, and Hengshuang Zhao. Utonia: Toward One Encoder for All Point Clouds. *arXiv preprint arXiv:2603.03283*, 2026.

Zaiwei Zhang, Rohit Girdhar, Armand Joulin, and Ishan Misra. Self-Supervised Pretraining of 3D Features on any Point-Cloud. In *International Conference on Computer Vision (ICCV)*, 2021.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. iBot: Image BERT Pre-training with Online Tokenizer. In *International Conference on Learning Representations (ICLR)*, 2022.

Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo Magnification: Learning View Synthesis using Multiplane Images. In *ACM SIGGRAPH Conference*, 2018.

A SUPPLEMENTARY

A.1 EXTENDED RELATED WORK

2D Self-supervised Learning. Self-supervised representation learning in 2D vision has evolved from contrastive objectives toward teacher–student self-distillation and masked prediction. Early contrastive methods such as MoCo (He et al., 2020) and SimCLR (Chen et al., 2020) learn invariances by bringing representations of different augmentations of the same image closer while separating those of different images. SwAV (Caron et al., 2020) instead introduces online clustering and swapped prediction, matching cluster assignments across augmented views without explicit pairwise comparisons. BYOL (Grill et al., 2020) further shows that strong representations can be learned without negative samples by training an online network to predict representations produced by a slowly updated target network. Building on these ideas, DINO (Caron et al., 2021) applies EMA-based teacher–student self-distillation to Vision Transformers (Dosovitskiy et al., 2021), matching teacher and student predictions across multiple image crops and revealing strong semantic structure in the learned features.

In parallel, masked image modeling emerged as an effective pretraining paradigm. BEiT (Bao et al., 2022) masks image patches and trains the model to predict discrete visual tokens produced by a separate tokenizer. MAE (He et al., 2022) instead reconstructs masked patches directly in pixel space, using a lightweight decoder to recover low-level image content from visible context. iBOT (Zhou et al., 2022) brings masked prediction into the self-distillation framework by predicting teacher assignments at masked patch locations, extending DINO-style supervision from image-level representations to dense latent tokens. DINOv2 (Oquab et al., 2024) further scales this family of objectives and demonstrates that self-supervised ViTs can serve as strong general-purpose visual representations across a broad range of downstream tasks. More recently, CAPI (Darcet et al., 2025) revisits the construction of masked prediction targets by clustering latent patch representations and decoupling target construction from prediction, while DINOv3 (Siméoni et al., 2025) further advances large-scale self-supervised ViT pretraining. Our work builds on this line of research, but revisits how its central ingredients (teacher–student distillation, prototype-based targets, and masked prediction from visible context) should be formulated for sparse 3D scenes.

3D Object-level Self-supervised Learning. For object-level point clouds, much of the notable work adapts masked image modeling to 3D. Point-BERT (Yu et al., 2022) and Point-MAE (Pang et al., 2022) both use masked reconstruction. Point-BERT predicts discrete tokens of masked point patches from a pretrained tokenizer. Point-MAE reconstructs the masked patches directly and leaves the mask tokens out of the encoder to avoid leaking positional information. Point2vec (Knaebel et al., 2023) applies data2vec (Baevski et al., 2022) to object-level point clouds and adds a separate predictor network. AsymDSD (Leijenaar & Kasaei, 2025) disables query-to-query self-attention in its decoder and adapts SwAV's (Caron et al., 2020) multi-crop mechanism to point clouds.

A.2 IMPLEMENTATION DETAILS

Input and View Construction. We use the same training datasets as Sonata (Wu et al., 2025), excluding the synthetic datasets. Indoor point clouds are voxelized at 2 cm and represented by RGB and surface normals. Outdoor point clouds use 5 cm voxels, LiDAR intensity, and normals which are estimated from local neighborhoods. Coordinates are used for spatial processing but are not used as input features in either setting. We sample two global spatial crops with ratios of 0.4–1.0 and four local crops with ratios of 0.1–0.4. We cap indoor crops at $65{,}536$ voxelized points and outdoor crops at $6{,}000$ tokens. All views independently receive standard geometric augmentations including scaling, rotations, axis flips, and coordinate jitter. For local views, we randomly select three of four indoor corruptions (elastic distortion, point dropout, color jitter, and chromatic translation), or two of four outdoor corruptions (normal dropout, point dropout, elastic distortion, and 1 m block dropout). We mask 55% of occupied blocks in student global views, using blocks with a side length of eight tokens. We establish cross-view token correspondences by nearest-neighbour matching of token centers in the original coordinate system, rejecting matches beyond a distance threshold. Self-supervised pretraining for 200k steps takes roughly 24 hours on 8 H100s indoors and 16 H100s outdoors.

