Towards Geometry-Grounded World Understanding

LIYAO TANG

Supervisor: Dacheng Tao
Associate Supervisor: Zhe Chen

A thesis submitted in fulfilment of
the requirements for the degree of
Doctor of Philosophy

School of Computer Science
Faculty of Engineering
The University of Sydney
Australia

1 June 2026

Statement of Originality

This is to certify that to the best of my knowledge, the content of this thesis is my own work. This thesis has not been submitted for any degree or other purposes. I certify that the intellectual content of this thesis is the product of my own work and that all the assistance received in preparing this thesis and sources have been acknowledged.

Liyao Tang
School of Computer Science
Faculty of Engineering
The University of Sydney
June 1, 2026

Authorship Attribution Statement

This thesis was conducted at the University of Sydney, under the supervision of Prof. Dacheng Tao and Dr. Zhe Chen, between 2022 and 2025. The main results presented in this dissertation were first introduced in the following publications:

(1) **Liyao Tang**, Yibing Zhan, Zhe Chen, Baosheng Yu, and Dacheng Tao. Contrastive Boundary Learning for Point Cloud Segmentation. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8479-8489. DOI: 10.1109/CVPR52688.2022.00830.* Presented in Chapter 3. I designed the research, implemented the systems, conducted the experiments, and wrote the draft of the paper.

(2) **Liyao Tang**, Zhe Chen, Shanshan Zhao, Chaoyue Wang, and Dacheng Tao. All Points Matter: Entropy-Regularized Distribution Alignment for Weakly-supervised 3D Segmentation. In: *Thirty-seventh Conference on Neural Information Processing Systems.* Presented in Chapter 4. I designed the research, implemented the systems, conducted the experiments, and wrote the draft of the paper.

(3) **Liyao Tang**, Zhe Chen, Shanshan Zhao, Chaoyue Wang, and Dacheng Tao. Towards Modality-agnostic Label-efficient Segmentation with Entropy-Regularized Distribution Alignment. *Under review.* Presented in Chapter 4. I designed the research, implemented the systems, conducted the experiments, and wrote the draft of the paper.

(4) **Liyao Tang**, Zhe Chen, and Dacheng Tao. On Geometry-Enhanced Parameter-Efficient Fine-Tuning for 3D Scene Segmentation. In: *The Thirty-ninth Annual Conference on Neural Information Processing Systems.* Presented in Chapter 5. I designed the research, implemented the systems, conducted the experiments, and wrote the draft of the paper.

In addition to the statements above, in cases where I am not the corresponding author of a published item, permission to include the published material has been granted by the corresponding author.

Liyao Tang
School of Computer Science
Faculty of Engineering
The University of Sydney
June 1, 2026

As the supervisor for the candidature upon which this thesis is based, I can confirm that the authorship attribution statements above are correct.

Dacheng Tao
School of Computer Science
Faculty of Engineering
The University of Sydney
June 1, 2026

Generative AI attribution statement

The student used ChatGPT for the purposes of text enhancement. The use of this generative AI tool includes spelling corrections, minor sentence restructuring and clarity enhancement. The author confirms that where text was modified by generative AI, the content was reviewed for possible errors, inaccuracies, and bias. The author takes full responsibility for the submitted thesis, confirms the work is their own, and has used generative AI in accordance with University guidelines and policies.

Liyao Tang
School of Computer Science
Faculty of Engineering
The University of Sydney
June 1, 2026

Acknowledgements

Time's worn away. Standing near the end of this journey, I find myself returning to the views along the way, now carved into my memory, and feeling a deep appreciation for each of them.

First and foremost, I wish to express my deepest gratitude to my supervisor, Professor Dacheng Tao. When I first reached out to him in 2019, I could not have imagined how this journey would unfold, nor how profoundly his mentorship would shape the researcher and person I have become. In the years that followed, his unwavering support and steady patience became the quiet foundation on which this thesis was built. "***Make an impact.***" He would say, and he guided me toward the true pathway of research, one that dares to explore the unexplored, treats difficulty not as a barrier but as an invitation, and pursues questions that matter. Through his example, I caught an early glimpse of what it means to work among the finest scientists and researchers, and that glimpse has remained a lasting compass as I move forward. Beyond guidance in research, his generous patience and encouragement, joined with a remarkable passion for discovery, carried me through the moments when progress felt distant and challenges seemed unyielding. For all he has given, I offer my sincere and enduring thanks.

