Appendix

For a thorough understanding of our Sonata, we have compiled a detailed Appendix. The table of contents below offers an overview and guide to specific sections of interest.

Contents

A. Discussion

We discuss *limitations and future works* as follows:

- *Enhancing semantic meaning*. We believe there is significant potential to enhance the semantic richness of Sonata's representations. Currently, our training does not yet leverage the vast resource of 1M object-level assets [53], which could provide valuable augmentation for our scene-level point cloud dataset. Integrating these object-level point clouds could deepen the model's semantic understanding by introducing finer object-specific details, creating a more robust foundation for scene-level and cross-instance semantics.
- *Unifying training scenarios*. Unifying training across indoor and outdoor scenarios is a promising direction for future work. Currently, Sonata separates pre-training for each setting to focus on a reliable SSL framework without introducing the additional challenge of a domain gap. However, unification is feasible. The main challenges lie in point density and input features: point density can be aligned by scaling, while enhancing outdoor LiDAR data with color from lifted images and pseudo normal vectors based on LiDAR viewing direction helps bridge feature gaps. Additionally, applying randomized noise and masking on these features could further enhance generalization.
- *Scaling with video data*. Natural 3D point cloud datasets have inherent scale limitations compared to video data. To address this, we aim to leverage video datasets in two ways: 1. using metric [8] or stereo [84] depth estimation to lift videos of static scenes into pixel-aligned point clouds, and 2. generating sparse point clouds from dynamic egocentric videos using SLAM algorithms [27]. This approach opens new possibilities for training on large-scale, real-world diverse scenes.

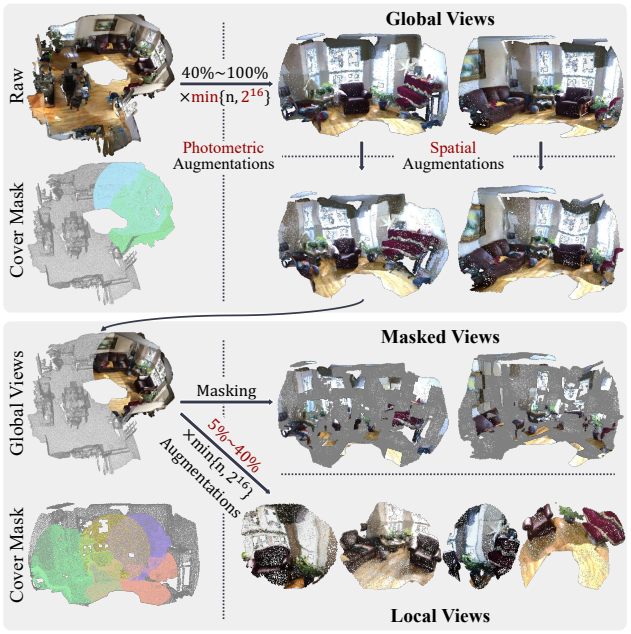


Figure 8. **View generation.** *Top*: we generate global crops using random crops with a crop ratio ranging from 40% to 100% of the minimal number of the raw point cloud size and 2^{16}, combined with random photometric and spatial augmentations. Photometric augmentation is shared among all global views, while spatial augmentation is applied independently to each global view to balance the challenges posed by spatial and photometric features. The first global view is designated as the principal view, and the center of the subsequent global view is restricted to fall within the principal view. *Bottom:* Local views are generated with a similar pipeline as global views but with a crop ratio restricted to 5% to 40%. All augmentations are applied independently to each local view. Additionally, masked views are obtained by applying random patch masks to the global views.

- *Cross-modal distillation*. Our evidence shows that self-supervised models from different modalities, like Sonata for point clouds and DINOv2 [60] for images, capture complementary representations, and combining them leads to stronger representation. This suggests promising potential for cross-modal self-distillation to enhance both 3D and image representations. A straightforward approach would be to lift DINOv2 features into 3D and integrate them within Sonata's pre-training paradigm. Additionally, developing a unified SSL framework with simultaneous self- and cross-modal distillation across point clouds and images could further enrich multi-modal representation learning.

We hope our insights with Sonata inspire innovations in reliable point self-supervised learning and pave the way for future research in 3D representations and its applications.

