Sonata: Self-Supervised Learning of Reliable Point Representations

Xiaoyang Wu[1,2] Daniel DeTone[2*] Duncan Frost[2*] Tianwei Shen[2*] Chris Xie[2*] Nan Yang[2*]
Jakob Engel[2] Richard Newcombe[2] Hengshuang Zhao[1] Julian Straub[2]

[1]The University of Hong Kong [2]Meta Reality Labs Research

https://github.com/facebookresearch/sonata

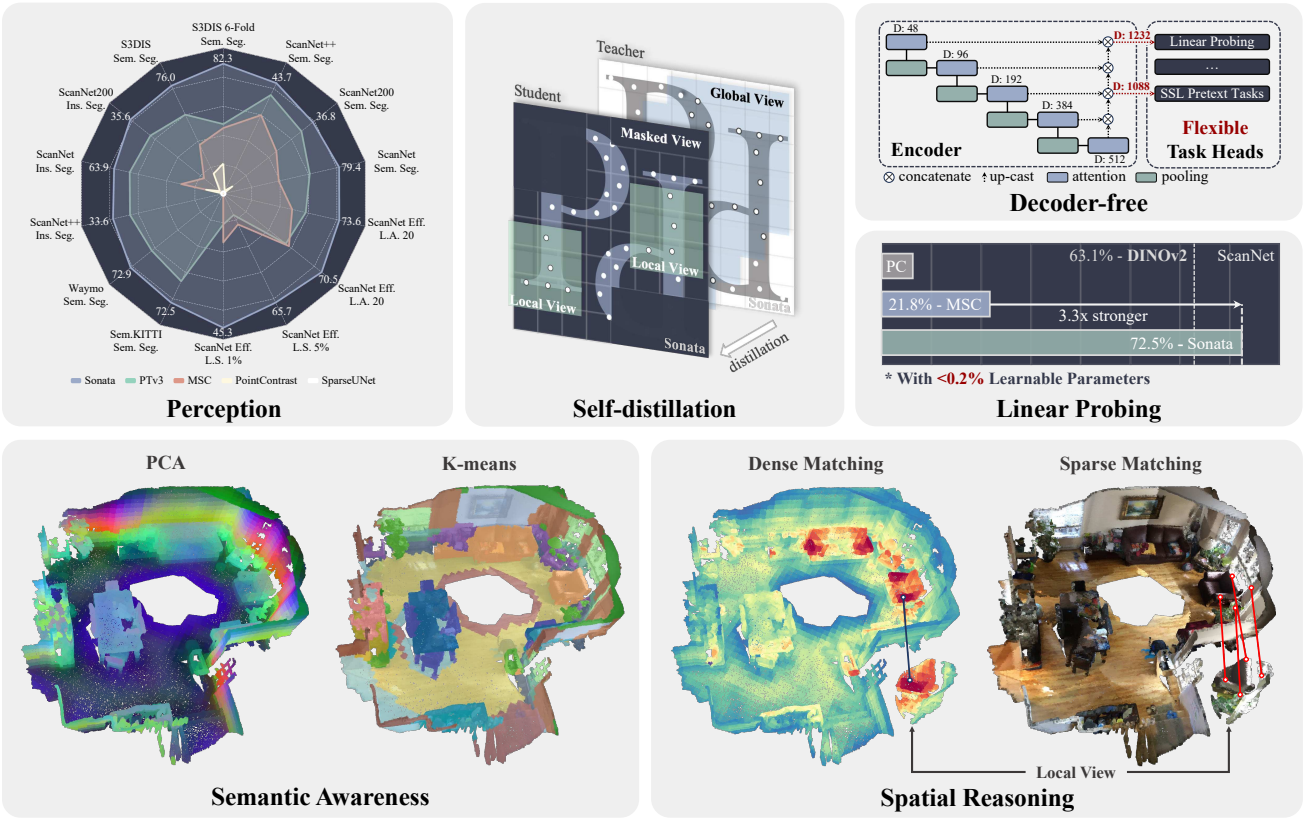

Figure 1. **Main properties.** Sonata leads to reliable 3D self-supervised pretraining with the following superior and emerging properties: 1. ***Perception***. Sonata advances state-of-the-art results across 3D indoor and outdoor perception tasks; 2. ***Linear probing***. With less than 0.2% learnable parameters, Sonata achieves strong and usable linear probing performance which is 3.3× better than previous SOTA; 3. ***Decoder-free***. Sonata moves beyond the inflexible U-Net structure, offering multi-scale representations that unchain future 3D research from previous architectural constraints. 4. ***Semantic awareness***. Sonata reveals semantic structure in PCA and K-means visualizations. 5. ***Spatial reasoning***. Sonata allows spatial correspondence even under strong augmentations as visualized via feature similarity.

Abstract

In this paper, we question whether we have a reliable self-supervised point cloud model that can be used for diverse 3D tasks via simple linear probing, even with limited data and minimal computation. We find that existing 3D self-supervised learning approaches fall short when evaluated on representation quality through linear probing. We hypothesize that this is due to what we term the "geometric shortcut", which causes representations to collapse to low-level spatial features. This challenge is unique to 3D and arises from the sparse nature of point cloud data. We address it through two key strategies: obscuring spatial information and enhancing the reliance on input features, ultimately composing a ***Sonata*** *of 140k point clouds through self-distillation. Sonata is simple and intuitive, yet its learned representations are strong and reliable: zero-shot visualizations demonstrate semantic grouping, alongside strong spatial reasoning through nearest-neighbor relationships. Sonata demonstrates exceptional parameter and data efficiency, tripling linear probing accuracy (from 21.8% to 72.5%) on ScanNet and nearly doubling performance with only 1% of the data compared to previous approaches. Full fine-tuning further advances SOTA across both 3D indoor and outdoor perception tasks.*

*Equal contribution in alphabetic order.

1. Introduction

Self-supervised learning (SSL) with images [2, 13, 34, 94, 104] has seen a continuous increase in model simplicity, capacity, and capability over the past decade [25]. Tuning a single linear layer can achieve performance close to full fine-tuning [31, 35, 104], fostering growing trust in its reliability. This trust has been further strengthened by witnessing the semantic meaning of learned image representations through direct visualization [60, 110]. Consequently, these reliable self-supervised models have become the foundation for emerging approaches [14, 84, 98, 99] across various fields involving images.

In contrast to the image domain, self-supervised learning with point clouds [18, 61, 93, 102, 106] is still in its early stages. Despite the broad reliance on 3D applications in autonomous driving [9, 74], robotic learning [32], mixed reality [5, 43] and egocentric perception [27, 73], the latest self-supervised point cloud models are seldom included in their pipelines. This gap prompts us to consider a simple yet critical question: *do we have a reliable point self-supervised learning approach* that provides strong representations, usable with simple linear probing across these applications? Not yet. Previous SOTAs [38, 88, 93] fall short on this higher-level criterion, achieving only a maximum of 21.8% mIoU on ScanNet [23] with linear probing, especially given that the performance from scratch is 77.6%.

We identify the *geometric shortcut* as the primary issue hindering these prior point cloud SSL approaches from learning reliable representations. This shortcut refers to the tendency of the model to collapse to easily accessible, low-level geometric cues, such as normal direction or point height, as demonstrated in prior works and visualized in Fig. 2. This spatial information is inevitably introduced into point cloud operators along with point coordinates rather than through input features, making it difficult to obscure and nearly impossible to mask effectively.

However, the model collapse caused by geometric shortcuts can be mitigated through two key strategies: *obscuring spatial information* and *emphasizing input features*. Specifically, we address this issue by applying SSL losses at coarser spatial scales, disturbing the spatial information of masked points without features, and progressively increasing task difficulty to reduce reliance on accessible geometric cues. Coupled with a point self-distillation framework and scaling techniques inspired by recent advances in image SSL [11, 12, 34, 60, 110], we ultimately compose a **Sonata** using 140k point cloud scenes [1, 3, 5, 23, 68, 101, 109].