I would also like to extend my sincere appreciation to my co-supervisor, Dr. Zhe Chen. Through countless hours of discussion, he supported me with steady patience and unfailing encouragement, refining my rough ideas into clearer directions, and restoring calm when progress faltered. "***It's okay,***" he would often begin, and with those simple words, he dispelled my doubts and brought me back to what mattered, step by step guiding me toward greater maturity as a researcher. Across every stage of my work, he deepened my understanding of the problems I was trying to solve and the standards by which they should be solved. Each project in this thesis bears his care and effort, for which I am deeply grateful.

I would further like to express my sincere appreciation to my collaborators, Dr. Yibing Zhan, Dr. Baosheng Yu, Dr. Shanshan Zhao, and Dr. Ce Ju. Dr. Yibing Zhan and Dr. Baosheng Yu introduced me to the rich landscape of modern computer vision, broadening my research horizon through countless paper readings and thoughtful exchanges With their support, I brought my first paper to completion. I am also grateful to Dr. Shanshan Zhao, whose insight and high standards sharpened my critical thinking in 3D vision and sustained my pursuit of top tier research. A special note of thanks goes to Dr. Ce Ju, who encouraged me to pursue a doctoral degree as early as 2018, when we first met at Baidu in Beijing. Through our long standing collaboration and his strong mathematical intuition, he deepened my appreciation for theoretical beauty and strengthened my resolve to seek it in my own work.

I extend my sincere thanks to my senior lab mates and mentors: Dr. Zhi Hou, Dr. Haibo Qiu, Dr. Yongcheng Jing, Dr. Yajing Kong, Dr. Zhihao Cheng, Dr. Jing Zhang, and Dr. Tongliang Liu, for their generosity with time, their careful advice, and the many conversations that shaped my work. I am equally grateful to my close mates in the lab: Dr. Xu Zhang, Dr. Shiye Lei, Dr. Yan Sun, Dr. Jianzhi Long, and Mr. Chen Chen, whose companionship brought warmth to long days, and whose questions often sparked fruitful discussions. My sincere thanks go as well to my colleagues and mentors not only within our research lab: Dr. Cheng Wen, Dr. Haimei Zhao, Dr. Qi Zheng, Dr. Gangan Zhu, Dr. Sihan Ma Dr. Jizhizi Li, Mr. Weiyang Chen, Dr. Zhuo Chen Dr. Sen Zhang, Dr. Yufei Xu, Dr. Jinlong Fan, Dr. Haoyu He, Dr. Qiming Zhang, Dr. Benteng Ma, Dr. Shaopeng Fu, Dr. Xikun Zhang, Dr. Kaining Zhang, Ms. Yutong Cao, Dr. Fengxiang He, Dr. Hao Guan, Dr. Xinqi Zhu, Dr. Jiaxian Guo, Dr. Qian Chen, Dr. Zhen Wang, Dr. Shuo Yang, Dr. Meng Lan, Ms. Yue He, Dr. Zeyu Feng, Dr. Yu Cao, Dr. Liu Liu, Dr. Liang Ding; but also across the research labs: Dr. Yunke Wang, Dr. Linwei Tao, Dr. Jinxu Lin, Dr. Xiyu Wang, Dr. Yuemin Wu Dr. Zijian Wang, Dr. Pei Xiaohuan, Dr. Mengyu Zheng, Ms. Leke Tan, Mr. Shuo Zhang, and Dr. Ziqin Zhou; as well as those who I was fortunate to meet during my internships: Dr. Shiwei Liu, Dr. Erdun Gao, Dr. Wen Wang, Dr. Xueqi Ma, Dr. Yaqing Liang, Dr. Muzhi Li, Dr. Yifei Zhang, Dr. Jincheng Xu, Dr. Xianbiao Qi, Dr. Ailing Zeng, Dr. Yang Luo, and Dr. Shengju Qian. Their assistance, encouragement, and willingness to share experience, helped me navigate both the

technical demands and the human rhythm of research. I am also thankful to Ms. Xiaofei Liu, Mr. Greg Ryan, and Mr. William for their kind help with administrative and IT support.

In addition, I am thankful to the University of Sydney for providing the research platform. This research was supported by The Commonwealth through an Australian Government Research Training Program (RTP) Scholarship DOI https://doi.org/10.82133/C42F-K220.

Finally, I am most grateful to my family, especially my mother, Dr. Xiulan Yao, and my father, Mr. Yuanming Tang. Their enduring love and unconditional support have been my quiet refuge, without which this journey, and all it has yielded, would not have been possible. From childhood onward, their steady guidance and the values they instilled in me kindled my first longing to seek the frontiers of human knowledge, and in time led me here. Their love has remained a constant, carrying me through every season, and grounding whatever I have been able to achieve.

