Vernata: Self-Supervised Learning of LiDAR Point Representations

Oliver Lemke[1,2], Alexander Liniger[1], Abel Gawel[1], Marco Hutter[1,2]

***Abstract*— LiDAR serves as a primary sensing modality for robots operating in outdoor environments. However, the performance of deep learning models in this domain is severely limited by the scarcity of labeled data, a direct result of the high cost of 3D annotation. Self-supervised learning addresses this scarcity by learning general-purpose features from unlabeled data. In this work, we present a multi-modal, multi-teacher distillation framework for self-supervised learning on outdoor LiDAR point clouds. Building upon the Sonata architecture, we introduce Vernata, consisting of three extensions: sparse view augmentation to improve robustness against varying point densities, a memory bank mechanism to stabilize resource-constrained training, and cross-modal distillation utilizing dense, high-resolution 2D image features to enable fine-grained semantic guidance. We evaluate our method on the GrandTour, TartanGround, and Waymo datasets, as well as data collected from our own robotic platforms. Our experiments demonstrate a significant performance improvement over Sonata baselines, yielding mIoU scores of 54.7 on TartanGround (+5.9 points, +12.1%) and 57.1 on Waymo (+7.3 points, +14.7%). Finally, we show that the self-supervised approach maintains strong performance even in reduced-modality settings (lacking color or normals), achieving competitive mIoU scores of 49.4 and 50.2 on the respective datasets. Our implementation is publicly available at https://github.com/rai-opensource/vernata.**

I. INTRODUCTION

Autonomous robots are increasingly transitioning from research demonstrations in controlled environments to operating in the unstructured real world [1], [2]. This progress is underpinned by advancements in scene understanding, which equip agents with the semantic context required to perform complex, intelligent actions in unstructured environments [3]–[8]. LiDAR sensors are central to this advanced perception stack. Due to their ability to construct metric 3D point clouds [9], they have emerged as primary sensors in robotics applications [10]–[13], frequently fused with camera image data to create a comprehensive semantic-geometric understanding of the robot's surroundings [14], [15].

Driven by fundamental scaling laws [16]–[19], the machine learning landscape has shifted towards massive data and compute, a transition that has already revolutionized 2D image understanding [20]–[22]. Yet, the 3D domain has not fully benefited from this trend, as traditional supervised learning on point clouds is heavily constrained by the prohibitive cost of manual data annotation [23], [24]. Self-supervised learning (SSL) offers a compelling alternative to this bottleneck, enabling the learning of general-purpose features directly from raw data [20], [21], [25] that can be efficiently adapted to downstream tasks [26]–[28].

[1]Robotics and AI Institute
[2]ETH Zurich
Corresponding author: Oliver Lemke olemke@ethz.ch

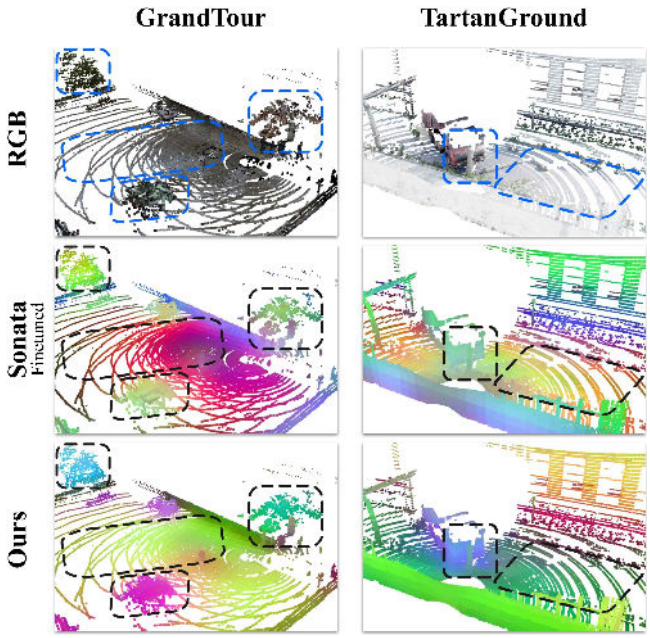

Fig. 1. **Visualization of PCA Features in LiDAR Scenes.** We visualize the principal components of point representations produced by Sonata [31] finetuned on LiDAR datasets, as well as our approach. Sonata struggles with separation between distinct semantic classes, as well as consistency across the inherent density variations of LiDAR point clouds. Our method integrates cross-modal distillation and a sparse-to-dense objective to overcome these limitations, demonstrating sharper semantic distinction, better consistency within object classes, and more robust coherence across planar surfaces like roads, irrespective of the distance to the sensor.

Historically, however, SSL on point clouds has suffered from performance degradation due to representation collapse [29]–[31]. The recent Sonata framework [31] addresses this "geometric shortcut", producing high-quality features from point representations. Motivated by this breakthrough, this work adapts the Sonata approach to the specific domain of outdoor LiDAR point clouds, aiming to enhance performance and robustness in this context.

In this paper, we present Vernata, an extension of the Sonata [31] framework tailored to the challenges of LiDAR point clouds. Specifically, we contribute:

1) A sparse view augmentation strategy to improve inference robustness under varying density conditions,
2) The adaptation of memory banks to 3D point cloud representation learning, stabilizing Sinkhorn-Knopp normalization under limited compute constraints, and
3) The integration of a cross-modal distillation objective using high-resolution image features to enable fine-grained semantic guidance in the LiDAR domain.

We validate our approach via linear probing for semantic segmentation across multiple datasets, including GrandTour [13], TartanGround [32], Waymo [12], and a custom real-world collection. To enable evaluation on TartanGround, we define a custom protocol by selecting scenes with reliable semantic labels, creating a class assignment, and setting up a train/validation split. We release this benchmark and evaluation protocol to the public. Our results demonstrate significant improvements over Sonata baselines, achieving mIoU scores of 54.7 on TartanGround (+5.9 points, +12.1%) and 57.1 on Waymo (+7.3 points, +14.7%). Finally, we show that the SSL approach maintains high performance even in reduced-modality settings, such as when lacking color or normals, achieving respective mIoU scores of 49.4 and 50.2.