Sonata demonstrates strong zero-shot capabilities, with PCA-colored visualizations of point clouds, k-means clustering of features, and nearest-neighbor matching between point clouds (see Fig. 1). Sonata also proves highly data-efficient, raising semantic segmentation performance from 25.8% to 45.3% under extremely limited data conditions (1% of ScanNet). Additionally, Sonata significantly boosts linear probing accuracy on ScanNet semantic segmentation, increasing it by over 3.3× from 21.8% to 72.5% and surpassing the accuracy of DINOv2 features aggregated onto the point cloud (63.1%). Moreover, combining Sonata features with DINOv2 features further enhances accuracy (76.4%), underscoring that Sonata captures unique 3D information beyond what is visible in images alone. Finally, Sonata achieves state-of-the-art results across various indoor and outdoor perception tasks with full fine-tuning.

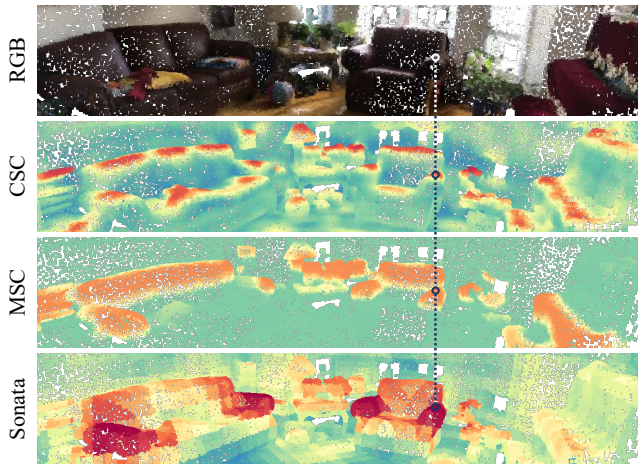

Figure 2. **Geometric shortcut.** We select a point on the sofa arm and compute pairwise similarity with other points. The similarity heatmap reveals that CSC [38] collapses to surface normals, and MSC [88] overfits to point height. In contrast, our Sonata can extract higher-level concepts, as can be seen by the high similarity between all sofa arms highlighted in red.

2. Related Work

Image self-supervised learning. Over the past decade, remarkable advancements [25, 26, 36, 59, 91] have been made in image self-supervised learning, and our research is largely inspired by two pivotal moments in this field. First, linear probing, a method that assesses representation quality by optimizing a minimal linear transformation, has become a standard in 2D image SSL [10, 13, 34, 91]. In some cases, such as when the distribution shift is large, linear probing surpasses full fine-tuning [105]. Second, the ability to directly perceive the semantic meaning of learned representations through zero-shot visualization like PCA or attention [12, 60] has further strengthened trust in reliability.

Point self-supervised learning. Sonata follows the research path initiated by PointContrast [38, 93], emphasizing self-supervised learning with scene-level data [23]. While previous efforts do implement strategies to prevent collapse from geometric shortcuts, they remain limited. For example, Masked Scene Contrast (MSC) [88] encourages learning beyond naive geometric cues by predicting color or normal vectors but still partially anchors representations to predefined tasks. GroupContrast (GC) [80] employs graph-

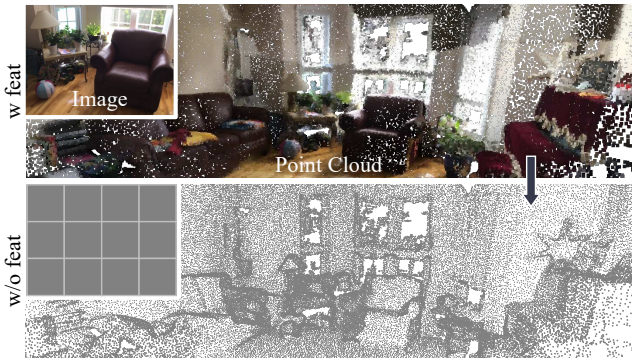

Figure 3. **The geometric shortcut is unique to 3D.** When comparing the information contained in 2D image and 3D point cloud data after removing the input feature (indicated by color), it is evident that in images all information is within the input feature. Whereas point clouds retain geometric information in point positions, which is directly utilized by operators. This characteristic leads to what we term geometric shortcuts in 3D SSL.

based segment guidance [28], though it is constrained by its reliance on human-designed algorithms. Building on MSC, Sonata directly addresses the geometric shortcut and scales up training [90] to establish a more reliable approach to point cloud self-supervised learning.

Point cloud backbones. This area [17, 51, 56, 64, 67, 77] has significantly benefited from the U-Net structure [70], particularly with its hierarchical decoding and skip connections, as first introduced by PointNet++ [66]. However, the tight coupling between the encoder and decoder restricts flexibility and generalization capacity [92]. Sonata addresses this limitation by focusing self-supervised learning exclusively on the encoder, thus removing the hierarchical decoder. Moreover, unlike previous SOTAs [38, 88, 93] in point SSL, which primarily use SparseUNet [17, 20, 29] to balance efficiency and accuracy, we leverage Point Transformer V3 (PTv3) [89], an efficient, accurate, and scalable transformer backbone. This shift alone yields a 7.7% improvement in linear probing performance over MSC [88].

3. Pilot Study and Design Principle

In this section, we qualitatively study the problems of current point self-supervised approaches and point cloud backbones to inform the Sonata approach.

Uncovering the geometric shortcut in point SSL. The history of image self-supervised learning [34, 60, 104, 112] can be summarized as a continuous battle against shortcuts, where models often exploit trivial solutions (mode collapse) rather than understanding deeper semantics. Each advancement has involved identifying, understanding, and overcoming these shortcuts, refining pretext tasks to push models to "struggle" in learning stronger representations.

However in 3D point SSL, despite various attempts [38, 88, 93] to increase the difficulty of pretext tasks, a curious phenomenon persists—the loss consistently reduces rapidly to an ideal range in the early stages of training. We hypothesize that this lack of "struggle" in learning indicates that a shortcut exists that leads to a collapse of the representation to trivial solutions. Indeed as Fig. 2 illustrates, previous 3D SSL approaches seem to learn representations that are sensitive to local surface normals or point height. The collapse to such naive solutions even for well-designed SSL pretext tasks is what we term the *geometric shortcut*.

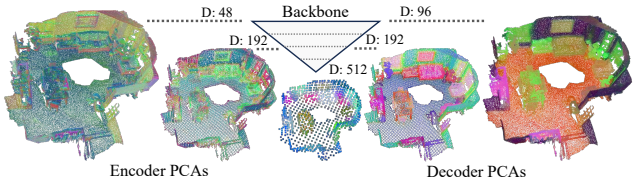

Figure 4. **What is learned by the hierarchical backbone?** We visualize PCA embeddings from different stages of a hierarchical encoder and decoder, trained for *semantic segmentation*. The encoder captures diverse and dispersed feature patterns, indicating a broad range of information. Notably, as the point cloud becomes coarser, accessible geometric information within point coordinates becomes increasingly global. In contrast, the decoder's representations are more uniform and structured, suggesting a focus on refining features for task-specific outputs.

Intuitively, representations affected by this geometric shortcut have not learned sufficient semantics. To quantify this problem, we leverage the standard criteria from 2D SSL of linear probing: we linearly probe the semantic class of each point from the learned representation. As illustrated in Fig. 1, with linear probing, previous 3D SSL methods PC [93] and MSC [88] indeed achieve only 5.6% mIoU and 21.8% mIoU respectively on ScanNet semantic segmentation. When compared to the 63.1% mIoU achieved by 3D-aggregated image representations from DINOv2 [60], it becomes obvious that the current 3D SSL methods do not learn semantic information.