Clockie's hands spin aruond non-stop, indicating confusion, furstration, and weakness. But ultimately, people still need to move forward, just like the hands, always pointing ahead. This is where my doctoral journey ends. From now on, it is a future path to walk. Research is trailblazing; it means taking paths your predecessors foreswore, and venturing even further.

I dreamed a starry night,

Sidere mens eadem mutato.

May this journey lead us starward.

Abstract

Understanding the three-dimensional (3D) world underpins embodied artificial intelligence, enabling robotics, autonomous driving, augmented and virtual reality, and, more broadly, spatial intelligence. Yet the most direct representation of a 3D scene can be deceptively simple: a point cloud, *i.e.*, a set of Cartesian coordinates sampled from scene surfaces, from which objects and semantic structure must be inferred. In practice, point clouds are noisy, incomplete, and irregularly sampled, making reliable interpretation fundamentally challenging.

Geometry-grounded world understanding demands semantics that are consistent with metric 3D geometry. Semantic segmentation provides a principled route from geometric measurements to high-level scene understanding. In 3D point clouds, semantic segmentation anchors geometry-grounded world understanding by coupling fine-grained semantics with explicit 3D geometry and confronting a core perceptual challenge: forming structured and coherent interpretation from noisy, incomplete, and irregular samples.

This thesis investigates how explicit 3D geometry can be elevated from a passive input to a source of structure for learning and generalization. We treat geometry as a prior that constrains what a plausible segmentation should look like, modulates how noisy supervision should be used, and guides how models adapt when spatial statistics shift. Building on this view, we advance geometry-grounded scene segmentation along three complementary aspects of the learning problem: (1) the *output*, by enforcing boundary-aware predictions that preserve crisp boundaries; (2) the *supervision*, by using entropy regularization to derive noise-aware learning signals; and (3) the *adaptation*, by modeling shifts in geometric context to enable robust and efficient fine-tuning across downstream scenarios.

First, to impose coherent structure on unorganized measurements, we target crisp semantic boundaries. We introduce Contrastive Boundary Learning (CBL), which enhances boundary discriminability and sharpens region separation, yielding segmentations that better respect

both geometry and scene boundaries. Second, to address ambiguous supervision, we formalize noise in the learning signal. We derive Entropy-Regularized Distribution Alignment (ERDA) within a unified framework of entropy minimization, which enables noise-aware learning and further overcomes geometric perturbations across modalities, including both 3D point clouds and 2D images. Third, to support deployment across diverse spatial structures, we address adaptation under spatial and geometric shift. We propose Geometric Encoding Mixer (GEM) to account for distribution shifts in both spatial patterns and geometric context, enabling efficient and geometry-aware transfer to downstream scenarios.

Overall, this thesis approaches scene segmentation by structuring the outcomes of learning, the supervision that guides learning, and the contexts in which learning occurs, all grounded in explicit 3D geometry. In doing so, we advance a geometry-grounded view of world understanding in which explicit 3D geometry shapes both learning and generalization.

Contents

List of Figures

List of Tables

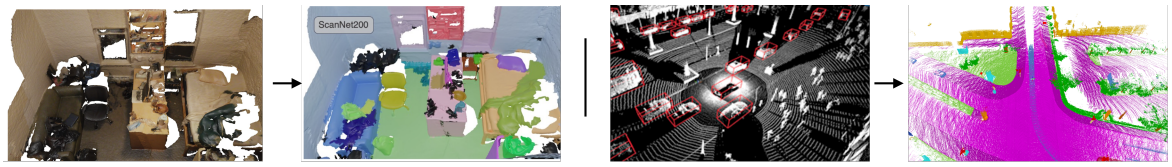
CHAPTER 1

Introduction

Understanding the three-dimensional (3D) world is central to embodied artificial intelligence. Robots must act with physical consequences, autonomous vehicles must reason about geometry in real time, and immersive AR/VR systems must anchor perception and interaction in metric space rather than pixels alone [1, 2]. Across these settings, intelligence hinges on spatial representations that support coherent interpretation and remain reliable under noisy and incomplete sensing.

Yet even the most direct representation of a scene can be deceptively simple: a point cloud, a set of Cartesian coordinates sampled from visible surfaces [3, 4]. From such sparse and irregular samples, the goal is to infer surfaces, objects, and semantic structure that support reasoning and action [3, 5]. This introduces challenges due to occlusion, range-dependent density, and incomplete observations. In particular, point clouds are noisy, incomplete, and irregular in space. The same pedestrian may appear as a dense cluster at close range but only as a few points at distance [6, 7]. The same boundary may be sharp in geometry but ambiguous in semantics, or vice versa [8].