II. RELATED WORK

Self-Supervised Learning. Motivated by findings that model performance scales effectively with data and compute [16]–[19], the field has increasingly adopted self-supervised learning. Learning representations directly from raw data, SSL allows models to utilize massive unlabeled datasets for scaling [35]. Early approaches largely relied on contrastive objectives, such as SimCLR [25] and MoCo [36], which enforce feature invariance across augmented views of the same sample, while pushing apart representations from different samples. Departing from contrastive instance discrimination, SwAV [37] introduced an online clustering mechanism that enforces consistency by comparing prototype assignments between views. Subsequently, Masked Image Modeling (MIM), exemplified by MAE [38], demonstrated that reconstructing masked portions of an input encourages deep structural understanding. More recently, self-distillation methods like DINO [39] and iBOT [40] have combined these insights using a student-teacher framework. Here, a student network predicts the output of a momentum-updated teacher from local and masked views, a strategy that has successfully scaled to foundation models like DINOv2 [20] and DINOv3 [21]. Our work employs a DINOv2-style architecture while adapting the SwAV [37] memory bank design for point clouds, showing its efficacy in stabilizing limited compute SSL training in the 3D domain.

Self-Supervised Learning on Point Clouds. Given the high cost of 3D annotation [23], [24], self-supervised learning represents a particularly promising path towards scalable scene understanding in 3D. Early works in this domain mirror the development of the wider SSL field, first focusing on contrastive learning [41] followed by masked modeling [30]. Recently, however, Sonata [31] identified a critical failure mode in these prior methods, coined the "geometric shortcut": models regress trivial geometric cues like point height rather than capturing deep semantic meaning. To address this, Sonata explicitly focuses on obscuring fine-grained spatial information to prevent the model from relying on trivial geometry. Most notably, the authors remove the decoder network to eliminate high-resolution skip connections that inherently leak geometric details, instead opting for parameter-free up-casting. In tandem with these architectural changes, Sonata adopts a DINOv2-style [20] self-distillation framework, utilizing a student-teacher setup to enforce local-to-global and masked-to-global consistency. Motivated by the success of this paradigm and the recent release of large-scale unlabeled datasets like GrandTour [13], we aim to extend this approach to the outdoor LiDAR domain. However, this transition is not trivial. Unlike indoor captures, LiDAR scans exhibit strong range-dependent density variations, overall lower spatial resolution, and often lack auxiliary modalities such as color or reliable surface normals. To bridge this domain gap, we introduce a sparse-to-dense matching objective to mitigate extreme density variations, alongside high-resolution cross-modal distillation from 2D Vision Foundation Models (VFMs) to encourage semantically consistent representations. We also investigate the effect of auxiliary modalities on model performance.

Cross-Modal Distillation. Integrating 2D semantic information into 3D backbones is a well-established strategy to boost performance [14], [42]. Beyond explicit fusion [42]–[45], cross-modal distillation aims to transfer representations from VFMs to 3D networks without requiring paired images at inference time. Early methods like PPKT [46] used pixel-to-point contrastive learning, while SLidR [47] improved alignment using superpixels. Most recently, ScaLR [48] demonstrated that simple cosine similarity-based objectives are sufficient for effective scaling when supported by three key pillars: high-capacity 2D teachers, capable 3D students, and diverse training data, a conclusion subsequently corroborated by DITR [49]. Our approach integrates cross-modal guidance into the Sonata framework. Recently, Concerto [50] similarly augmented Sonata with 2D feature distillation, matching 3D points to low-resolution DINOv2 [20] patches via spatial averaging in the indoor setting. In contrast, we explore an alternative tailored for the LiDAR domain by investigating point-wise distillation with high-resolution features to preserve fine-grained semantics, especially at range. Furthermore, rather than bilinearly interpolating the raw low-resolution patches as in ScaLR [48], we first upsample the features with a learned module [34]. Ultimately, these works mutually reinforce the value of leveraging 2D VFMs to enhance 3D representation learning.

III. METHOD

We present Vernata, a multi-modal, multi-teacher distillation framework designed to learn robust point representations from LiDAR data. Building upon Sonata [31], our approach introduces three extensions to overcome inherent density variations and limited semantic cues in outdoor LiDAR, as well as the practical bottleneck of resource-constrained training. An overview is shown in Fig. 2.

A. Preliminaries: The Sonata Framework

Architecturally, Sonata [31] adapts a DINOv2-style self-distillation framework [20] to the 3D domain. It employs a dual-network structure consisting of a student and a teacher backbone, where the teacher's parameters are updated via an exponential moving average (EMA) of the student's.