Algorithm 1 point self-distillation pseudocode.

```
'''
To simplify, we present the pseudocode using a single
local(masked)-global pair.
'''

# gs, gt: student and teacher networks
# cs, ct: student and teacher online clustering head
# tps, tpt, student and teacher temperatures
# m: network momentum rates
# k: upcast level

# initialize student and teacher network and head
gt.params, ct.params = gs.params, cs.params
gt.requires_grad = False
ct.requires_grad = False

for p in loader:  # load a batch of point cloud
    # ps: local(mask) view, pt: global view
    ps, pt = view_generator(p)
    if ps is MaskedView:
        # apply gaussian noise to masked points
        ps.coord[p1.mask] += gaussian(s)
    # encode network feature
    fs, ft = gs(ps), gt(pt)
    # up-cast network feature
    fs, ft = upcast(fs, k), upcast(ft, k)
    # compute similarity with online cluster (SwAV)
    ss, st = cs(s1), ct(s2)
    # center with sinkhorn-knopp
    st = centering(st)

    # match neighbor point pairs with the original
    # coordinate before augmentation, return index
    is, it = match(ps.origin_coord, pt.origin_coord)
    loss = H(ss[is], st[it])
    loss.backward()

    # update student and teacher network and head
    update(gs.params)
    update(cs.params)
    gt.params =  m*gt.params + (1-m)*gs.params
    ct.params =  m*ct.params + (1-m)*cs.params

def H(t, s):
    s = softmax(s / tps, dim=-1)
    # center with sinkhorn-knopp and sharpen
    t = softmax(center(t) / tpt, dim=-1)
    return - (t * log(s)).sum(dim=1).mean()
```

B. Additional Implementation

B.1. View Generation

In Fig. 8, we illustrate the view generation pipeline of Sonata. Specifically, global views are generated with a random crop ratio between 40% and 100%, while local views use a ratio between 5% and 40%. The crop ratio is applied to the smaller of the raw point cloud size or 2^{16} points. The first global view is designated as the principal view, and subsequent global and local view centers are restricted to lie within this principal view. Random photometric and spatial augmentations [88] are applied to all views. For global views, photometric augmentations are shared after being randomized, whereas spatial augmentations are applied independently to each view. Masked views are generated by applying random patch masks to the global views.

B.2. Point Self-distillation

In Algo.1, we provide a simplified pseudocode for point self-distillation using a single local (masked)-global pair of random views. In the actual implementation, we use a total of 4 local views, 2 masked views, and 2 global views.

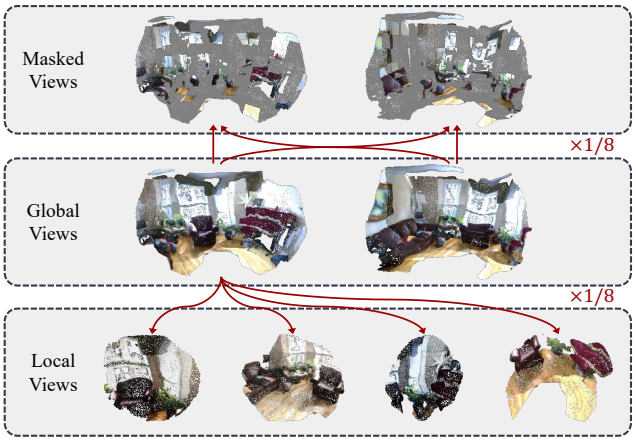


Figure 9. **Point self-distillation loss items.** The pair-wise point self-distillation between masked views and global views, and between local views and the principal global view. We evenly weight the loss terms for the 8 point self-distillation pairs.

Sem. Seg.	Params		AEO [73]		
Methods	Learn.	Pct.	mIoU	mAcc	allAcc
○ PTv3	124.8M	100%	34.91	47.92	63.79
• Sonata (lin.)	<0.2M	<0.2%	32.03	47.45	53.25
• Sonata (full)	124.8M	100%	**55.90**	**63.49**	**84.50**

Table 9. **Out-of-distribution perception capability.** We evaluate this capability on the AEO dataset [73] with 22 sparse SLAM point clouds, supervised by semantic labels from object bounding boxes.

For the local views, point self-distillation is conducted between each local view and the principal global view. For the masked views, pair-wise point self-distillation is performed with each global view. The loss terms for all point self-distillation pairs are evenly weighted (visualized in Fig. 9).

C. Additional Zero-shot Visualization

In Fig. 10, we provide additional zero-shot visualizations, including PCA and dense matching, using the Habitat-Matterport 3D Dataset (HM3D) [68]. Specifically, we encode a house-scale point cloud comprising 2 floors and 12 rooms, visualizing the learned representations with PCA-mapped colors to highlight the semantic structure of the space. Furthermore, we select five representative points from various objects, including a sofa arm, chair, table, pillow, and side table, and visualize dense matching by computing the similarity of each selected point with the rest of the house-scale point cloud. This process highlights the semantic coherence and clustering of features across objects and spaces. The visualization demonstrates that Sonata consistently provides semantically rich and informative representations across diverse indoor environments. These representations effectively capture distinct object patterns, exhibit a high degree of semantic granularity, and enable meaningful queries without any supervision, reinforcing the robustness and utility of the Sonata features.