Geometry-grounded world understanding demands semantics that are consistent with metric 3D geometry. Semantic segmentation provides a principled route from geometric measurements to high-level scene understanding [9–11]. It assigns a semantic label to each point, turning raw observations into a structured representation. In 3D, however, segmentation inherits the full complexity of real-world perception. It must contend with irregular sampling, boundary sensitivity, and supervision that can be sparse, noisy, and context-dependent, as in Fig. 1.1. Even when human annotation is available, label ambiguity persists in sensor failures, cluttered scenes, and long-tail categories [12–14]. When supervision is scarce, the challenge



(a) Indoor segmentation **(b)** Outdoor segmentation

FIGURE 1.1. For an indoor scene (a) and an outdoor scene (b), the raw input (left) and the corresponding per-point semantic labels (right) illustrate the effects of noise, irregular sampling, and incomplete observations across sensing contexts.

broadens from accuracy to generalization, from fitting benchmark distributions to deploying in the wild [15–17].

This thesis is devoted to learning geometric semantics through point cloud scene segmentation: mapping irregular 3D measurements to coherent per-point semantics that support downstream reasoning. Its guiding principle is that explicit 3D geometry is not only an input modality but also a source of structure for learning and generalization. Following this principle, the thesis treats semantic segmentation as a learning problem and develops structure along three complementary axes: (i) the *outputs* of learning, via semantic boundaries; (ii) the *supervision* that guides learning, via noise-aware learning signals; and (iii) the *context* in which learning occurs, via geometry-aware adaptation. Together, these axes motivate three research questions (Sec. 1.2) that anchor the thesis, establishing a unified perspective on geometry-grounded world understanding.

1.1 From Points to Representations

Early progress in point cloud understanding clarified both the promise and the limitations of learning directly on sets of points. PointNet [18] established that permutation-invariant processing can yield strong features from unordered points. PointNet++ [19] extended this insight with hierarchical grouping that better captured local structure. Subsequent work explored neighborhood operators and geometric priors, including graph-based aggregation [20, 21], convolution kernels [22, 23], and serializations [24, 25]. More recently, transformer architectures have offered a compelling route to long-range relation modeling and large

context [25–27], while large-scale pre-training has begun to reshape the field by scaling data and models for more general representations [17, 28, 29].

In parallel, two-dimensional vision has provided a playbook for dense semantic learning. Fully convolutional networks and encoder-decoder designs demonstrate how semantics can be predicted at pixel resolution [10, 30], and are further strengthened by multi-scale context and large receptive fields [31]. More recent segmentation frameworks advance high-quality dense features through transformer-based architectures [32–35]. Foundation models have further emphasized scale and generality, with pre-trained visual representations that transfer across tasks and domains [11, 35–38].

These developments suggest a coherent direction for 3D scene segmentation. We can build expressive representations, but we still confront what makes the 3D case distinct. In three dimensions, the sampling pattern is part of the input yet varies across space. The geometry is explicit yet incomplete. The neighborhood structure is neither fixed nor grid-aligned. As a result, semantic understanding depends not only on backbone capacity but also on the structure imposed on learning.

1.2 Scene Segmentation through the Lens of Structure

Across decades of segmentation research, three themes recur, largely independent of backbone choice. For 3D scene understanding, they motivate three research questions:

(1) **Output:** How can semantic partitions preserve sharp and faithful boundaries in 3D?
(2) **Supervision:** How can learning remain stable under intrinsic noise and ambiguity in supervision?
(3) **Adaptation:** How can pre-trained backbones be adapted efficiently under spatial and geometric distribution shifts?

In answering them, the thesis advances the broader aim of using explicit 3D geometry as structure for learning that is accurate, efficient, and adaptable.

First, the outcome must be structured. Segmentation is not only a set of isolated classifications, but a partition of space into coherent regions with clear semantic meanings, often decided at boundaries [39, 40]. Traditional graphical models made this explicit by coupling local predictions through global consistency, and by sharpening boundaries through dense contextual smoothing [41]. Modern deep segmentation inherited the same principle that dense prediction should respect spatial structure rather than merely complete a per-point labeling task [10, 33].

Second, the supervision signal must be structured. Labels in real scenes can be ambiguous. Ambiguity arises from sensor noise, occlusion, annotation inconsistencies, and category overlap, making noise an intrinsic part of supervision [13, 42]. Even widely used datasets such as ScanNet and S3DIS exhibit noisy and inconsistent annotations that can degrade generalization beyond the training distribution [43, 44]. Outdoor benchmarks such as SemanticKITTI further amplify this through motion, sparsity, and range-dependent density [6]. Learning should therefore leverage raw data while being robust to noisy supervision [15, 45].