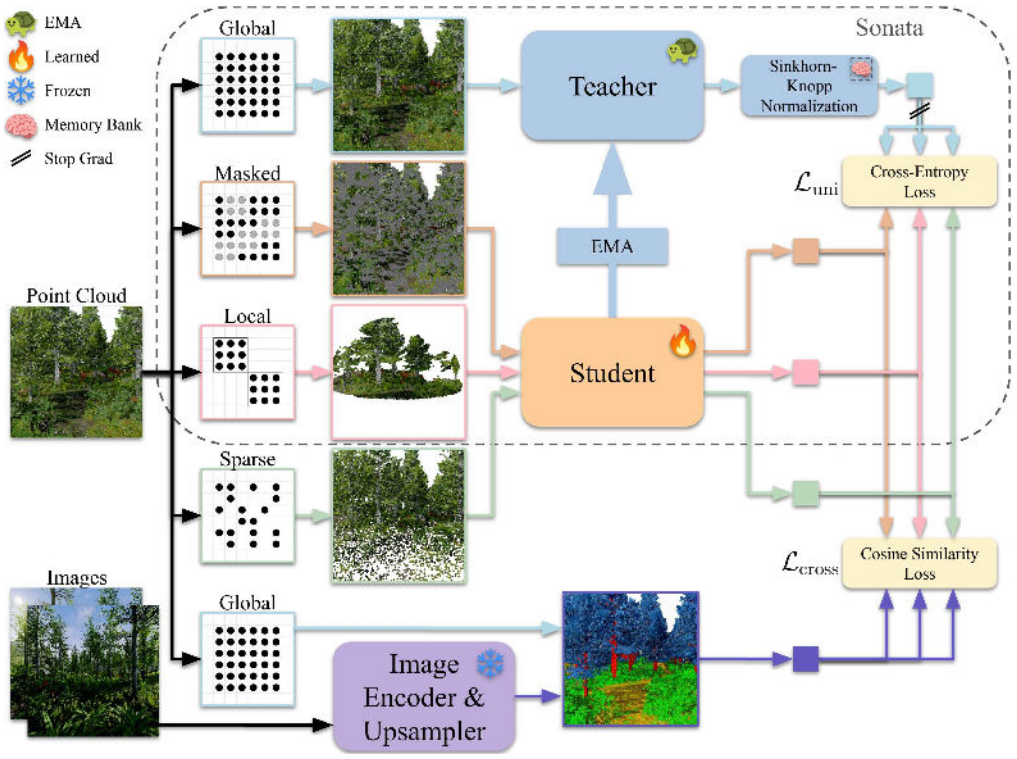

Fig. 2. **Overview of the Framework.** We present a multi-modal multi-teacher distillation architecture for self-supervised point cloud learning. Building upon the Sonata [31] framework, highlighted in gray, our model is anchored by two teachers processing global views of the scene: A 3D teacher (*top*), employing a PTv3 [33] encoder and updated via EMA, and a frozen 2D teacher (*bottom*), producing DINOv2-S [20] features, upsampled with LoftUp [34], and backprojected into the point cloud. In contrast, the student receives three variations of the global view: masked views, local crops, and sparse subsampled views. The student embeddings are aligned with the 3D teacher via cross-entropy loss and with the 2D teacher via a cosine similarity-based loss.

View Generation and Objective. The training process generates distinct inputs for each branch: The teacher receives a set of "global" views $\mathcal{V}_T = \{t_1, t_2\}$, which are large crops covering 40%-100% of the scene. The student receives a set of partial views $\mathcal{V}_S = \mathcal{V}_{\text{mask}} \cup \mathcal{V}_{\text{loc}}$, consisting of "masked" global views and "local" crops (5%-40% coverage). To align representations, the student minimizes the cross-entropy between its predicted prototype probabilities $P^{(s)}$ and the teacher's target assignments $Q^{(t)}$ for spatially aligned points. The targets $Q^{(t)}$ are computed by mapping the teacher's features to learnable prototypes and normalizing the resulting scores via the Sinkhorn-Knopp (SK) algorithm [37], [51]. The overall uni-modal objective is defined as:

$$\mathcal{L}_{\text{uni}} = \sum_{s \in \mathcal{V}_S} \lambda_s \sum_{t \in \mathcal{A}(s)} \mathcal{L}_{\text{CE}}(Q^{(t)}, P^{(s)}). \tag{1}$$

Here, $\mathcal{A}(s)$ defines the set of teacher views t used as targets for a given student view s, and λ_s the associated loss weights. Fig. 2 highlights the Sonata framework in gray.

Geometric Shortcut. Sonata not only adapts the DINOv2 architecture to 3D, but also designs its approach specifically to address the "geometric shortcut". The authors identify this as a failure mode unique to 3D SSL, where models collapse to trivial low-level geometry, such as point height or surface normals, rather than learning semantic structures. To prevent this, Sonata actively obscures fine-grained geometric details by eliminating the standard hierarchical decoder in favor of feature up-casting. In addition, it employs targeted data augmentations, as well as progressive parameter schedules that gradually increase task difficulty and stabilize training.

B. View Generation with Sparse Augmentation

Our preliminary exploration indicates that Sonata-style models are highly sensitive to point density variations. While this issue is negligible for indoor datasets like ScanNet [24] with uniform density, LiDAR data exhibits a quadratic density decay with range. The model implicitly encodes local sparsity patterns, introducing a range-dependent bias that causes representations to drift based on their distance from the sensor. We illustrate this phenomenon in Fig. 3.

To mitigate this, we introduce a Sparse View augmentation. We simulate sparsity by uniformly subsampling the global view with a ratio $r_s \sim \mathcal{U}(r_{\min}, r_{\max})$. The student is tasked with aligning its representation of the sparse view with the teacher's dense view. This sparse-to-dense regularization encourages the model to learn density-invariant features, critical for consistent perception across density ranges. Formally, we extend the student view set to $\mathcal{V}_S = \mathcal{V}_{\text{mask}} \cup \mathcal{V}_{\text{loc}} \cup \mathcal{V}_{\text{sparse}}$. Similar to the masked views, sparse views are supervised by the full set of teacher views $\mathcal{A}(s) = \{t_1, t_2\}$ to maximize the supervision signal. In alignment with Sonata, loss weights are chosen such that each view is weighted equally, and we employ a progressive scheduler, decreasing $(r_{\min}, r_{\max})$ from $(0.9, 1.0)$ to $(0.5, 0.7)$ during training.

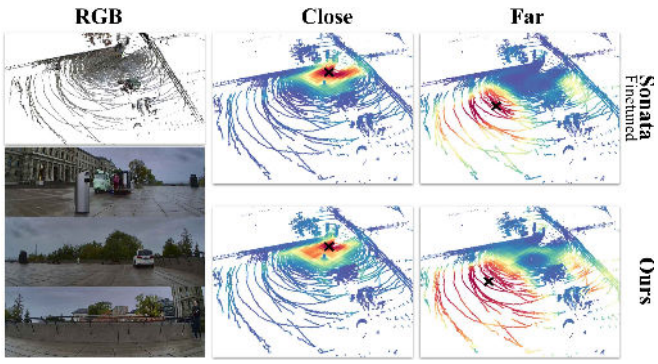

Fig. 3. **Sparsity Encoding Phenomenon**. We show cosine similarity maps of a scene (left) for a reference point close (middle) and far away (right) from the robot. Despite the uniform terrain at both references, the embeddings diverge. Although our method (bottom) demonstrates improved consistency, particularly for distant points, the distinction between dense and sparse regions remains clearly visible. Reference highlighted with a cross.

C. Resource-Efficient Self-Distillation

Our objective minimizes the cross-entropy between the student's prototype probability predictions and the teacher's SK-normalized prototype assignments [37]. While SK normalization prevents mode collapse, it benefits from a large batch size to accurately estimate the global prototype distribution, a scale Sonata achieves using 32 GPUs. To stabilize training on only 4 GPUs, we utilize a memory bank adapted from SwAV [37]. We maintain a FIFO queue of prototype scores S_{bank} and concatenate the current batch's B point-wise scores $S^{(t)}_{\text{batch}}$ with this queue prior to SK normalization:

$$Q^{(t)} = \text{SinkhornKnopp}(\text{Concat}(S^{(t)}_{\text{batch}}, S_{\text{bank}}))[:B]. \quad (2)$$

This effectively decouples the normalization statistics from the batch size, stabilizing assignment on limited compute.