3D Self-Supervised Pretraining. We use AdamW with batch size 32, learning rate 8×10^{-4}, 5% warmup, weight decay 0.05, and gradient clipping 3.0. The teacher's trainable clustering head uses half the learning rate. Teacher EMA momentum is 0.997. We randomly sample up to 512 masked queries per global view. We use 4096 prototypes, a teacher temperature τ_t of 0.05, a clustering prediction temperature τ_C of 0.1, and three Sinkhorn–Knopp iterations.

The self-supervised objective is made up of: local prediction $\mathcal{L}_{\text{local}}$, masked prediction $\mathcal{L}_{\text{masked}}$, and cross-view clustering $\mathcal{L}_{\text{sc}}$. They each use cross-entropy against the corresponding detached targets. The masked term is the sum of within-view and cross-view masked prediction losses:

$$\mathcal{L}_{\text{masked}} = \mathcal{L}_{\text{masked-within}} + \mathcal{L}_{\text{masked-cross}}.$$

The overall objective is

$$\mathcal{L}_{\text{SSL}} = 0.25\,\mathcal{L}_{\text{local}} + 0.25\,\mathcal{L}_{\text{masked}} + 0.25\,\mathcal{L}_{\text{sc}}.$$

$\mathcal{L}_{\text{sc}}$ optimizes only the clustering head on detached teacher features and does not backpropagate through either encoder.

The Volt-B encoder uses 16 register tokens, LayerScale is initialized to 10^{-5}, and a stochastic-depth rate of 0.1 is used. The predictor consists of four layers. We sample the queries randomly from the tokens dropped by masking.

The student's local projection head is a $768 \to 2048 \to 128$ MLP with GELU activation, followed by ℓ_2 normalization and a linear layer with 4096 outputs. The final linear layer has no bias and is initialized with unit-standard-deviation Gaussian weights. We do not apply additional temperature scaling to the student distillation logits. Analogously for masked prediction, an ℓ_2-normalized predictor representation is mapped to the 4096-dimensional target space using a linear layer.

2D–3D Distillation. We initialize the 3D encoder from the fully self-supervised checkpoint and use a frozen DINOv2-g/14 (Oquab et al., 2024) image encoder. Separately, we train the image prototype head on the same training data using the swapped clustering loss between image patches that observe the same 3D points. This training uses a queue of up to 16,384 image features per GPU and a teacher temperature of 0.04. We then freeze the image prototype head and continue 3D pretraining for 30k steps with added 2D–3D supervision. All 3D distillation objectives remain active. The learning rate decays by a cosine schedule from 8×10^{-4} to 8×10^{-7}. This stage takes approximately 9 hours on eight H100 GPUs indoors and 8 hours on sixteen H100 GPUs outdoors.

Following Concerto (Zhang et al., 2025), each indoor sample pairs a point cloud with up to four posed RGB images, processed at 518×518. ScanNet, ScanNet++, ARKitScenes, and RealEstate10K are sampled approximately uniformly. Outdoors we use all six nuScenes cameras, all five Waymo cameras, and the left SemanticKITTI camera, resizing images to width 672 while preserving aspect ratio and padding to a multiple of 14. We use a budget of 8192 tokens and one local crop with ratio 0.3–0.7. Indoors we additionally apply color dropout with probability 0.3 and rotations around the x- and y-axes with probability 0.3. We calculate 2D–3D correspondences by projecting 3D points into the posed images. For indoor correspondences we require depth agreement with the image's depth map within 2 cm. For both indoor and outdoor data, we discard correspondences more than 5 cm behind the nearest projected point within an image patch. Two image patches correspond for the swapped clustering loss when they share a valid 3D point.