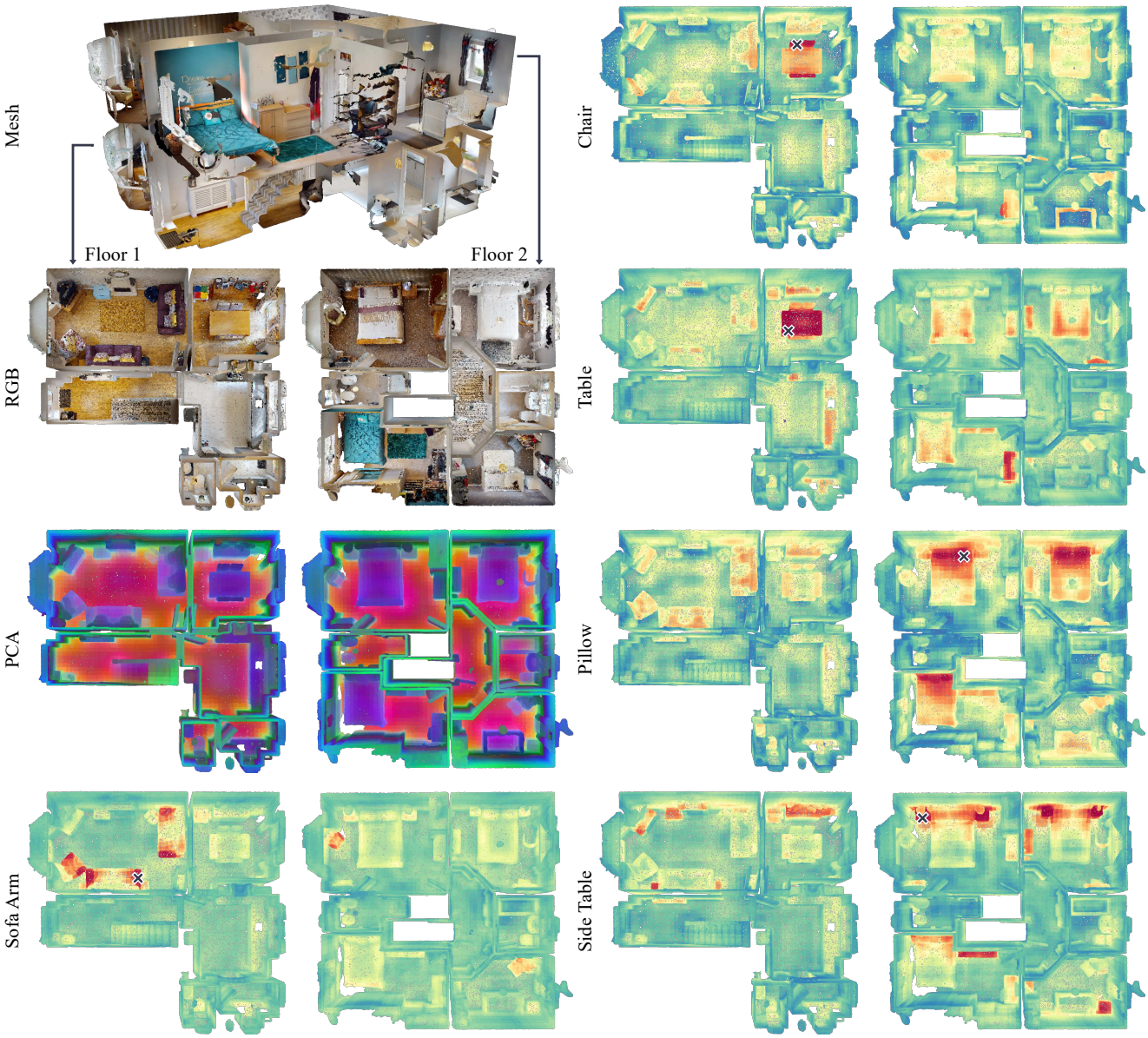


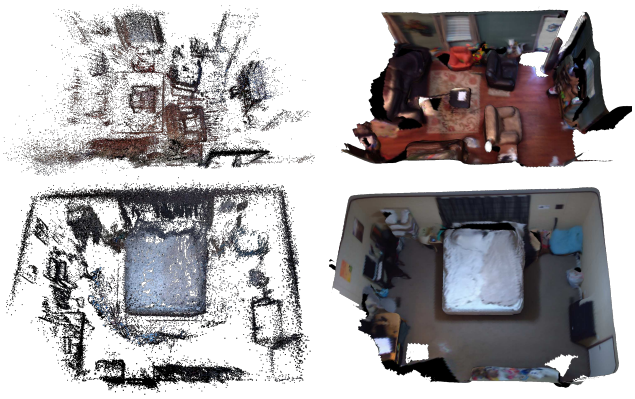
Figure 10. **Zero-shot visualization.** We provide PCA-mapped colors and dense matching (with five representative points marked with $\times$) on a house-scale point cloud from HM3D [68], comprising 2 floors and 12 rooms (*left:* floor 1, *right:* floor 2). The visualization demonstrates that Sonata consistently delivers semantically rich and informative representations across diverse indoor environments.

D. Additional Properties

D.1. Out-of-distribution (OOD) Perception.

In Tab. 9, we evaluate the out-of-distribution perception capability of Sonata using the Aria Everyday Objects (AEO) dataset [73] for semantic segmentation. This dataset consists of 25 sparse SLAM-generated point clouds, each annotated with 17 object categories. Among these, three samples (IDs: 0, 5, 24) are reserved for validation, while the remaining 22 are used for training. This experimental setup presents significant challenges, including unseen data patterns (as shown by the sparse pattern of SLAM-generated point clouds in Fig. 11, left column), limited training data, and imprecise annotations derived from bounding box labels. We assess Sonata by performing both probing and fine-tuning on this semantic segmentation task, supervised by the transferred semantic labels.

The results show that the linear probing of Sonata achieves a mIoU of 32.0%, which still has a gap of 2.9% compared to the 34.9% mIoU achieved by training from scratch. This indicates a current limitation of Sonata: insufficient diversity in training data patterns. Currently, we only include dense indoor point clouds, focusing on building a reliable point SSL framework without introducing addi-



Input Point Cloud Output SDF Mesh

Figure 11. **Surface reconstruction.** Scene surface is reconstructed with SDF regression from frozen Sonata features, demonstrating strong geometric priors and cross-domain generalization.