We hypothesize that the geometric shortcut stems from the sparse nature of point cloud data. Every operator, whether for point clouds or images, inherently relies on point (pixel) coordinates to define the kernel. However, unlike images with regularly spaced dense pixels, the sparsity of point cloud data necessitates introducing point coordinate information into point cloud operators rather than just through input features (see Fig. 3). This fact makes it difficult to obscure and nearly impossible to effectively mask out, ultimately resulting in the geometric shortcut.

These observations motivate the training methodology of Sonata aimed to prevent the dominance of naive spatial information inherent in point cloud organization.

Focusing self-supervised learning on encoder only. Following our hypothesis that the geometric arrangement leaks information into the learned representation via the geometric shortcut, we examine the point cloud model. The typical U-Net structure [66, 70] is effective at handling large point clouds in a coarse-to-fine way, but the tight coupling of the encoder and decoder via skip connections restricts

flexibility. Specifically, the decoder enforces per-point features at the original high-resolution scale, with shallow feature channels. This constraint limits the capacity to provide richer representations, which is crucial for SSL as evidenced by large channel dimensions in state-of-the-art 2D approaches [12]. Most importantly, decoding point clouds at the original scale unavoidably introduces local geometric cues into operators facilitating the geometric shortcut.

In Fig. 4, we visualize PCA embeddings from different stages of PTv3 encoder and decoder, trained in a supervised manner for semantic segmentation. We observe that the encoder learns diverse features capturing spatial features at different scales in contrast to the decoder which produces task-specific higher-level representations. Additionally, as the spatial resolution of the point cloud decreases via the max-pooling stages, feature representations become less local. This resolution reduction fundamentally limits reliance on fine spatial information tied to point coordinates.

These observations motivate us to remove the decoder during self-supervised learning for two main reasons. First and foremost, training directly with features at coarser point resolutions inherently restricts access to fine-grained spatial information, reducing the possibility of geometric shortcuts. Second, task-specific features can be probed or fine-tuned on top of a more expressive multi-scale point representation.

4. Point Self-distillation with Sonata

This section details the methodology of Sonata, the point self-distillation framework designed to address geometric shortcuts, remove structural constraints, and deliver strong linear probing results as discussed in Sec. 3. A roadmap of incremental ablation is illustrated in Fig. 6.

4.1. Macro Framework

Before diving into the specific micro designs, we begin with a point self-distillation framework derived from insights gained through previous efforts in point [88, 93] and image [11, 34, 60, 104, 110] self-supervised learning. This macro framework provides a solid foundation for pretext task design, allowing us to focus on addressing the geometric shortcut without additional concerns.

In essence, (point) self-supervised learning aims to *make things (points) that should be the same, the same (identical in representation)*. This forms the basic recipe of point SSL: generating two views of a given point cloud with random spatial (e.g., crop, rotate, distort) and photometric (e.g., jitter) augmentations, then matching and aligning the feature embeddings of points that are close in the original space.

However, the superior robustness of self-supervised representation is rooted in a core principle: *continuously increasing the difficulty of pretext tasks as long as the model continues to converge*. This principle encourages enhancing the basic recipe with local-global view alignment and mask-unmask view alignment. Specifically, it involves aligning

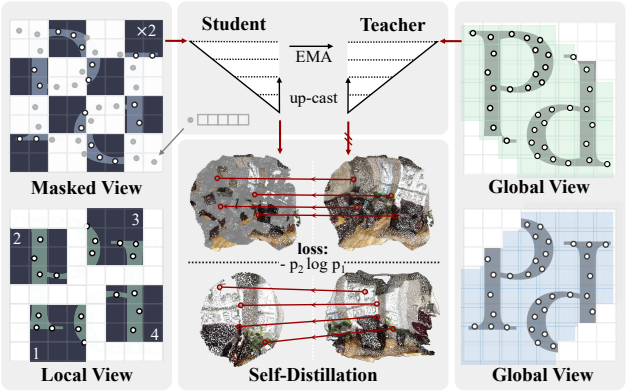

Figure 5. **Self-distillation framework of Sonata.** (1) Local views (*bottom left*) and global views (*right*) are generated with dedicated spatial and photometric augmentations, while masked views are created by randomly masking out grid-based patches from the global views (*top left*). (2) Embeddings from local and masked views are extracted by the student, with global views processed by the teacher (*top*). (3) Points from local and masked views are matched with corresponding points in the global views based on their original spatial distance, allowing for the distillation of embeddings from global views to local and masked views (*bottom*).

neighboring points from local views generated by small cropping ratios and masked views created with large masking ratios (see Fig. 5 left), with unmasked global views that contain relatively richer information (see Fig. 5 right). The difficulty of pretext tasks can be scaled by adjusting the cropping and masking ratios.

The more challenging the pretext tasks become, the higher the risk of model collapse. This instability calls for an *asymmetric encoding approach*, specifically exponential moving average (EMA) [34]: rather than encoding all views with a shared-weight model, the approach encodes challenging local and masked views with an actively learned student model, while using a stable teacher model, updated with a moving average of the student parameters, to encode global views (see Fig. 5 top). With the teacher preventing the student from being misled, the student is less likely to get lost in "mission impossible" and has a greater chance to discover treasure within "impossible" (*i.e.* extreme 5% crop ratio for local views and 70% mask ratio for masked views).

In terms of SSL criteria, we move away from prior approaches that rely on contrastive and generative learning [88]. Contrastive learning, limited by the number of point pairs in pairwise similarity computations, restricts scalability. Generative learning, meanwhile, partially anchors representations to predefined cues, limiting the model's capacity to capture more generalizable features. Following DINOv2, we adopt a self-distillation approach driven by Sinkhorn-Knopp centering [11], KoLeo regularization [72], and clustering assignments [11]. This adaptation initially exacerbates collapse into geometric shortcuts but holds greater potential for robust representation once this challenge is addressed.

4.2. Micro Design

We now discuss micro designs aimed at addressing geometric shortcuts. Since the problematic spatial information is inherently tied to point coordinates and directly used by operators, it is nearly impossible to mask out. This constraint defines the key strategies of our micro designs: *obscuring spatial information* and *emphasizing input features*.

Decoder removal. Previous approaches adhere to the original U-Net-style backbone for feature extraction; however, our observations in Sec. 3 motivate us to remove the complex hierarchical decoder and perform self-distillation directly using the encoder's output. This simple adjustment is key in our fight against geometric shortcuts, as it: (a) increases the feature channels participating in self-distillation (from 96 to 512), (b) streamlines the pipeline by involving fewer points in the pretext task after hierarchical pooling, and, most importantly, (c) introduces a natural way to obscure geometric cues: the positional information of points becomes naturally disturbed during hierarchical encoding and pooling. In fact, this removal proves to be even more beneficial than expected, boosting the linear probing result from 20.7% to 60.4%.

Feature up-casting. While the removal effectively prevents over-reliance on naive geometric cues, it also introduces certain limitations, particularly in leveraging multi-scale contexts. In the original U-Net structure, the decoder plays a key role in progressively aggregating features across scales to reconstruct semantic details. Without this process, self-distillation struggles to capture multi-scale spatial information and broader contextual relationships. To retain multi-scale features, we introduce a parameter-free feature up-casting process similar to hypercolumns in image segmentation [33]: progressively up-casting features back to the scale of the previous encoding stage, with the mapping relationships preserved through pooling layers and concatenation with features from the prior encoding stage. This approach provides richer, coarse-to-fine features from multi-scale encoding. While it does increase the risk of the model falling into geometric shortcuts, finding the right balance is key. Our ablation study shows that up-casting features twice achieve the best performance.