For the application of the 2D–3D loss, we follow Concerto's (Zhang et al., 2025) patch-level alignment approach. Encoded 3D features are averaged per corresponding image patch, then passed through the projection head to produce prototype logits. We supervise with the cross-entropy loss between the prototype assignments of the image patches and the pooled 3D predictions. The 3D embeddings for 2D–3D supervision come from a separate forward pass of the student encoder on the unmasked first global view. The 2D–3D loss is assigned a weight of 1.0.

Point Decoder. To obtain per-point predictions from the encoder's coarse patch embeddings, we use a lightweight cross-attention decoder. The design is inspired by AnyUp (Wimmer et al., 2026). A $7 \times 7 \times 7$ submanifold sparse convolution maps per-voxel features to 256-dimensional features. These features are fed into a query and a geometry-key branch. Each branch contains two residual blocks. Each block applies RMS normalization followed by a $256 \to 1024 \to 256$ MLP with a SiLU activation function and adds the result to its input. The query branch produces one query

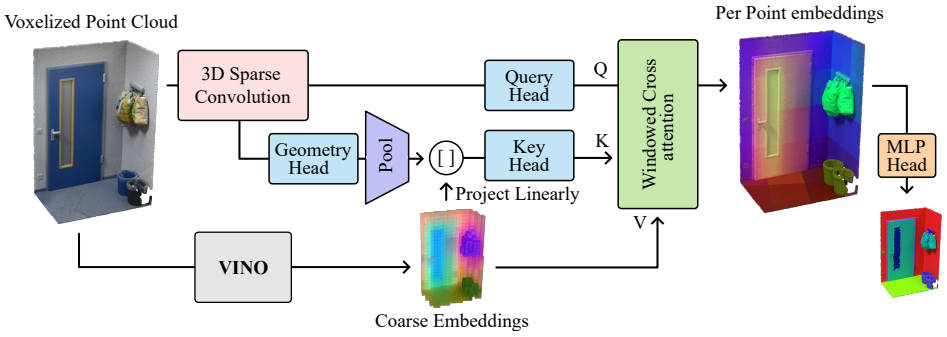

Figure 6: **Lightweight Decoder Design.** The lightweight decoder upsamples coarse VINO patch embeddings to per-point representations using windowed cross-attention. We attach this decoder to the encoder when fully fine-tuning the entire model.

per point. The geometry-key features are averaged within each patch and concatenated with a 256-dimensional linear projection of the ℓ_2-normalized encoder embedding. A $512 \to 256$ linear layer and a residual MLP block produce one key per patch from the concatenated features. The original VINO embeddings serve directly as values. We do not apply a learned value projection.

We apply RMS-norm and 3D rotary positional encoding to queries and keys, using point positions for queries and mean point positions within each patch for keys. For efficient sparse attention, we partition the coarse token grid into non-overlapping $6 \times 6 \times 6$-token query regions. Points in each region attend to an expanded $8 \times 8 \times 8$-token key/value window, adding one token of context on each side. This way we can use variable-length fused FlashAttention kernels. A pointwise $768 \to 3072 \to 768$ MLP followed by a linear classifier produces the per-point predictions.

A.3 Ablations on ScanNet++

The ablations in the main paper report results on ScanNet200 (Rozenberszki et al., 2022) only. Here, we additionally evaluate the same design choices on ScanNet++ (Yeshwanth et al., 2023) to assess whether the observed trends transfer across datasets. As shown in Tab. 11, the conclusions are largely consistent. Replacing PTv3 (Wu et al., 2024) with Volt (Yilmaz et al., 2026) improves the Sonata (Wu et al., 2025) baseline, while removing scheduling has negligible effect. For local prediction, asymmetric augmentations and the decoupled target head both improve performance. Finally, the 2D–3D distillation ablation exhibits the same ordering on both datasets: delaying alignment until the second stage is preferable to aligning from the start, and learning view-consistent prototypes with swapped prediction clearly outperforms direct feature alignment. Overall, the ScanNet++ results support the design choices identified on ScanNet200 and indicate that their benefits are not specific to a single benchmark.