Third, the context must be structured. Deploying pre-trained models in downstream scenarios inevitably encounters distribution shift [46, 47]. Sensors change, environments change, and the geometric statistics of a scene drift. Adaptation is necessary, yet naive fine-tuning can overfit small downstream sets and catastrophically forget general features. In 2D vision, parameter-efficient fine-tuning and modular design have emerged as practical tools for adaptation, including adapters and low-rank updates [48, 49]. In 3D perception, the need is sharper because downstream distribution shifts often manifest directly in geometry, not merely in appearance.

This thesis treats these three themes as methodological anchors. Each contribution introduces a geometry-grounded structure that mitigates a recurring challenge in point cloud scene segmentation.

1.3 Contributions and Scope

Chapter 3: Structuring Output via Semantic Boundaries. The first contribution concerns the structure of the output. Semantic segmentation is not merely pointwise classification, but a geometric partition of space whose fidelity is often determined at boundaries [41, 50].

This chapter introduces Contrastive Boundary Learning (CBL), a framework that treats semantic boundaries as privileged supervision. We show that boundaries are especially fragile in point clouds. Boundary areas are sparse and difficult to recognize under irregular neighborhoods with varying point density, yet they decisively separate semantics. Quantitatively, we introduce three metrics, mIoU@boundary, mIoU@inner, and B-IoU, to examine model performance on boundary areas. To address the unsatisfactory performance on boundaries, CBL improves boundary discriminability by encouraging representations to separate across boundaries while remaining coherent within regions. By seeking crisp segmentation of 3D scenes, CBL brings boundary-aware learning into the permutation-invariant and non-uniform geometry of point sets.

Chapter 4: Structuring Supervision via Entropy-Regularized Alignment. The second contribution concerns the structure of supervision. In practical settings, dense labels are costly, incomplete, or altogether unavailable. Learning must then draw supervision from weaker sources, such as partial annotations, self-training, and distillation through pseudo-labeling, which can however be affected by noise [13, 45, 51].

This chapter develops Entropy-Regularized Distribution Alignment (ERDA) by formalizing the noise in supervision signals under a unified framework of entropy minimization. By analyzing the noise level of supervision signals and the discrepancies between generated pseudo-labels and model predictions, ERDA yields a simple cross-entropy-based objective grounded in entropy regularization. As ERDA enables noise-aware learning in general, we further extend pseudo-labeling to remain informative under fully supervised, semi-supervised, sparsely supervised, and unsupervised settings. To overcome geometric disturbances, not

only in noisy 3D point clouds but also in heavily augmented 2D images, we devise query-based pseudo-labels that better exploit the ERDA learning. Overall, ERDA treats noise as an intrinsic component of supervision, enabling modality-agnostic learning from raw data.

Chapter 5: Structuring Adaptation via Geometric Context. The third contribution concerns the structure of context. Recent advances in large-scale pre-training have turned transformers into strong general backbones for 3D understanding [17, 29, 36, 52]. Yet deployment rarely matches the training scenarios, as sensors change, environments change, and geometric statistics drift, making adaptation a necessity rather than a choice [46].

This chapter proposes the Geometric Encoding Mixer (GEM), a geometry-aware module designed for parameter-efficient fine-tuning (PEFT) of pre-trained 3D models. We demonstrate that existing PEFT mechanisms generally overlook the irregularity and structural variability of 3D scenes shaped by various sensing protocols and scene geometry. GEM explicitly enriches the positional encoding to capture spatial patterns and integrates scene context via efficient global attention. By adapting to both local structure and geometric context, GEM supports geometry-aware adaptation and represents the first exploration of PEFT for large-scale 3D scene segmentation.

Summary. In summary, this thesis contributes a unified view of geometry-grounded world understanding through three concrete advances.

- A boundary-centered learning framework that improves region separation by structuring representation learning around semantic boundaries.
- A noise-aware pseudo-labeling strategy that leverages entropy regularization to stabilize learning under diverse label regimes and geometric disturbances.
- A geometry-aware adaptation module that supports robust parameter-efficient fine-tuning under distribution shift by explicitly modeling geometric context.