Masked points jitter. A slight random Gaussian jitter ($\sigma = 0.005$) is applied to point coordinates as part of data augmentation. However, specifically for points to be masked, we additionally apply a stronger Gaussian jitter ($\sigma = 0.01$) to further disrupt their spatial relationships. We pay special attention to these masked points because models are more likely to collapse to naive geometric cues, especially when point features are masked, making it difficult to derive solutions from neighboring unmasked points.

Progressive parameter scheduler. Geometric shortcuts are like traps on the convergence path of point cloud self-supervised learning. Similarly, we can set our own "trap" during training. For example, rather than starting with a challenging large mask size and mask ratio, we begin with a relatively small mask size (10 cm) and mask ratio (30%), gradually increasing them to 40 cm ($0.1 \Rightarrow 0.4$) and 70% ($0.3 \Rightarrow 0.7$) over the first 5% of the training process. This strategy encourages the model to rely more on input features to solve the pre-defined pretext tasks, preventing it from shifting its reliance to point coordinates as training difficulty increases. This approach aligns with curriculum learning [7], progressively challenging the model as it adapts. Similarly, in addition to the common learning rate scheduler, we also implement custom progressive schedulers for teacher temperature ($0.04 \Rightarrow 0.07$) and weight decay ($0.04 \Rightarrow 0.2$). We found that this design pushes these parameters to more extreme levels previously unexplored, further enhancing model performance.

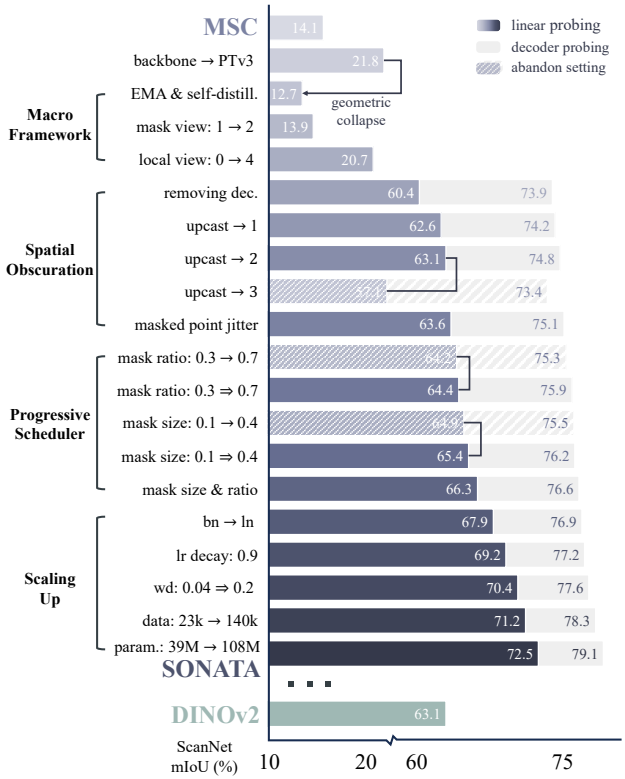

Figure 6. **The roadmap.** We evolve Mask Scene Contrast [88] into our Sonata by modernizing self-supervised learning with self-distillation, addressing the geometric shortcut, and scaling up training. Our designs are validated through progressive ablation with linear and decoder probing on ScanNet semantic segmentation [23]. Starting with 23k training data (a combination of ScanNet and Structured3D [109]) and a 39M PTv3 model [89], we ultimately scale up to 140k assets (Tab. 1) and a 108M PTv3 model.

4.3. Implementation and Evaluation Protocols

In this section, we introduce the implementation details and the evaluation protocols for our experiments.

Backbone. We build our Sonata with Point Transformer V3 (PTv3) [89] and refer to Pointcept [19] for details of imple-

Dataset	Source	Train	Val	Test	Total
ScanNet [23]	real	1,201	312	100	1,613
ScanNet++ [101]	real	712	178	126	1,016
S3DIS [1]	real	204	68	0	272
ArkitScenes [5]	real	4,498	549	0	5,047
HM3D [68]	real	8,881	1,119	0	10,000
Structured3D [109]	sim.	18,348	1,776	1,697	21,821
ASE [3]	sim.	90,000	10,000	0	100,000
Sonata (ours)	mixed	123,844	14,002	1,923	**139,769**

Table 1. **Data source collection.**

Method	Real	Sim	Total	Multipler
PC [93]	1,613	0	1,613	×1
MSC [88]	6,660	0	6,660	×4.1
PPT [90]	1,885	21,821	23,706	×14.7
Sonata (ours)	17,948	121,821	**139,768**	**×86.7**

Table 2. **Data scale comparison.**

mentation. Building on this, we made an additional adjustment to enhance scalability: replacing all Batch Normalization (BN) [41] layers with Layer Normalization (LN) [4]. Although this replacement results in some initial accuracy degradation, it enhances domain adaptation by eliminating the need for additional domain-specific adjustments when scaling up with multi-dataset joint training [90]. Along with the scaling up of data, we also scale up the encoder block depths from `[2,2,2,6,2]` to `[3,3,3,12,3]` and widths from `[32,64,128,256,512]` to `[48,96,192,384,512]`. This PTv3 model has 108M parameters.

Data. We extend the multi-dataset joint training approach introduced by PPT [90], further expanding the data scale by removing the constraint of human labeling through unsupervised learning. This results in a collection of 140k scene-level point clouds from both real-world and simulated environments (outlined in Tab. 1), making it 86.7× larger than the data scale of PointContrast [93] and 5.9× larger than the data collection of PPT, as detailed in Tab. 2.

Training. We train Sonata on the 140k data collection for 200 epochs, using the AdamW optimizer [54] with a batch size of 96, distributed across 32 GPUs. The learning rate linearly warms up over the first 10 epochs to a base value of 0.004, then decays following a cosine schedule [55]. Additionally, a layer-wise learning rate decay of 0.9 is applied to model parameters [111]. Weight decay is also controlled by a cosine schedule, progressively increasing from 0.04 to 0.2. For EMA, the student temperature is set to 0.1, while the teacher temperature gradually rises from 0.04 to 0.07 during the first 10 epochs [60]. The momentum starts at 0.994 and increases to 1 by the final iteration. For data augmentation and view generation, we follow the augmentation pipeline designed by MSC [88]. We generate 2 global views (sampling 40% to 100% of scene points) and 4 local views (sampling 5% to 40% of scene points) for training, with 2 masked views generated based on the global views.

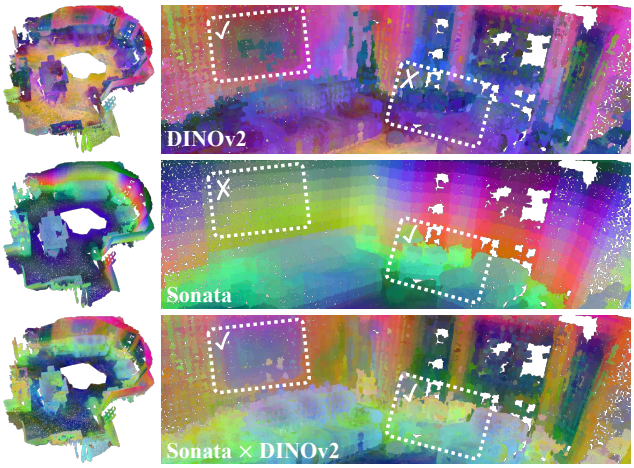

Figure 7. **Zero-shot comparison with DINOv2.** We compare the PCA visualizations of DINOv2, Sonata, and their combined feature representation. DINOv2 excels at capturing photometric details, while Sonata better distinguishes spatial information. The combined model demonstrates improved coherence and detail, showcasing the complementary strengths of both models.