	ScanNet200	ScanNet++
Baseline		
Sonata† (PTv3)	28.3	38.0
Sonata (Volt)	29.6	39.1
No scheduling	29.5	39.1
No masked loss	28.9	38.4
Local Prediction		
Baseline	28.9	38.4
Asymmetric augm.	31.3	39.3
Decoupled head	32.8	40.9
2D–3D Distillation		
No distillation	35.9	42.4
Align from start	37.2	44.8
Linear + cosine	38.6	46.1
Prototype + swapped	39.8	48.6

Table 11: **Additional Ablations.** Semantic segmentation linear-probing performance (mIoU) on ScanNet200 and ScanNet++. † is reproduced. ScanNet200 results correspond to Tabs. 1, 2, and 4.

A.4 Linear Probing With PTv3 and Volt

For encoders with a PTv3 backbone, we upsample the features from all encoder stages to the finest resolution and concatenate them for per-point features. For VINO, we concatenate the features from

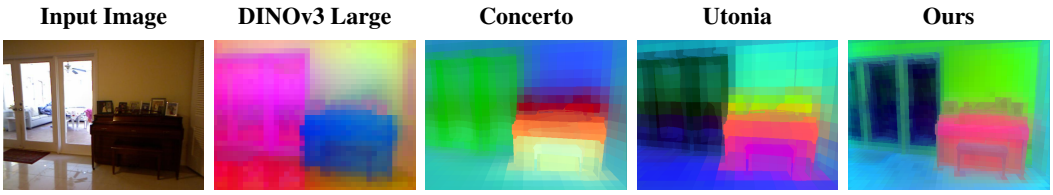

Figure 8: **Qualitative Transfer to NYUv2.** PCA of frozen features for a single NYUv2 image. For the 3D models, the image is lifted to 3D.

the last three encoder layers for each token. We assign the resulting token feature to each point in its patch.

A.5 Point Cloud Registration With PTv3

In Tab. 7, we compare point cloud encoders using a shared registration backend. Utonia and Concerto use hierarchical PTv3 encoders. Similar to what they do for linear probing, we upsample features from all encoder stages to the finest resolution and concatenate them. We then ℓ_2-normalize the resulting per-point descriptors for registration. This leads to poor registration performance in our experiments, since deeper stages with larger feature norms contribute more to descriptor similarity. We therefore ℓ_2-normalize each stage's features before concatenation, followed by normalization of the concatenated descriptor. This increases registration success rate on 3DMatch/3DLoMatch from 62.0/18.5% to 76.5/36.3% for Utonia and from 54.4/13.4% to 65.0/22.7% for Concerto. We report these stage-normalized results in Tab. 7.

A.6 Representation Artifacts

Fig. 7 shows the distribution of token ℓ_2 norms under our artifact-mitigation steps. Without registers or voxel averaging, the representation contains a pronounced tail of high-norm tokens. Adding register tokens removes many of these outliers, while averaging voxel features within each token suppresses most of the remaining tail. This supports our observation that representation artifacts arise both from isolated high-norm tokens and from feature magnitude scaling with voxel occupancy.

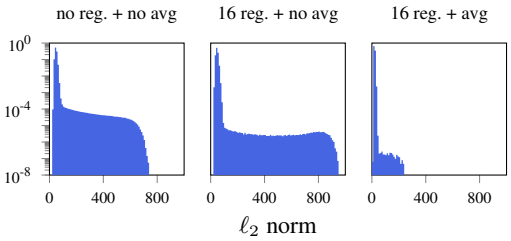

Figure 7: **Token Norm Histograms.** Distribution of token ℓ_2 norms, with frequency shown on a logarithmic scale. Registers remove many high-norm outliers; voxel averaging removes most of the remaining tail.



Figure 9: **Qualitative results on Replica.** VINO features of a large indoor scene from Replica (Straub et al., 2019). We visualize PCA on the bottom left and cosine similarity maps between tokens marked with a red cross and the other 3D tokens.