Scope. Geometry-grounded world understanding spans a broad stack of problems, from reconstruction and mapping to tracking, interaction, and decision-making. As outlined above, this thesis takes semantic segmentation of 3D point clouds as the central pathway from

perceived geometry to semantic meaning. Its scope is therefore the learning problem: how to map irregular 3D measurements to coherent per-point semantics, and how to make this mapping boundary-faithful, robust to noise, and adaptable under geometric shift. Accordingly, we intentionally leave out complementary tasks such as 3D reconstruction and SLAM, tracking, object detection, instance or panoptic segmentation, and policy learning or planning, though our methods are informed by or transferable to them. Instead, we focus on a question that is both fundamental and enduring. *How can explicit 3D geometry be used as structure, not merely as data, to support learning that is accurate, efficient, and adaptable?*

Through these contributions, the thesis argues that, in three dimensions, structure is not an auxiliary constraint but the substance of learning itself. By treating geometry as a primary organizing principle for output, supervision, and adaptation, the thesis seeks to advance semantic segmentation toward a more reliable foundation for embodied perception in real-world environments.

1.4 Thesis Organization

The remainder of the thesis is organized as follows.

- Chapter 2 provides detailed background for the scope of the thesis and reviews recent progress in point cloud processing.
- Chapter 3 develops Contrastive Boundary Learning (CBL) and studies how boundary structure improves semantic discriminability and spatial partition quality.
- Chapter 4 presents Entropy-Regularized Distribution Alignment (ERDA) and analyzes how structured and noise-aware supervision enables learning across modalities and label regimes.
- Chapter 5 introduces the Geometric Encoding Mixer (GEM) and examines geometry-aware adaptation under downstream distribution shift.
- Chapter 6 concludes with limitations and future directions, emphasizing geometry as a first-class source of structure for scalable 3D perception.

CHAPTER 2

Background

In this chapter, we review background most relevant to this thesis, with an emphasis on point-cloud-based 3D perception and semantic segmentation. Our goal is not to enumerate tasks in isolation, but to clarify a unifying theme that will recur throughout the dissertation: explicit geometry can act as a source of structure, sitting at the center of the design of methodologies.

Point clouds have become a central pathway to real-world perception, supported by the rapid maturation of depth sensors and 3D reconstruction pipelines. At the same time, point clouds differ fundamentally from images, as they are irregular samples of surfaces embedded in 3D space, typically sparse, unordered, and affected by noise and occlusion. These properties have motivated a family of deep-learning architectures. They can be viewed through a representation-oriented lens, where different methods *impose or recover structure* in different ways in order to make learning feasible and effective.

2.1 Introduction to Point Cloud

Perceiving the 3D world is pivotal to embodied intelligence, underpinning applications such as autonomous driving, robotics, and augmented and virtual reality. Among common 3D representations, *e.g.*, meshes, depth images, and implicit fields, point clouds are particularly attractive because they can be acquired at scale and in real time from LiDAR and RGB-D sensors, and they preserve geometry in a direct and lightweight form.

Following the success of deep learning in 2D vision [53, 54], learning-based approaches have become a dominant paradigm for point cloud understanding. Milestones such as PointNet

and PointNet++ [18, 19] have demonstrated that strong performance is possible even when the input is unstructured, without an image-like grid. These works have grown into a rapidly expanding literature.

A point cloud is essentially a set of 3D samples, often augmented with attributes such as color or intensity. Compared with images, point clouds exhibit several distinctive difficulties [4, 9, 18]. (1) *Irregularity*: points are unstructured and unordered, not lying on a canonical lattice; (2) *Sparsity*: observations and sampling typically lie on thin surface manifolds, leaving most of the 3D volume empty; (3) *Varying density and incompleteness*: sampling density varies with range, viewpoint, and occlusion, leading to varying and missing spatial distribution; (4) *Measurement noise*: sensor artifacts and clutters introduce outliers and missing regions. These factors complicate feature extraction, neighborhood reasoning, and the coherent understanding of scene semantics.

Many successful architectures can be interpreted as different strategies for injecting structure into irregular geometry. In particular, we group methods into three representative families according to the representation they construct. The first are *multi-view networks*, which project points onto one or more 2D views, perspective images or range images, and then leverage mature designs from 2D vision. The price is that projection may discard or distort the explicit spatial relations among points and introduce occlusions during perspective projection [4]. The second are *voxel networks*, which directly discretize the space into structured 3D grids, trading geometric fidelity for efficiency and resolution [55]. The third are *point-based networks*, which retain the raw samples and operate directly on unordered points, using permutation-invariant mappings, hierarchical set abstraction, and various local operators.

In addition to the choice of representation, we further complement these representation-oriented taxonomy by discussions on geometry-guided learning signals and generalization under imperfect supervision and distribution shift.