Evaluation. We evaluate the quality of the learned representation using the following three protocols after initializing the encoder with Sonata:

- In *linear probing*, we keep the encoder frozen and upcast the features to their original scale. A single linear layer, comprising less than 0.2% of the total parameters, is then used to adapt these features to downstream tasks.
- In *decoder probing*, we take a step back and reintroduce a lightweight hierarchical decoder, which accounts for 13% of the total parameters. We then freeze the encoder, allowing only the decoder to actively learn.
- In *full fine-tuning*, we follow the traditional approach by unfreezing the entire PTv3 U-Net-style backbone, tuning the learned representation to downstream tasks.

We advocate linear probing as the primary evaluation criterion for point SSL, considering other methods as intermediate steps. We look forward to the day when a linear-probed self-supervised model outperforms a fully fine-tuned one.

5. Main Results

We validate the reliability of the Sonata representation using the evaluation protocols discussed in Sec. 4.3 and analyze the main properties based on these results.

Comparison with image self-supervised model. In Tab. 3, we compare the linear probing and decoder probing results on ScanNet and ScanNet200 semantic segmentation with the linear probing accuracy of representations transferred from the image self-supervised models DINOv2 [60] and DINOv2.5 [24]. Specifically, we aggregate unprojected pixel embeddings using ground truth camera poses and depth calculated by ray intersection with a reconstructed mesh, which provides more accuracy than sensor per-frame depth. Our results indicate that while the DINOs' represen-

2D × 3D	ScanNet Val [23]			ScanNet200 Val [23]		
Methods	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc
• DINOv2 (lin.) [60]	63.09	75.50	82.42	27.42	37.59	72.80
• DINOv2.5 (lin.) [24]	63.36	75.94	82.30	27.75	39.23	72.53
• Sonata (lin.)	72.52	83.11	89.74	29.25	41.61	81.15
+DINOv2 (lin.)	75.91	85.36	91.25	36.67	46.98	**82.85**
+DINOv2.5 (lin.)	**76.44**	**85.68**	**91.33**	**36.96**	**48.23**	82.77
• Sonata (dec.)	79.07	86.57	**92.68**	33.54	44.48	**84.07**
+DINOv2 (dec.)	79.12	**87.23**	92.47	37.73	**49.38**	83.31
+DINOv2.5 (dec.)	**79.19**	86.66	92.50	**38.27**	48.57	83.77

Table 3. **Numerical comparison with DINO series.**

Data Efficiency	Limited Scenes (Pct.)					Limited Annotation (Pts.)				
Methods	1%	5%	10%	20%	Full	20	50	100	200	Full
○ SparseUNet [17]	26.0	47.8	56.7	62.9	72.2	41.9	53.9	62.2	65.5	72.2
• CSC [38]	28.9	49.8	59.4	64.6	73.8	55.5	60.5	65.9	68.2	73.8
• MSC [88]	29.2	50.7	61.0	64.9	75.4	61.0	65.6	68.9	69.6	75.4
○ PTv2 [87]	24.8	48.1	59.8	66.3	75.4	58.4	66.1	70.3	71.2	75.4
○ PTv3 [89]	25.8	48.9	61.0	67.0	77.2	60.1	67.9	71.4	72.7	77.2
• PPT [90] (sup.)	31.1	52.6	63.3	68.2	78.2	62.4	69.1	74.3	75.5	78.2
• Sonata (lin.)	43.6	62.5	68.6	69.8	72.5	69.0	70.5	71.1	71.5	72.5
• Sonata (dec.)	44.5	64.1	69.8	72.5	79.1	69.8	73.1	75.0	76.3	79.1
• Sonata (full)	**45.3**	**65.7**	**72.4**	**72.8**	79.4	**70.5**	**73.6**	**76.0**	**77.0**	79.4

Table 4. **Data efficiency.**

Param. Effciency	Params		ScanNet Val [23]			ScanNet200 Val [71]			ScanNet++ Val [101]			S3DIS Area 5 [1]			S3DIS 6-fold [1]		
Methods	Learn.	Pct.	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc
○ SparseUNet [17]	39.2M	100%	72.3	80.2	90.0	25.0	32.9	80.4	28.8	38.4	80.1	66.3	72.5	89.8	72.4	80.9	89.9
• PC [93] (lin.)	<0.2M	<0.1%	5.6	9.7	50.0	0.5	0.9	40.3	1.8	3.1	46.4	11.4	18.6	52.3	11.7	19.0	51.2
• CSC [38] (lin.)	<0.2M	<0.1%	12.6	18.1	64.2	1.3	2.1	53.0	2.8	4.5	53.6	24.4	32.0	66.4	24.9	32.5	66.9
• MSC [88] (lin.)	<0.2M	<0.1%	14.1	20.3	62.9	1.5	2.5	53.6	4.5	6.6	61.3	27.9	35.5	71.1	29.9	37.9	71.3
○ PTv3 [89]	124.8M	100%	77.6	85.0	92.0	35.3	46.0	83.4	42.1	53.4	85.6	73.4	78.9	91.7	77.7	85.3	91.5
• MSC [88] (lin.)	<0.2M	<0.2%	21.8	32.2	65.5	3.3	5.5	57.5	8.1	11.9	64.7	32.1	42.4	70.9	34.6	46.0	71.3
• Sonata (lin.)	<0.2M	<0.2%	72.5	83.1	89.7	29.3	41.6	81.2	37.3	50.9	84.3	72.3	81.2	90.9	76.5	87.4	90.8
• Sonata (dec.)	16.3M	13%	**79.1**	**86.6**	**92.7**	**33.5**	**44.5**	**84.1**	**40.9**	**52.6**	**86.3**	**74.5**	**80.4**	**92.6**	**81.5**	**88.8**	**93.0**

Table 5. **Parameter efficiency.**

tation demonstrates impressive robustness, Sonata offers a more suitable representation for 3D tasks, achieving +9.2% on ScanNet and +1.5% on ScanNet200 for semantic segmentation. Furthermore, combining Sonata with the DINOs yields higher accuracy than any single data modality alone (+3.9% and +7.7%, respectively), underscoring the promising potential of cross-modal self-distillation. Additional zero-shot comparisons (see Fig. 7) through PCA visualizations further corroborate these numerical findings.

Data efficiency. In Tab. 4, we present the semantic segmentation performance of Sonata when probed or fine-tuned on the ScanNet dataset with limited scenes and annotations [38]. The results demonstrate the exceptional data efficiency of Sonata, with improvements of 19.5% in extreme data scarcity (1% of scenes) and 10.4% in limited annotation scenarios (20 points per scene) compared to training from scratch. Notably, even linear probing surpasses previous SOTA by a substantial margin (12.5% with 1% scenes), highlighting Sonata's reliability in low-data scenarios.

Parameter efficiency. In Tab. 5, we demonstrate parameter efficiency using both linear and decoder probing across various indoor semantic segmentation benchmarks, including ScanNet [23], ScanNet200 [71], ScanNet++ [101], S3DIS Area5[1], and S3DIS 6-fold cross-validation [65]. Semantic segmentation is emphasized as it provides a direct measure of point cloud representation quality. Results show that a single linear layer with a negligible number of parameters (<0.02% of total parameters) is sufficient for Sonata to achieve strong performance on these benchmarks (*e.g.*, 72.5% on ScanNet and 73.4% on S3DIS Area5). Furthermore, probing a decoder with only 13% of the model's parameters yields even higher accuracy (*e.g.*, 79.1% on ScanNet and 81.5% on S3DIS 6-fold cross-validation). However, while decoder probing achieves SOTA results on ScanNet (20 classes) and S3DIS (13 classes), performance on ScanNet200 (200 classes) and ScanNet++ (100 classes) remains limited. This shows a limitation of the learned representation in distinguishing a large number of classes.