D. Cross-Modal Distillation

To further improve our representations, we add a secondary cross-modal distillation objective. Leveraging image features derived from frozen Vision Foundation Models offers two advantages: they provide robust representations due to their massive pretraining scales, and they serve as stable targets to guide the 3D student's learning [48]–[50]. To backproject the image features onto each point, we use known camera intrinsics and extrinsics. However, perspective projection causes a single low-resolution feature patch to cover an increasingly large 3D volume at long ranges. Therefore, instead of backprojecting the DINOv2 [20] patches directly, we first upsample them with LoftUp [34] (224 px short side) before bilinearly sampling at the projected point coordinates. We thus obtain a high-resolution feature $h^{(t)}_j$ per point j and teacher view t. Let $z^{(s)}_i$ be the student's feature associated with point i in student view s, projected via a small MLP to match the feature dimension of the teacher, and $\tilde{\cdot}$ denote ℓ_2-normalization. Following [48], we define the distillation loss as:

$$\mathcal{L}_{\text{sim}}(s,t) = \frac{1}{|\mathcal{M}_{s,t}|} \sum_{(i,j)\in\mathcal{M}_{s,t}} \|\tilde{h}^{(t)}_j - \tilde{z}^{(s)}_i\|_2. \quad (3)$$

$\mathcal{M}_{s,t} = \{(i,j)\}$ defines the set of nearest-neighbor spatial assignments between the two views. The total cross-modal loss follows (1):

$$\mathcal{L}_{\text{cross}} = \sum_{s\in\mathcal{V}_S} \lambda_s \sum_{t\in\mathcal{A}(s)} \mathcal{L}_{\text{sim}}(s,t). \quad (4)$$

Finally, the overall training objective is a simple sum of both losses, requiring no hyperparameter tuning:

$$\mathcal{L} = \mathcal{L}_{\text{uni}} + \mathcal{L}_{\text{cross}}. \quad (5)$$

IV. EXPERIMENTS

We evaluate our framework across two domains: unstructured field environments and urban driving. In the former, we combine the real-world GrandTour [13] and synthetic TartanGround [32] for self-supervised pretraining. However, as GrandTour lacks semantic annotations, we perform downstream evaluation exclusively on TartanGround. For the latter, we utilize the Waymo Open Dataset [12] as a self-contained benchmark, employing it for both pretraining and evaluation to validate our method on an established dataset.

TartanGround Standardization. As TartanGround lacks a canonical evaluation protocol, we establish one for semantic segmentation. First, to mitigate LiDAR sparsity, we accumulate point clouds over 3 frames during preprocessing. Second, we map the 1496 scene-specific raw labels to 7 canonical semantic classes (e.g., *Manmade Surface*, *Natural Surface*, *Static Obstacles*, *Vegetation*) and define a stratified training/validation split. Third, we establish distinct dataset scales for our two training phases. To maximize variety during self-supervised pretraining, we utilize a broad dataset of 24,722 samples. For linear probing, we restrict the dataset to a curated subset of 6,501 samples to ensure the network learns from only the highest-quality semantic labels.

Custom Dataset. To evaluate the real-world applicability of our model, we introduce a small, in-house annotated dataset. This dataset comprises 311 labeled frames collected across four distinct trail trajectories: two sourced from GrandTour [13] ("Grindelwald - Canyon" 1 and 2) and two that are self-collected. The labeling consists of 30 classes tailored to our use case, such as *Path*, *Vegetation*, *Person*, or *Skill Element*. We reserve one of the self-collected trajectories as a hold-out validation set for quantitative evaluation.

Implementation Details. We initialize all our models from a ScanNet [24]-pretrained Sonata checkpoint, as it is the only publicly available pretrained checkpoint at this time. Subsequently, we first perform self-supervised pretraining on our target datasets, followed by linear probing for semantic segmentation. We report mean Intersection-over-Union (mIoU), mean Accuracy (mAcc), and overall Accuracy (Acc). For details regarding hyperparameter setup, please refer to Table VI in the Appendix. To manage the significant computational overhead of self-supervised pretraining and validation across diverse datasets, we restrict all point clouds to a 50 m $\times$ 50 m region centered around the sensor, voxelized with a grid size of 0.1 m during training.

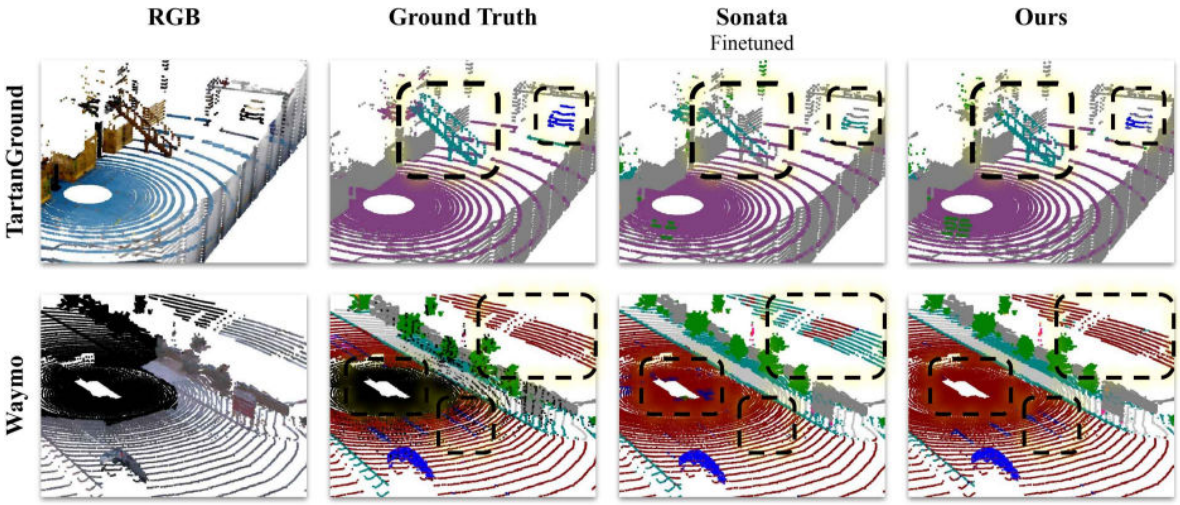

Fig. 4. **Qualitative Results.** We compare qualitative segmentation results against the finetuned Sonata variant on TartanGround (top) and Waymo (bottom). In TartanGround, Vernata demonstrates significant improvements in segmenting challenging classes (stairs), alongside better performance in sparse regions (distant vehicle). However, both models exhibit limitations in resolving the walkway and occasionally misclassify structural elements as vegetation. In Waymo, we achieve more accurate road segmentation especially in the dense and sparse regions, as well as noticeably cleaner lane markers.