In the following, we review common datasets with point cloud data in Sec. 2.2, examine three representative model families, as summarized in Tab. 2.1, extend the discussion to spatial intelligence in Sec. 2.6, and conclude with a brief summary.

		Trends and Focuses	Advantages	Challenges
Sec. 2.3	Multi-View Network	Multi-view Specific view	Dense, compact and ordered representation. Well-established research with mature network designs.	Losing 3D spatial structure. Introducing occlusion and cluttering in view projection. Resolution limited by images.
Sec. 2.4	Voxel Network	Grid voxelization Non-grid voxelization	Ordered representation. Retaining rough spatial structure. Easy extension from 2D CNN to 3D CNN.	Losing detail due to limited resolution. Computation burden due to additional dimension. Sparsity due to large unoccupied space.
Sec. 2.5	Point Network	Point-wise MLP Point hierarchy Local operation Additional modality	All spatial information and detail retained. Light-weight and affordable computation.	Unordered representation demanding new designs. Sparsity due to large unoccupied space. Not explicitly representing the empty space.

TABLE 2.1. Overview on Networks

2.2 Popular Datasets

We briefly review commonly used datasets and benchmarks for point cloud understanding. There are usually two sources for creating a dataset, synthetic assets, such as CAD models, and real-world captures from RGB-D or LiDAR sensors. While the former provides scalable and clean geometry, the latter faithfully reflects the noise, incompleteness, and complexity of the real-world environment.

Object-centric benchmarks. These datasets provides collection of 3D objects and shapes. ModelNet [56] and ShapeNet [57] are widely used for object classification and part segmentation on CAD models. PartNet [58] further provides fine-grained and hierarchical part annotations aligned with semantic trees. ScanObjectNN [59] complements these synthetic benchmarks with object instances cropped from real scans. Objaverse [60] scales the object collections to internet-scale 3D assets and has become a common source for large-scale 3D pre-training.

Indoor scenes. Indoor datasets feature scenes with cluttered objects and many serve as common benchmarks for 3D semantic segmentation and scene understanding.

ScanNet [43] and S3DIS [44] are widely adopted, covering diverse indoor scenes with per-point annotations. SUN RGB-D [61] aggregates multiple RGB-D sources with dense annotations, and Matterport3D [62] provides high-quality reconstructions across buildings. Recent extensions broaden the semantic and geometric scope. ScanNet200 [14] expands ScanNet to a long-tail 200-class vocabulary. ScanNet++ [63] improves capture fidelity with high-resolution

imagery and laser scans. To scale the data diversity and coverage beyond curated scans, synthetic and video/mobile captures are increasingly used. Structured3D [64] offers large synthetic indoor layouts with photorealistic renderings, and RealEstate10K [65] collects posed internet videos to support reconstruction of large-scale 3D scenes. ARKitScenes [66] and EFM3D [67] provide mobile and ego-centric sequences, broadening evaluation beyond static scans.

Outdoor scenes. Outdoor datasets cover large open environments with diverse geometry, dynamics, and sensing conditions.

Semantic3D [68] provides dense terrestrial scans captured by static high-resolution LiDAR. Mobile mapping and aerial platforms broaden the scale and viewpoint variation, ranging from Toronto-3D [69], NPM3D [70], to city-scale datasets such as SensatUrban [71]. Autonomous driving benchmarks provide multi-modal sensor suites and support a range of understanding tasks, including detection, segmentation, tracking, and scene flow. SemanticKITTI [6] supplies point-wise annotations for LiDAR sequences derived from the classic KITTI benchmark [72]. Other comprehensive autonomous driving datasets include nuScenes [7], Waymo [73], and A2D2 [74]. To reduce annotation cost in these settings, weakly labeled variants such as ScribbleKITTI [75] study LiDAR segmentation under label efficiency. Large-scale unlabeled corpora such as Argoverse2 [76] further support self-supervised pre-training.

2.3 Multi-View Network

These networks first project the 3D object to one or multiple views and then process with 2D CNNs. The extracted feature maps can either directly produce the desired output or be reprojected back to the point cloud for point-wise results.

By projecting to 2D image plane, the point cloud representation is no longer unordered or sparse, and can take advantage of the established study in image processing.

The downside of such projection is the loss of explicit 3D structure. In a single image, the 2D relation usually corresponds to multiple valid 3D scene and the same pixel distance on the image can correspond to very different Euclidean distance in 3D space. As a result, apart from the information loss caused by limited resolution and potential occlusion, it also incurs obscured 3D structure, resulting in the inconsistency between 2D measure and the 3D scene.