Indoor semantic segmentation. In Tab. 6, we further enhance Sonata's semantic segmentation accuracy through full fine-tuning, consistently pushing SOTA results to new heights across the five widely recognized benchmarks, *e.g.*, achieving 79.4% on ScanNet and 82.3% on S3DIS 6-fold cross-validation. However, we view full fine-tuning as an intermediate step toward a future where linear probing surpasses it. At present, full fine-tuning remains essential to achieve the highest performance on these benchmarks and close the remaining gap of 7.1% and 5.8% respectively.

Indoor instance segmentation. In Tab. 7, We also validate the robustness of Sonata representation on indoor instance segmentation benchmarks, including ScanNet [23], ScanNet200 [71], ScanNet++ [101], and S3DIS [1]. Consistent with our findings in semantic segmentation, Sonata demonstrates strong parameter efficiency, achieving significant improvements with linear probing (10× mAP50 on ScanNet and 21× on ScanNet200) and decoder probing (12× mAP50 on ScanNet and 33× on ScanNet200). Full fine-tuning further boosts these results, achieving SOTA

Indoor Sem. Seg	Params		ScanNet Val [23]			ScanNet200 Val [71]			ScanNet++ Val [101]			S3DIS Area 5 [1]			S3DIS 6-fold [1]		
Methods	Learn.	Pct.	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc
○ SparseUNet [17]	39.2M	100%	72.3	80.2	90.0	25.0	32.9	80.4	28.8	38.4	80.1	66.3	72.5	89.8	72.4	80.9	89.9
• PC [93]	39.2M	100%	72.3	80.9	90.1	26.2	33.0	79.9	29.2	39.7	82.7	68.1	73.5	90.0	74.7	83.3	90.6
• CSC [38]	39.2M	100%	72.8	81.0	90.7	26.9	33.7	80.6	32.5	41.1	83.7	70.7	76.4	90.8	75.5	84.0	90.9
• MSC [88]	39.2M	100%	75.7	83.4	91.3	32.0	41.6	82.3	39.4	49.6	84.9	70.7	76.1	91.0	77.4	85.3	91.5
○ PTv3 [89]	124.8M	100%	77.6	85.0	92.0	35.3	46.0	83.4	42.1	53.4	85.6	73.4	78.9	91.7	77.7	85.3	91.5
• MSC [88]	124.8M	100%	78.2	85.3	92.2	33.4	43.7	83.4	42.4	53.6	85.9	69.9	74.9	91.2	77.4	84.7	91.7
• PPT [90] (sup.)	124.8M	100%	78.6	85.9	92.3	36.0	46.2	83.8	43.3	55.7	86.4	74.3	80.1	92.0	80.8	87.7	92.6
• Sonata	124.8M	100%	**79.4**	**86.1**	**92.5**	**36.8**	**46.5**	**84.4**	**43.7**	**55.8**	**86.6**	**76.0**	**81.6**	**93.0**	**82.3**	**89.9**	**93.3**

Table 6. **Indoor semantic segmentation.**

Indoor Ins. Seg	Params		ScanNet Val [23]			ScanNet200 Val [71]			ScanNet++ Val [101]			S3DIS Area 5 [1]		
Methods	Learn.	Pct.	mAP25	mAP50	mAP	mAP25	mAP50	mAP	mAP25	mAP50	mAP	mAP25	mAP50	mAP
○ PointGroup [45]	124.8M	100%	77.5	61.7	40.9	40.1	33.2	23.1	36.7	30.7	20.9	55.7	49.4	37.8
• MSC (lin.)	<0.2M	<0.2%	13.3	5.3	2.3	2.3	1.0	0.4	4.8	2.6	1.3	19.0	13.0	9.7
• Sonata (lin.)	<0.2M	<0.2%	72.6	53.9	30.7	30.9	21.3	10.9	31.6	22.4	12.2	45.8	36.6	26.1
• Sonata (dec.)	16.3M	13%	**76.8**	**62.8**	**40.8**	**40.8**	**33.3**	**22.8**	**38.1**	**29.1**	**18.8**	**63.7**	**57.1**	**45.1**
• MSC (full)	124.8M	100%	78.4	62.9	41.1	40.5	33.8	23.4	38.9	30.9	21.7	56.3	50.5	38.1
• PPT [90] (sup.)	124.8M	100%	78.9	63.5	42.1	40.8	34.1	24.0	39.3	32.8	21.9	57.5	51.2	39.7
• Sonata (full)	124.8M	100%	**79.2**	**63.9**	**42.4**	**42.1**	**35.6**	**25.4**	**40.3**	**33.6**	**22.3**	**63.8**	**57.4**	**45.5**

Table 7. **Indoor instance segmentation.**

Outdoor Sem. Seg.	nuScenes Val [9]			Waymo Val [74]			Sem.KITTI Val [6]		
Methods	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc
○ PTv3 [89]	80.4	87.2	94.7	71.3	80.5	94.7	69.1	76.1	92.6
• Sonata (lin.)	66.1	77.2	92.4	60.5	72.5	92.5	62.0	72.5	91.0
• Sonata (dec.)	**77.3**	**85.9**	**94.2**	**70.8**	**78.8**	**94.3**	**68.4**	**76.5**	**92.3**
• PPT [90] (sup.)	81.2	87.5	94.8	72.1	81.3	94.8	72.3	77.5	93.4
• Sonata (full)	**81.7**	**87.9**	**95.0**	**72.9**	**81.9**	**94.9**	**72.6**	**77.9**	**93.4**

Table 8. **Outdoor semantic segmentation.**

pre-training performance across benchmarks. This clearly demonstrates that, unlike previous SSL approaches, the Sonata representation encodes instance-level information.

Outdoor semantic segmentation. In Tab. 8, we adapt pre-training paradigm of Sonata to outdoor LiDAR scenarios through joint training on nuScenes [9], Waymo [74], and SemanticKITTI [6] and evaluate semantic segmentation performance using our evaluation protocols. With linear probing, Sonata sets a robust parameter efficiency baseline. Decoder probing achieves significant gains. In full fine-tuning, Sonata surpasses the supervised PPT [90], establishing new SOTA mIoU scores of 81.7, 72.9, and 72.6 across these benchmarks, underscoring the effectiveness of Sonata in outdoor perception tasks. Note that most of that performance can be recovered by more efficient decoder-only probing with a 95%, 97%, and 94% of full fine-tuning.

6. Conclusion and Discussion

In this work, we make progress towards self-supervised learning of a strong and reliable 3D point representation, that can zero-shot correspond semantically similar 3D points, down to the instance level, even under the presence of spatial and visual perturbations. We demonstrate that such a representation can serve as the foundation for 3D tasks in semantic and instance-level grouping. We find existing 3D self-supervised learning approaches lacking, and hypothesize that this is due to what we term the geometric shortcut, a problem unique to 3D, which causes representations to collapse to low-level spatial features. We demonstrate the deficiency of these collapsed representations through linear probing. Beginning with a point self-distillation framework, we tackle the geometric shortcut by attaching SSL losses at coarser spatial scales, disturbing the spatial information of masked points with no features, and progressively increasing task difficulty to prevent over-reliance on accessible geometric cues. This change enables effective scaling up, ultimately composing a **Sonata** from 140k point clouds. Sonata demonstrates semantically meaningful zero-shot visualization, as well as exceptional parameter and data efficiency. Full fine-tuning further advances SOTA across 3D indoor and outdoor perception tasks.