tional domain gap challenges. However, fine-tuning Sonata demonstrates its robustness, achieving a remarkable 21.0% improvement over training from scratch. This substantial leap underscores the strength and adaptability of Sonata representations, particularly in challenging OOD perception tasks. These findings further reinforce Sonata's potential as a foundation for tackling tasks with limited or noisy training data in diverse and complex environments.

D.2. Surface Reconstruction.

Previous experiments have already demonstrated the rich semantic information encoded in Sonata representations. To further investigate whether Sonata also captures dense geometric priors, we conducted a surface regression experiment using frozen Sonata features. A Truncated Signed Distance Function (TSDF) [21] volume is defined within a $4m \times 4m \times 4m$ local coordinate system with a resolution of $96 \times 96 \times 96$ (≈4cm per voxel). The volume is patchified into $8 \times 8 \times 8$ patches, with each patch projected to a feature dimension of 512. This results in $512 \times 12 \times 12 \times 12$ volume tokens. We applied learned positional encodings to the patches, and tri-linear interpolation ensured consistency between the positional encodings and Sonata features. To decode the volume tokens into a dense TSDF volume, we simply use the standard TransformerDecoder [79] implemented in PyTorch [63], with the Sonata features being the "memory" and the voxel tokens being the decoding "target". One can also see the Sonata features as the context while the voxel tokens are the queries. After decoding, the outputs were reshaped to reconstruct the dense TSDF volume. This approach is inspired by the Large Reconstruction Models (LRM)s [37]. We employed the EVL training and TSDF fusion [58, 73] pipeline, training the network on the synthetic ASE dataset [3]. The cross-domain generalization was tested on the SLAM-generated point cloud inputs of the AEO dataset, as illustrated in the Fig. 11. The results qualitatively demonstrate that dense scene geometry can be reconstructed solely from frozen Sonata features, showcasing learned geometric priors within Sonata representations.