TABLE I

SEMANTIC SEGMENTATION VIA LINEAR PROBING. WE EVALUATE VERNATA AGAINST THE ORIGINAL SONATA [31] BASELINE (PRETRAINED ON SCANNET [24]) AND A DOMAIN-ADAPTED VERSION OF SONATA (SELF-SUPERVISED FINETUNED ON THE RESPECTIVE LIDAR DATASETS). PERFORMANCE IS REPORTED USING MIOU, MACC, AND ACC. BY ADDRESSING LIDAR SPARSITY AND MODALITY GAPS, OUR APPROACH CONSISTENTLY OUTPERFORMS BOTH BASELINES.

	TartanGround			**Waymo**		
Method	mIoU	mAcc	Acc	mIoU	mAcc	Acc
Sonata [31]	48.7	60.6	83.6	43.7	58.7	85.7
Sonata [31] Finetuned	48.8	63.0	83.8	49.8	63.3	89.1
Ours	**54.7**	**69.0**	**87.0**	**57.1**	**69.0**	**91.5**

TABLE II

ABLATION OF CONTRIBUTIONS. WE ABLATE THE SPARSE VIEW (SP), SINKHORN-KNOPP MEMORY BANK (MB), AND CROSS-MODAL DISTILLATION (CMD) EXTENSIONS. ACROSS BOTH DATASETS, WE OBSERVE MODERATE GAINS USING SP AND MB, WHILE THE LARGEST GAINS STEM FROM CMD. COMBINING ALL EXTENSIONS YIELDS THE BEST MIOU ACROSS BOTH DATASETS.

SP	MB	CMD	**TartanGround**		**Waymo**	
			mIoU	mAcc	mIoU	mAcc
-	-	-	48.8	63.0	49.8	63.3
✓	-	-	49.6	64.5	50.9	64.9
-	✓	-	50.8	65.8	51.0	65.1
✓	✓	-	51.6	66.2	51.8	65.8
-	-	✓	*53.4*	*67.4*	*56.5*	**69.5**
✓	✓	✓	**54.7**	**69.0**	**57.1**	*69.0*

A. Main Results

We benchmark our method against two self-supervised baselines: the official Sonata [31] checkpoint (frozen, pretrained on ScanNet [24]) and a Sonata variant finetuned on our respective target datasets. As shown in Table I, our approach consistently outperforms both baselines across all metrics. On TartanGround, we observe a gain of +5.9 mIoU (+12.1%), while on the Waymo dataset, we improve by +7.3 mIoU (+14.7%) over the finetuned baseline. Qualitative results further reinforce these findings (see Fig. 1 and Fig. 4). Consistent across both datasets, Vernata demonstrates more robust segmentation, especially within the nearby dense and distant sparse regions, while significantly improving the detection of challenging semantic classes. Note that the quantitative results are not directly comparable to those cited in Sonata [31], as we only finetune from a ScanNet [24] checkpoint instead of training from scratch, consider only a 50 m $\times$ 50 m subsection of the scene, use a coarser downsampling grid for inference, and omit test-time augmentation.

B. Ablation on Proposed Extensions

We investigate the contribution of each component in Table II. The Sparse View (SP) augmentation provides consistent gains, validating its effectiveness in handling the inherent density variations of LiDAR scans. The Sinkhorn-Knopp Memory Bank (MB) further boosts performance by stabilizing the training signal during small-batch training. Finally, Cross-Modal Distillation (CMD) yields the single largest improvement across both datasets, confirming that 2D semantic guidance is highly valuable for scene understanding. It is, however, slightly less effective on TartanGround (+4.6 mIoU) compared to Waymo (+6.7 mIoU), likely due to the domain gap between synthetic images and the real-world DINOv2 pretraining data. In summary, the final variant incorporating all extensions yields the highest mIoU across both datasets, with only a small mAcc dip on Waymo, which we attribute to a minor class-balancing trade-off where false positives are reduced at the cost of negligible accuracy.

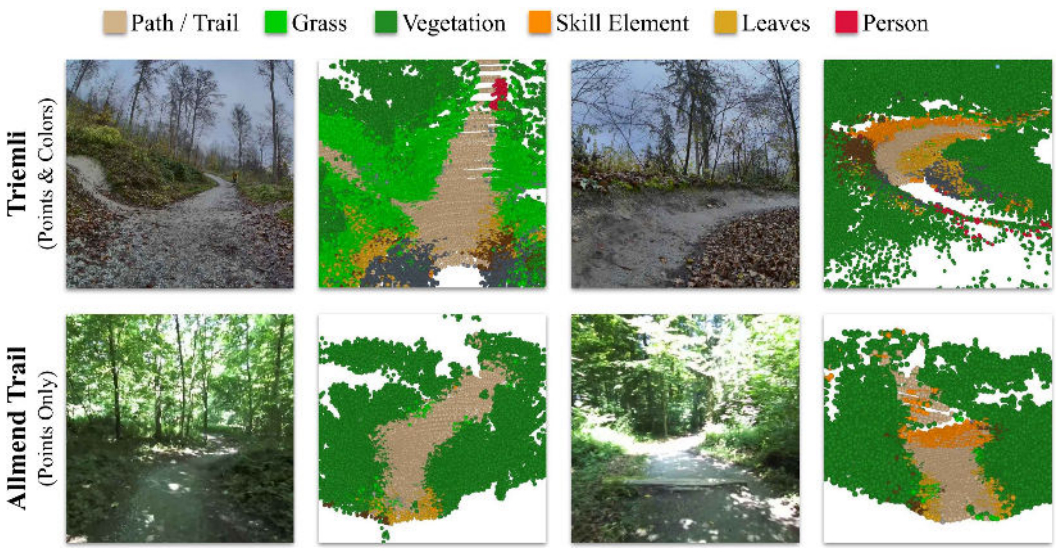

Fig. 5. **Inference on Real-World Trajectories.** We present inference results for our CMD-less model variant, pretrained on GrandTour and TartanGround, with a linear head finetuned on an in-house dataset of just 311 frames. The top row displays inference on the GrandTour Triemli scene using only coordinates and colors, while the bottom row shows a trajectory from RAI's Ultra Mobility Vehicle (UMV) [52] using only point coordinates, no colors or surface normals. Visualized point clouds are accumulated over 5 frames. Both models exhibit robust path detection, capturing humans and skill elements despite constrained modalities and lower LiDAR resolution.