We summarize our limitations and future works and encourage readers to refer to the *supplementary* for details: 1. Sonata representations could be further semantically enriched through augmentation with 1M object-level point clouds [53]; 2. Pursuing unified training across indoor and outdoor scenarios as a feasible and promising direction; 3. Scaling up training with pixel-aligned [8] and SLAM-generated [27] point clouds from video data; 4. Leveraging cross-modal distillation to allow Sonata and DINOv2 to complement each other. We hope our insights will contribute to future research in 3D representation learning.

Acknowledgements

We extend our gratitude to Maxime Oquab and Piotr Bojanowski for their guidance on the DINOv2 training details, to Saining Xie for insightful discussions on the vision of 3D representation learning, and to Paul Mcvay for his thoughts on the JPEA framework with sparse point cloud data.

References

[1] 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 *CVPR*, 2016. 2, 6, 7, 8, 16

[2] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *CVPR*, 2023. 2

[3] Armen Avetisyan, Christopher Xie, Henry Howard-Jenkins, Tsun-Yi Yang, Samir Aroudj, Suvam Patra, Fuyang Zhang, Duncan Frost, Luke Holland, Campbell Orme, et al. Scenescript: Reconstructing scenes with an autoregressive structured language model. In *ECCV*, 2024. 2, 6, 16

[4] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *Stat*, 2016. 6

[5] 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 *NeurIPSW*, 2021. 2, 6

[6] 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 *ICCV*, 2019. 8

[7] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, 2009. 5

[8] Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. *arXiv preprint arXiv:2410.02073*, 2024. 8, 13

[9] 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 *CVPR*, 2020. 2, 8

[10] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018. 2

[11] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020. 2, 4

[12] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *CVPR*, 2021. 2, 4

[13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 2

[14] Xi Chen, Lianghua Huang, Yu Liu, Yujun Shen, Deli Zhao, and Hengshuang Zhao. Anydoor: Zero-shot object-level image customization. In *CVPR*, 2024. 2

[15] Yukang Chen, Jianhui Liu, Xiangyu Zhang, Xiaojuan Qi, and Jiaya Jia. Largekernel3d: Scaling up kernels in 3d sparse cnns. In *CVPR*, 2023. 16

[16] Hung-Yueh Chiang, Yen-Liang Lin, Yueh-Cheng Liu, and Winston H Hsu. A unified point-based framework for 3d segmentation. In *3DV*, 2019. 16

[17] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *CVPR*, 2019. 3, 7, 8, 16, 17

[18] Christopher Choy, Jaesik Park, and Vladlen Koltun. Fully convolutional geometric features. In *ICCV*, 2019. 2

[19] Pointcept Contributors. Pointcept: A codebase for point cloud perception research. https://github.com/Pointcept/Pointcept, 2023. 5

[20] Spconv Contributors. Spconv: Spatially sparse convolution library. https://github.com/traveller59/spconv, 2022. 3

[21] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 303–312, 1996. 16

[22] Angela Dai and Matthias Nießner. 3dmv: Joint 3d-multi-view prediction for 3d semantic scene segmentation. In *ECCV*, 2018. 16

[23] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 2, 5, 6, 7, 8, 16

[24] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. *arXiv:2309.16588*, 2023. 6, 7

[25] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, 2015. 2

[26] Alexey Dosovitskiy, Philipp Fischer, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *TPAMI*, 2015. 2

[27] Jakob Engel, Kiran Somasundaram, Michael Goesele, Albert Sun, Alexander Gamino, Andrew Turner, Arjang Talattof, Arnie Yuan, Bilal Souti, Brighid Meredith, et al. Project aria: A new tool for egocentric multi-modal ai research. *arXiv:2308.13561*, 2023. 2, 8, 13

[28] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 2004. 3

[29] Benjamin Graham and Laurens van der Maaten. Submanifold sparse convolutional networks. *arXiv:1706.01307*, 2017. 3

[30] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *CVPR*, 2018. 16

[31] 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 *NeruIPS*, 2020. 2

[32] Jiayuan Gu, Fanbo Xiang, Xuanlin Li, Zhan Ling, Xiqiang Liu, Tongzhou Mu, Yihe Tang, Stone Tao, Xinyue Wei, Yunchao Yao, Xiaodi Yuan, Pengwei Xie, Zhiao Huang, Rui Chen, and Hao Su. Maniskill2: A unified benchmark for generalizable manipulation skills. In *ICLR*, 2023. 2

[33] Bharath Hariharan, Pablo Arbelaez, Ross Girshick, and Jitendra Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, 2015. 5

[34] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 2, 3, 4

[35] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 2

[36] R. Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *ICLR*, 2018. 2

[37] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023. 16

[38] Ji Hou, Benjamin Graham, Matthias Nießner, and Saining Xie. Exploring data-efficient 3d scene understanding with contrastive scene contexts. In *CVPR*, 2021. 2, 3, 7, 8, 16, 17

[39] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *CVPR*, 2020. 16

[40] Zeyu Hu, Mingmin Zhen, Xuyang Bai, Hongbo Fu, and Chiew-lan Tai. Jsenet: Joint semantic segmentation and edge detection network for 3d point clouds. In *ECCV*, 2020. 16

[41] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 6

[42] Ayush Jain, Pushkal Katara, Nikolaos Gkanatsios, Adam W. Harley, Gabriel Sarch, Kriti Aggarwal, Vishrav Chaudhary, and Katerina Fragkiadaki. Odin: A single model for 2d and 3d perception. In *CVPR*, 2024. 16

[43] Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J. Davison. Rlbench: The robot learning benchmark & learning environment. *RAL*, 2020. 2

[44] Li Jiang, Hengshuang Zhao, Shu Liu, Xiaoyong Shen, Chi-Wing Fu, and Jiaya Jia. Hierarchical point-edge interaction network for point cloud semantic segmentation. In *ICCV*, 2019. 17

[45] Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Pointgroup: Dual-set point grouping for 3d instance segmentation. *CVPR*, 2020. 8

[46] Maxim Kolodiazhnyi, Anna Vorontsova, Anton Konushin, and Danila Rukhovich. Oneformer3d: One transformer for unified point cloud segmentation. In *CVPR*, 2024. 16

[47] Xin Lai, Jianhui Liu, Li Jiang, Liwei Wang, Hengshuang Zhao, Shu Liu, Xiaojuan Qi, and Jiaya Jia. Stratified transformer for 3d point cloud segmentation. In *CVPR*, 2022. 16, 17

[48] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *CVPR*, 2018. 17

[49] Huan Lei, Naveed Akhtar, and Ajmal Mian. Seggcn: Efficient 3d point cloud segmentation with fuzzy spherical kernel. In *CVPR*, 2020. 16, 17

[50] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. Pointcnn: Convolution on x-transformed points. *NeurIPS*, 2018. 16, 17

[51] Dingkang Liang, Xin Zhou, Wei Xu, Xingkui Zhu, Zhikang Zou, Xiaoqing Ye, Xiao Tan, and Xiang Bai. Pointmamba: A simple state space model for point cloud analysis. In *Advances in Neural Information Processing Systems*, 2024. 3

[52] Haojia Lin, Xiawu Zheng, Lijiang Li, Fei Chao, Shanshan Wang, Yan Wang, Yonghong Tian, and Rongrong Ji. Meta architecture for point cloud analysis. In *CVPR*, pages 17682–17691, 2023. 16, 17

[53] Minghua Liu, Ruoxi Shi, Kaiming Kuang, Yinhao Zhu, Xuanlin Li, Shizhong Han, Hong Cai, Fatih Porikli, and Hao Su. Openshape: Scaling up 3d shape representation towards open-world understanding. In *NeurIPS*, 2023. 8, 13