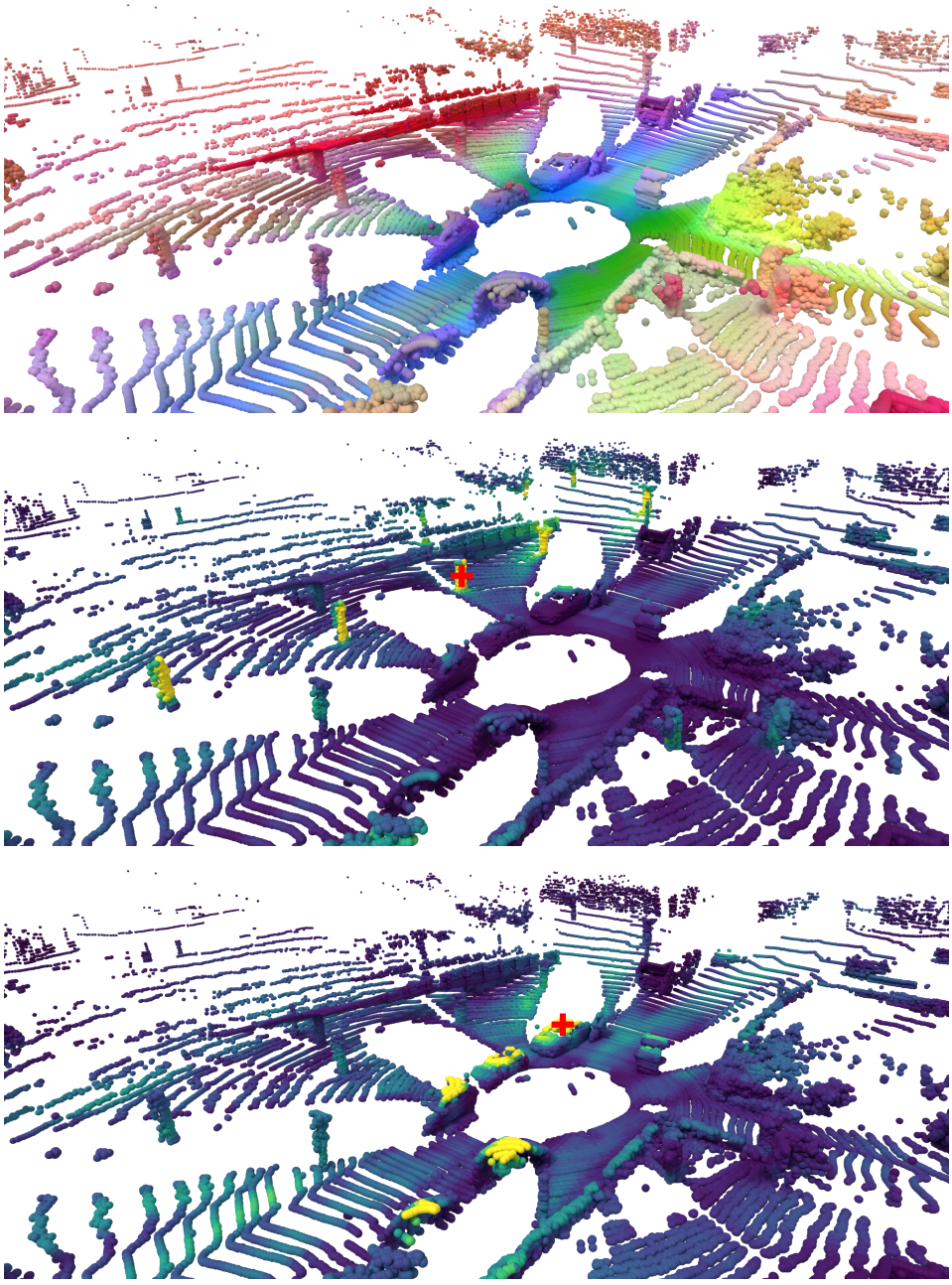

Figure 10: **Qualitative results on KITTI.** features of an outdoor scene from KITTI (Geiger et al., 2012). We visualize PCA on the top and cosine similarity maps between tokens marked with a red cross and the other 3D tokens.