Methods	Year	Val	Test
○ PointNet++ [66]	2017	53.5	55.7
○ 3DMV [22]	2018	-	48.4
○ PointCNN [50]	2018	-	45.8
○ SparseConvNet [30]	2018	69.3	72.5
○ PanopticFusion [57]	2019	-	52.9
○ PointConv [85]	2019	61.0	66.6
○ JointPointBased [16]	2019	69.2	63.4
○ KPConv [77]	2019	69.2	68.6
○ PointASNL [96]	2020	63.5	66.6
○ SegGCN [49]	2020	-	58.9
○ RandLA-Net [39]	2020	-	64.5
○ JSENet [40]	2020	-	69.9
○ FusionNet [103]	2020	-	68.8
○ FastPointTransformer [62]	2022	72.4	-
○ SratifiedTranformer [47]	2022	74.3	73.7
○ PointNeXt [67]	2022	71.5	71.2
○ LargeKernel3D [15]	2023	73.5	73.9
○ PointMetaBase [52]	2023	72.8	71.4
○ PointConvFormer [86]	2023	74.5	74.9
○ OctFormer [82]	2023	75.7	76.6
○ Swin3D [100]	2023	77.5	77.9
• Supervised [100]	2023	76.7	77.9
○ KPConvX [78]	2024	76.3	-
○ OneFormer3D [46]	2024	76.6	-
○ ODIN [42]	2024	77.8	74.4
○ SparseUNet [17]	2019	72.2	73.6
• PC [93]	2020	74.1	-
• CSC [38]	2021	73.8	-
• MSC [88]	2023	75.5	-
• PPT (sup.) [90]	2023	76.4	76.6
○ PTv1 [108]	2021	70.6	-
○ PTv2 [87]	2022	75.4	74.2
○ PTv3 [89]	2023	77.5	77.9
• MSC [88]	2023	78.2	-
• PPT (sup.) [90]	2023	78.6	**79.4**
• Sonata (linear probing)	2024	72.5	-
• Sonata (decoder probing)	2024	79.1	-
• Sonata (full fine-tuning)	2024	**79.4**	-

Table 10. **ScanNet V2 semantic segmentation.**

E. Additional Comparision

In this section, we expand the combined results table for indoor semantic segmentation from the main paper, providing a more detailed comparison of results on two key benchmarks: ScanNet [23] (see Tab. 10) and S3DIS [1] (see Tab. 11). Specifically, the ScanNet v2 dataset contains 1,513 room scans reconstructed from RGB-D frames, with 1,201 scenes allocated for training and 312 for validation. The input point clouds are derived from the vertices of reconstructed meshes, where each point is labeled with one of 20 semantic categories (e.g., wall, floor, table). The S3DIS dataset includes 271 rooms distributed across six areas from three buildings, specifically designed

Methods	Year	Area5	6-fold
○ PointNet [65]	2017	41.1	47.6
○ SegCloud [76]	2017	48.9	-
○ TanConv [75]	2018	52.6	-
○ PointCNN [50]	2018	57.3	65.4
○ ParamConv [83]	2018	58.3	-
○ PointWeb [107]	2019	60.3	66.7
○ HPEIN [44]	2019	61.9	-
○ KPConv [77]	2019	67.1	70.6
○ GACNet [81]	2019	62.9	-
○ PAT [97]	2019	60.1	-
○ SPGraph [48]	2018	58.0	62.1
○ SegGCN [49]	2020	63.6	-
○ PAConv [95]	2021	66.6	-
○ StratifiedTransformer [47]	2022	72.0	-
○ PointNeXt [67]	2022	70.5	74.9
○ SuperpointTransformer [69]	2023	68.9	76.0
○ PointMetaBase [52]	2023	72.0	77.0
○ Swin3D [100]	2023	72.5	76.9
• Supervised [100]	2023	74.5	79.8
○ MinkUNet [17]	2019	65.4	65.4
• PC [93]	2020	70.3	-
• CSC [38]	2021	72.2	-
• MSC [88]	2023	70.1	-
• PPT (sup.) [90]	2023	74.7	78.1
○ PTv1 [108]	2021	70.4	65.4
○ PTv2 [87]	2022	71.6	73.5
○ PTv3 [89])	2023	73.4	77.7
• PPT [90]	2023	74.7	80.8
• Sonata (linear probing)	2024	72.3	76.5
• Sonata (decoder probing)	2024	74.5	81.5
• Sonata (full fine-tuning)	2024	**76.0**	**82.3**

Table 11. **S3DIS semantic segmentation.**

for semantic scene parsing. Following established practices [66, 76, 108], area 5 is reserved for testing, and 6-fold cross-validation is performed across the remaining areas. Unlike ScanNet v2, S3DIS features densely sampled points on mesh surfaces, with annotations across 13 categories. For both datasets, we adopt the mean class-wise intersection over union (mIoU) as the primary metric to evaluate performance on indoor semantic segmentation tasks, adhering to standard conventions [66]. These expanded tables provide a detailed breakdown of performance metrics alongside the publication years of previous works, allowing readers to trace the evolution of advancements in 3D representation learning. Entries labeled as ○ correspond to models trained from scratch, while • denotes results achieved using pre-trained models.