TABLE III

COMPARISON WITH PTV3 BASELINE. WE COMPARE OUR SELF-SUPERVISED APPROACH AGAINST THE FULLY SUPERVISED PTV3 BASELINE. n DENOTES THE NUMBER OF TRAINING SAMPLES. WHILE THE FULLY SUPERVISED BASELINE DOMINATES ON LARGE-SCALE DATASETS LIKE WAYMO, OUR METHOD SIGNIFICANTLY CLOSES THE GAP ON SMALLER DATASETS (TARTANGROUND) AND ACHIEVES HIGHER MIOU AND MACC IN EXTREMELY LOW-DATA REGIMES (CUSTOM).

	TartanGround ($n = 6,501$)		**Waymo** ($n = 23,691$)		**Custom** ($n = 220$)	
Method	mIoU	mAcc	mIoU	mAcc	mIoU	mAcc
PTv3 [33]	**56.2**	65.4	**68.3**	**77.6**	19.6	24.6
Ours	54.7	**69.0**	57.1	69.0	**22.1**	**29.3**

TABLE IV

INPUT MODALITY ROBUSTNESS. WE EVALUATE A CMD-LESS VERNATA VARIANT BY SEQUENTIALLY ABLATING SURFACE NORMALS AND COLOR FEATURES. NOTE THAT TO MITIGATE COMPUTATIONAL COSTS, CROSS-MODAL DISTILLATION WAS OMITTED FROM THIS SET OF EXPERIMENTS. AS EXPECTED, MIOU CONSISTENTLY DECREASES AS THESE AUXILIARY MODALITIES ARE REMOVED. HOWEVER, THE MODELS MAINTAIN COMPETITIVE PERFORMANCE OVERALL.

Coords	Color	Normals	**TartanGround**		**Waymo**	
			mIoU	mAcc	mIoU	mAcc
✓	✓	✓	**51.6**	**66.2**	**51.8**	**65.8**
✓	✓	-	50.2	62.4	51.2	64.9
✓	-	-	49.4	62.0	50.2	63.5

C. Comparison to Supervised Approaches

We compare our method to fully supervised approaches to evaluate SSL pretraining in small data regimes. Table III compares a fully supervised PTv3 [33] baseline, trained from scratch, to a linear head trained on our SSL-pretrained models. Benefiting from a 7.5M-parameter decoder compared to our lightweight 9–37k-parameter linear head, PTv3 heavily outperforms our approach on the larger Waymo dataset ($n = 23,691$), leading by 11.2 mIoU. However, this performance gap narrows significantly on the smaller TartanGround dataset ($n = 6,501$). On TartanGround, PTv3's lead shrinks to just 1.5 mIoU, and our SSL approach actually surpasses the fully supervised baseline by 3.6 mAcc. Finally, on our custom dataset with very few examples ($n = 220$), the SSL approach demonstrates its strength in low-data regimes by outperforming the fully supervised baseline by 2.5 mIoU and 4.7 mAcc. These results demonstrate that as the availability of annotated data decreases, the representations learned during large-scale pretraining allow SSL models to surpass supervised baselines. To further illustrate this real-world generalization, Fig. 5 presents real-world inference results from our CMD-less model finetuned on this custom dataset. Whether evaluated on a GrandTour trail (top row) or a trajectory captured via RAI's UMV [52] (bottom row), the model exhibits robust path detection and accurately captures dynamic obstacles like humans, even when operating under constrained modalities without normals or color.

D. Robustness to Reduced Modalities

The standard architecture expects 3 inputs: coordinates, colors, and surface normals. However, in robotics, high-quality normals or omnidirectional calibrated camera coverage is often unavailable. Thus, we evaluate our model trained on a reduced modality set. Because we omit cross-modal distillation for this experiment due to the additional training overhead, we refer to this as the CMD-less variant, which is

TABLE V

COMPARISON OF DISTILLATION MECHANISM. TO COMPARE OUR HIGH-RESOLUTION FEATURE DISTILLATION AGAINST CONCERTO'S [50] PATCH-BASED APPROACH, WE EVALUATE MODELS PRETRAINED WITH ONLY THE CROSS-MODAL DISTILLATION EXTENSION. WHILE BOTH APPROACHES PERFORM ON PAR OVERALL, OUR DENSE MATCHING IS MORE ROBUST AT LONGER RANGES, ACHIEVING HIGHER MIOU ON LIDAR POINTS BEYOND 20 METERS (DENOTED AS 20M+) IN BOTH TARTANGROUND (TG) AND WAYMO.

	TG		**Waymo**		**TG**$_{20m+}$		**Waymo**$_{20m+}$	
Method	mIoU	mAcc	mIoU	mAcc	mIoU	mAcc	mIoU	mAcc
Patch	**53.4**	67.2	**56.9**	**69.7**	48.0	**55.4**	50.1	63.5
Ours	**53.4**	**67.4**	56.5	69.5	**48.2**	55.2	**50.7**	**64.4**

also showcased in Fig. 5. Encouragingly, Table IV shows that our model retains strong performance even without colors or normals, achieving 49.4 mIoU on TartanGround and 50.2 mIoU on Waymo using only coordinates.

E. Comparison to Patch-based Distillation

We compare the patch-based distillation mechanism proposed in Concerto [50] to our high-resolution feature distillation. As shown in Table V, both methods perform on par overall, with marginal trade-offs across datasets. However, it also reveals a performance shift at longer ranges: when evaluating exclusively on LiDAR points beyond a 20-meter radius, our mechanism demonstrates an advantage. We attribute this to our high-resolution matching mitigating the geometric footprint expansion of low-resolution patches at a distance. This demonstrates that upsampled feature association holds promise for improving point representations especially at long ranges, and we leave a comprehensive investigation into these architectural trade-offs for future work.

V. CONCLUSION

In summary, this work introduces Vernata, a framework for self-supervised learning on outdoor LiDAR point clouds. By extending the Sonata architecture with sparse view augmentation, Sinkhorn-Knopp memory banks, and cross-modal distillation, we effectively address the unique challenges of the outdoor domain. Our evaluations show that combining these techniques maximizes performance across diverse datasets, with distillation proving exceptionally performant on real-world data like the Waymo Open Dataset. Furthermore, ablation studies confirm that Vernata maintains robust inference even in limited-modality settings where color and surface normals are unavailable. While our approach demonstrates clear improvements, it currently operates strictly on a per-frame basis. Future work will focus on integrating temporal context to ensure inference consistency and deploying these robust representations for execution on mobile robots.