[54] I Loshchilov. Decoupled weight decay regularization. In *ICLR*, 2019. 6

[55] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *ICLR*, 2017. 6

[56] Xu Ma, Can Qin, Haoxuan You, Haoxi Ran, and Yun Fu. Rethinking network design and local geometry in point cloud: A simple residual mlp framework. *ICLR*, 2022. 3

[57] Gaku Narita, Takashi Seno, Tomoya Ishikawa, and Yohsuke Kaji. Panopticfusion: Online volumetric semantic mapping at the level of stuff and things. In *IROS*, 2019. 16

[58] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE international symposium on mixed and augmented reality*, pages 127–136. Ieee, 2011. 16

[59] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016. 2

[60] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve

Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *TMLR*, 2024. 2, 3, 4, 6, 7, 13

[61] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. In *ECCV*, 2022. 2

[62] Chunghyun Park, Yoonwoo Jeong, Minsu Cho, and Jaesik Park. Fast point transformer. In *CVPR*, 2022. 16

[63] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *NeurIPS*, 2019. 16

[64] Bohao Peng, Xiaoyang Wu, Li Jiang, Yukang Chen, Hengshuang Zhao, Zhuotao Tian, and Jiaya Jia. Oa-cnns: Omni-adaptive sparse cnns for 3d semantic segmentation. In *CVPR*, 2024. 3

[65] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017. 7, 17

[66] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, 2017. 3, 16, 17

[67] Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Hammoud, Mohamed Elhoseiny, and Bernard Ghanem. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. In *NeurIPS*, 2022. 3, 16, 17

[68] 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 *NeurIPS*, 2021. 2, 6, 14, 15

[69] Damien Robert, Hugo Raguet, and Loic Landrieu. Efficient 3d semantic segmentation with superpoint transformer. In *ICCV*, 2023. 17

[70] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 3

[71] David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3d semantic segmentation in the wild. In *ECCV*, 2022. 7, 8

[72] Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, and Hervé Jégou. Spreading vectors for similarity search. In *ICLR*, 2019. 4

[73] Julian Straub, Daniel DeTone, Tianwei Shen, Nan Yang, Chris Sweeney, and Richard Newcombe. Efm3d: A benchmark for measuring progress towards 3d egocentric foundation models. *arXiv:2406.10224*, 2024. 2, 14, 15, 16

[74] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *CVPR*, 2020. 2, 8

[75] Maxim Tatarchenko, Jaesik Park, Vladlen Koltun, and Qian-Yi Zhou. Tangent convolutions for dense prediction in 3d. In *CVPR*, 2018. 17

[76] Lyne Tchapmi, Christopher Choy, Iro Armeni, JunYoung Gwak, and Silvio Savarese. Segcloud: Semantic segmentation of 3d point clouds. In *3DV*, 2017. 17

[77] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *ICCV*, 2019. 3, 16, 17

[78] Hugues Thomas, Yao-Hung Hubert Tsai, Timothy D Barfoot, and Jian Zhang. Kpconvx: Modernizing kernel point convolution with kernel attention. In *CVPR*, 2024. 16

[79] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 16

[80] 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 *CVPR*, 2024. 2

[81] Lei Wang, Yuchun Huang, Yaolin Hou, Shenman Zhang, and Jie Shan. Graph attention convolution for point cloud semantic segmentation. In *CVPR*, 2019. 17

[82] Peng-Shuai Wang. Octformer: Octree-based transformers for 3D point clouds. *SIGGRAPH*, 2023. 16

[83] Shenlong Wang, Simon Suo, Wei-Chiu Ma, Andrei Pokrovsky, and Raquel Urtasun. Deep parametric continuous convolutional neural networks. In *CVPR*, 2018. 17

[84] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024. 2, 13

[85] Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. In *CVPR*, 2019. 16

[86] Wenxuan Wu, Li Fuxin, and Qi Shan. Pointconvformer: Revenge of the point-based convolution. In *CVPR*, 2023. 16

[87] Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. Point transformer v2: Grouped vector attention and partition-based pooling. In *NeurIPS*, 2022. 7, 16, 17

[88] Xiaoyang Wu, Xin Wen, Xihui Liu, and Hengshuang Zhao. Masked scene contrast: A scalable framework for unsupervised 3d representation learning. In *CVPR*, 2023. 2, 3, 4, 5, 6, 7, 8, 14, 16, 17

[89] Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler, faster, stronger. In *CVPR*, 2024. 3, 5, 7, 8, 16, 17

[90] Xiaoyang Wu, Zhuotao Tian, Xin Wen, Bohao Peng, Xihui Liu, Kaicheng Yu, and Hengshuang Zhao. Towards large-scale 3d representation learning with multi-dataset point prompt training. In *CVPR*, 2024. 3, 6, 7, 8, 16, 17

[91] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018. 2

[92] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *NeurIPS*, 2021. 3

[93] Saining Xie, Jiatao Gu, Demi Guo, Charles R Qi, Leonidas Guibas, and Or Litany. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In *ECCV*, 2020. 2, 3, 4, 6, 7, 8, 16, 17

[94] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. In *CVPR*, 2022. 2

[95] Mutian Xu, Runyu Ding, Hengshuang Zhao, and Xiaojuan Qi. Paconv: Position adaptive convolution with dynamic kernel assembling on point clouds. In *CVPR*, 2021. 17

[96] Xu Yan, Chaoda Zheng, Zhen Li, Sheng Wang, and Shuguang Cui. Pointasnl: Robust point clouds processing using nonlocal neural networks with adaptive sampling. In *CVPR*, 2020. 16

[97] Jiancheng Yang, Qiang Zhang, Bingbing Ni, Linguo Li, Jinxian Liu, Mengdie Zhou, and Qi Tian. Modeling point clouds with self-attention and gumbel subset sampling. In *CVPR*, 2019. 17

[98] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *CVPR*, 2024. 2

[99] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. In *NeurIPS*, 2024. 2

[100] Yu-Qi Yang, Yu-Xiao Guo, Jian-Yu Xiong, Yang Liu, Hao Pan, Peng-Shuai Wang, Xin Tong, and Baining Guo. Swin3d: A pretrained transformer backbone for 3d indoor scene understanding. *arXiv:2304.06906*, 2023. 16, 17

[101] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *ICCV*, 2023. 2, 6, 7, 8

[102] 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 *CVPR*, 2022. 2

[103] Feihu Zhang, Jin Fang, Benjamin Wah, and Philip Torr. Deep fusionnet for point cloud semantic segmentation. In *ECCV*, 2020. 16

[104] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M. Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv:2203.03605*, 2022. 2, 3, 4

[105] Michael Zhang, Aditi Raghunathan Wang, Shiori Sagawa, Tatsunori B Hashimoto, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *ICLR*, 2022. 2

[106] Renrui Zhang, Ziyu Guo, Peng Gao, Rongyao Fang, Bin Zhao, Dong Wang, Yu Qiao, and Hongsheng Li. Point-m2ae: Multi-scale masked autoencoders for hierarchical point cloud pre-training. In *NeurIPS*, 2022. 2

[107] Hengshuang Zhao, Li Jiang, Chi-Wing Fu, and Jiaya Jia. Pointweb: Enhancing local neighborhood features for point cloud processing. In *CVPR*, 2019. 17

[108] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. Point transformer. In *ICCV*, 2021. 16, 17

[109] Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. Structured3d: A large photo-realistic dataset for structured 3d modeling. In *ECCV*, 2020. 2, 5, 6

[110] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. In *ICLR*, 2022. 2, 4

[111] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. Image BERT pre-training with online tokenizer. In *ICLR*, 2022. 6

[112] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Simple multi-dataset detection. In *CVPR*, 2022. 3