APPENDIX

Inference. On an NVIDIA A5000, we achieve inference rates (FP32 / FP16) of 8.9 / 16.1 Hz on Waymo, 7.2 / 13.3 Hz on TartanGround, and 8.8 / 16.8 Hz on our own dataset.

Experiment Setup. Table VI shows our hyperparameters.

TABLE VI

HYPERPARAMETER SETUP.

Training Stage	**Parameter**	**Value**
Self-Supervised	Total Steps	20,000
	Batch Size	24 (Per GPU: 1, #GPU: 4, GradAcc: 6)
	GPU	NVIDIA H100 80GB
	Learning Rate	0.0002
	Schedule	Cosine with warmup
	Global View Size	(0.7, 1.0) $\rightarrow$ (0.4, 1.0)
	Local View Size	(0.1, 0.4)
	Masked View	Size: 0.1 $\rightarrow$ 0.4 Ratio: 0.3 $\rightarrow$ 0.7
	Sparse View	(0.9, 1.0) $\rightarrow$ (0.5, 0.7)
	Memory Bank	50,000 (GT, TG) 100,000 (Waymo)
Linear Probing	Training Steps	10,000 (TG, Custom) 20,000 (Waymo)
	Batch Size	24
	Learning Rate	0.001
	Schedule	Cosine with warmup
	Loss	CE & Lovasz [53]

REFERENCES

[1] A. Bouman, M. F. Ginting, N. Alatur, M. Palieri, D. D. Fan, T. Touma, T. Pailevanian, S.-K. Kim, K. Otsu, J. Burdick, and A.-a. Agha-Mohammadi, "Autonomous spot: Long-range autonomous exploration of extreme environments with legged locomotion," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 2518–2525.

[2] F. Favaro, L. Fraade-Blanar, S. Schnelle, T. Victor, M. Peña, J. Engstrom, J. Scanlon, K. Kusano, and D. Smith, "Building a credible case for safety: Waymo's approach for the determination of absence of unreasonable risk," 2023. [Online]. Available: https://arxiv.org/abs/2306.01917

[3] Q. Gu, A. Kuwajerwala, S. Morin, K. M. Jatavallabhula, B. Sen, A. Agarwal, C. Rivera, W. Paul, K. Ellis, R. Chellappa *et al.*, "Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 5021–5028.

[4] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual language maps for robot navigation," *arXiv preprint arXiv:2210.05714*, 2022.

[5] Y. Liu, W. Chen, Y. Bai, X. Liang, G. Li, W. Gao, and L. Lin, "Aligning cyber space with physical world: A comprehensive survey on embodied ai," *IEEE/ASME Transactions on Mechatronics*, 2025.

[6] Y. Ma, Z. Song, Y. Zhuang, J. Hao, and I. King, "A survey on vision-language-action models for embodied ai," *arXiv preprint arXiv:2405.14093*, 2024.

[7] O. Lemke, Z. Bauer, R. Zurbrügg, M. Pollefeys, F. Engelmann, and H. Blum, "Spot-compose: A framework for open-vocabulary object retrieval and drawer manipulation in point clouds," in *2nd Workshop on Mobile Manipulation and Embodied Intelligence at ICRA 2024*, 2024.

[8] T. Behrens, R. Zurbrügg, M. Pollefeys, Z. Bauer, and H. Blum, "Lost & found: Tracking changes from egocentric observations in 3d dynamic scene graphs," *IEEE Robotics and Automation Letters*, 2025.

[9] J. Shan and C. K. Toth, *Topographic laser ranging and scanning: principles and processing*. CRC press, 2018.

[10] J. Zhang, S. Singh *et al.*, "Loam: Lidar odometry and mapping in real-time." in *Robotics: Science and systems*, vol. 2, no. 9. Berkeley, CA, 2014, pp. 1–9.

[11] D. Wisth, M. Camurri, and M. Fallon, "Vilens: Visual, inertial, lidar, and leg odometry for all-terrain legged robots," *IEEE Transactions on Robotics*, vol. 39, no. 1, pp. 309–326, 2023.

[12] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *CVPR*, 2020, pp. 2446–2454.

[13] J. Frey, T. Tuna, F. Fu, K. Patterson, T. Xu, M. Fallon, C. Cadena, and M. Hutter, "Grandtour: A legged robotics dataset in the wild for multi-modal perception and state estimation," *arXiv preprint arXiv:2602.18164*, 2026.

[14] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "Pointpainting: Sequential fusion for 3d object detection," in *CVPR*, 2020, pp. 4604–4612.

[15] Z. Zhuang, R. Li, K. Jia, Q. Wang, Y. Li, and M. Tan, "Perception-aware multi-sensor fusion for 3d lidar semantic segmentation," in *ICCV (ICCV)*, October 2021, pp. 16 280–16 290.

[16] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling laws for neural language models," *arXiv preprint arXiv:2001.08361*, 2020.

[17] Y. Bahri, E. Dyer, J. Kaplan, J. Lee, and U. Sharma, "Explaining neural scaling laws," *Proceedings of the National Academy of Sciences*, vol. 121, no. 27, p. e2311878121, 2024.

[18] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. de Las Casas, L. A. Hendricks, J. Welbl, A. Clark *et al.*, "Training compute-optimal large language models (2022)," *arXiv preprint arXiv:2203.15556*, 2022.

[19] M. Baniodeh, K. Goel, S. Ettinger, C. Fuertes, A. Seff, T. Shen, C. Gulino, C. Yang, G. Jerfel, D. Choe *et al.*, "Scaling laws of motion forecasting and planning–a technical report," *arXiv preprint arXiv:2506.08228*, 2025.

[20] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby *et al.*, "Dinov2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, 2023.

[21] O. Siméoni, H. V. Vo, M. Seitzer, F. Baldassarre, M. Oquab, C. Jose, V. Khalidov, M. Szafraniec, S. Yi, M. Ramamonjisoa *et al.*, "Dinov3," *arXiv preprint arXiv:2508.10104*, 2025.

[22] N. Carion, L. Gustafson, Y.-T. Hu, S. Debnath, R. Hu, D. Suris, C. Ryali, K. V. Alwala, H. Khedr, A. Huang *et al.*, "Sam 3: Segment anything with concepts," *arXiv preprint arXiv:2511.16719*, 2025.

[23] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "Semantickitti: A dataset for semantic scene understanding of lidar sequences," in *ICCV*, 2019, pp. 9297–9307.

[24] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *CVPR*, 2017, pp. 5828–5839.

[25] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PmLR, 2020, pp. 1597–1607.

[26] R. J. Chen, T. Ding, M. Y. Lu, D. F. Williamson, G. Jaume, A. H. Song, B. Chen, A. Zhang, D. Shao, M. Shaban *et al.*, "Towards a general-purpose foundation model for computational pathology," *Nature medicine*, vol. 30, no. 3, pp. 850–862, 2024.

[27] E. Vorontsov, A. Bozkurt, A. Casson, G. Shaikovski, M. Zelechowski, K. Severson, E. Zimmermann, J. Hall, N. Tenenholtz, N. Fusi *et al.*, "A foundation model for clinical-grade computational pathology and rare cancers detection," *Nature medicine*, vol. 30, no. 10, pp. 2924–2935, 2024.

[28] Y. Cong, S. Khanna, C. Meng, P. Liu, E. Rozi, Y. He, M. Burke, D. Lobell, and S. Ermon, "Satmae: Pre-training transformers for temporal and multi-spectral satellite imagery," *Advances in Neural Information Processing Systems*, vol. 35, pp. 197–211, 2022.

[29] J. Hou, B. Graham, M. Nießner, and S. Xie, "Exploring data-efficient 3d scene understanding with contrastive scene contexts," in *CVPR*, 2021, pp. 15 587–15 597.

[30] X. Wu, X. Wen, X. Liu, and H. Zhao, "Masked scene contrast: A scalable framework for unsupervised 3d representation learning," in *CVPR*, 2023, pp. 9415–9424.

[31] X. Wu, D. DeTone, D. Frost, T. Shen, C. Xie, N. Yang, J. Engel, R. Newcombe, H. Zhao, and J. Straub, "Sonata: Self-supervised learning of reliable point representations," in *CVPR*, 2025, pp. 22 193–22 204.

[32] M. Patel, F. Yang, Y. Qiu, C. Cadena, S. Scherer, M. Hutter, and W. Wang, "Tartanground: A large-scale dataset for ground robot perception and navigation," *arXiv preprint arXiv:2505.10696*, 2025.

[33] X. Wu, L. Jiang, P.-S. Wang, Z. Liu, X. Liu, Y. Qiao, W. Ouyang, T. He, and H. Zhao, "Point transformer v3: Simpler faster stronger," in *CVPR*, 2024, pp. 4840–4851.

[34] H. Huang, A. Chen, V. Havrylov, A. Geiger, and D. Zhang, "Loftup: Learning a coordinate-based feature upsampler for vision foundation models," in *ICCV*, 2025, pp. 9913–9923.

[35] J. Gui, T. Chen, J. Zhang, Q. Cao, Z. Sun, H. Luo, and D. Tao, "A survey on self-supervised learning: Algorithms, applications, and future trends," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 12, pp. 9052–9071, 2024.

[36] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *CVPR*, 2020, pp. 9729–9738.

[37] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *Advances in neural information processing systems*, vol. 33, pp. 9912–9924, 2020.

[38] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *CVPR*, 2022, pp. 16 000–16 009.

[39] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *ICCV*, 2021, pp. 9650–9660.

[40] J. Zhou, C. Wei, H. Wang, W. Shen, C. Xie, A. Yuille, and T. Kong, "ibot: Image bert pre-training with online tokenizer," *arXiv preprint arXiv:2111.07832*, 2021.

[41] S. Xie, J. Gu, D. Guo, C. R. Qi, L. Guibas, and O. Litany, "Pointcontrast: Unsupervised pre-training for 3d point cloud understanding," in *ECCV*. Springer, 2020, pp. 574–591.

[42] X. Bai, Z. Hu, X. Zhu, Q. Huang, Y. Chen, H. Fu, and C.-L. Tai, "Transfusion: Robust lidar-camera fusion for 3d object detection with transformers," in *CVPR*, 2022, pp. 1090–1099.

[43] A. Dai and M. Nießner, "3dmv: Joint 3d-multi-view prediction for 3d semantic scene segmentation," in *ECCV*, 2018, pp. 452–468.

[44] W. Hu, H. Zhao, L. Jiang, J. Jia, and T.-T. Wong, "Bidirectional projection network for cross dimension scene understanding," in *CVPR*, 2021, pp. 14 373–14 382.

[45] Z. Zhang, Z. Zhang, Q. Yu, R. Yi, Y. Xie, and L. Ma, "Lidar-camera panoptic segmentation via geometry-consistent and semantic-aware alignment," in *ICCV*, 2023, pp. 3662–3671.

[46] Y.-C. Liu, Y.-K. Huang, H.-Y. Chiang, H.-T. Su, Z.-Y. Liu, C.-T. Chen, C.-Y. Tseng, and W. H. Hsu, "Learning from 2d: Contrastive pixel-to-point knowledge transfer for 3d pretraining," *arXiv preprint arXiv:2104.04687*, 2021.

[47] C. Sautier, G. Puy, S. Gidaris, A. Boulch, A. Bursuc, and R. Marlet, "Image-to-lidar self-supervised distillation for autonomous driving data," in *CVPR*, 2022, pp. 9891–9901.

[48] G. Puy, S. Gidaris, A. Boulch, O. Siméoni, C. Sautier, P. Pérez, A. Bursuc, and R. Marlet, "Three pillars improving vision foundation model distillation for lidar," in *CVPR*, 2024, pp. 21 519–21 529.

[49] K. A. Zeid, K. Yilmaz, D. de Geus, A. Hermans, D. Adrian, T. Linder, and B. Leibe, "Dino in the room: Leveraging 2d foundation models for 3d segmentation," *arXiv preprint arXiv:2503.18944*, 2025.

[50] Y. Zhang, X. Wu, Y. Lao, C. Wang, Z. Tian, N. Wang, and H. Zhao, "Concerto: Joint 2d-3d self-supervised learning emerges spatial representations," *Advances in Neural Information Processing Systems*, vol. 38, pp. 69 498–69 522, 2026.

[51] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," *Advances in neural information processing systems*, vol. 26, 2013.

[52] B. Bokser, D. Gonzalez, A. Preston, A. Bahner, A. Wollschläger, A. Ilvonen, A. Eckert-Erdheim, A. Khadke, B. Hammoud, D. Molinaro *et al.*, "System design of the ultra mobility vehicle: A driving, balancing, and jumping bicycle robot," *arXiv preprint arXiv:2602.22118*, 2026.

[53] M. Berman, A. R. Triki, and M. B. Blaschko, "The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks," in *CVPR*, 2018, pp. 4413–4421.