To overcome these drawbacks, there are two forms in the projection from continuous Cartesian coordinate to discrete image coordinate, which differ by how the view is chosen for such projection.

Multi-view processing. Multi-view is obtained by sampling a series of image planes, on which the point cloud is projected. A common choice is to select or sample a series of camera poses, providing descriptive descriptions from different perspectives and thus reducing ambiguity.

MVCNN [77] is a seminal work, which takes several perspectives of a given 3D object and processes them with standard 2D CNN. The results from each view are finally aggregated by a max pooling into a global feature vector for classification. For segmentation, multi-view CNN can be combined with surface-based CRFs for part segmentation [78], where CNN predicts per-view confidence map that are backprojected to 3D surface, and CRFs consider geometric consistency in producing the final result. Similarly, SnapNet [79] and [80] target on large-scale segmentation, where a series of virtual camera positions are sampled for view rendering and per-view segmentation result are re-projected back to original point cloud. Multi-view features can also enhance the 3D backbones via feature fusion. Representative backbones include 3DMV [81] and MVPNet [82], which reproject multi-view CNN features into 3D for point-wise prediction.

Front-view processing. Front view (FV) is obtained by spherical or cylindrical projection, resulting in an input image with the range information encoded in image channel. It converts point cloud into a dense and compact representation, yet other challenges emerge, such as varying scale, occlusion and clutter, similar to 2D image [83, 84].

SqueezeSeg [85] constructs FV by spherical projection, followed by a SqueezeNet [86]-based CNN and a Conditional Random Fields (CRF) for segmentation. The image segmentation is then reprojected to points and a clustering algorithm is then applied to further perform instance segmentation on the semantically segmented points.

2.4 Voxel-based Network

Such networks first preprocess point cloud into 3D voxels, i.e. voxelization, which is done by placing a 3D grid in the space to slice and assign points into voxels. 3D convolution layers are applied on voxels to extract feature volumes, which are then fed into task-specific prediction head. For segmentation, models need to consider per-point label given its voxel feature.

Indeed, with 3D convolution, voxel-based network can extend classic 2D framework to voxelized point cloud data, as it can be regarded as a direct generalization of 2D convolution to 3D space to perform hierarchical feature extraction. It also offers an easy construction of local neighborhood for each point, equipped with 3D convolution to explore the local context.

Nonetheless, it heavily relies on the discretization imposed by the voxelization and the 3D convolution is usually slower than the 2D version, due to its large search space caused by additional z axis. These constraints result in slow inference if using high resolution, or loss of detail if using low resolution, thus composing a trade-off between computation burden and performance [55].

As a result, apart from using a uniformly separated grid, there are also other approaches to perform voxelization. The discussion is grouped by these voxelization approaches, including grid voxelization and non-grid voxelization.

2.4.1 Grid Voxelization

Grid voxelization is arguably the simplest way. It places a uniform 3D grid and each cell of the grid becomes a voxel. The feature of each voxel are extracted from the points inside. For example, the occupancy feature is to place a 1 if there is any point inside and 0 otherwise.

Additionally, in order to achieve the balance in computation and preserving detail, it requires to choose a suitable pre-defined resolution according to the density of point cloud at hand. It is thus difficult to describe the non-uniformly distributed points, especially when point density varies greatly across different upcoming point cloud.

Direct processing. These methods directly use 3D convolutions, and focus on the design of 3D CNN as well as matching. extracted feature volumes to task-specific requirement.

This line of work starts from a classic 3D CNN approach based on such grid voxelization [56]. Following works [55, 87, 88] further explores designs to enhance the resolutions and improve the efficiency.

For segmentation, it is commonly required to re-project from voxel to point. 3D CNN approach [89] starts by assigning the same label to all points within a voxel. Later works focus on improving the mapping from voxel features back to original points and enforces consistency [90, 91].

As a widely used baseline, SparseUNet [92] modernizes such 3D convolutional backbones by providing an efficient CUDA implementation.

Sparse convolutions. To improve efficiency and overcome the sparsity in 3D space, various forms of sparse convolutions are explored, enable high-resolution voxel processing with affordable compute.

Early work [93] generalizes 2D sparse convolutions to 3D voxels, avoiding the wasteful calculation on empty space. Vote3Deep [94] casts 3D convolution as feature-centric voting from occupied locations and obtains the result by vote aggregation, thus avoiding the iterative query and greatly reducing the complexity of 3D convolution to be proportional to the number of non-empty voxels. SECOND [95] further enables GPU acceleration in sparse 3D convolution, making more complex network design affordable in real-time 3D detection with voxels.

Meanwhile, submanifold convolutions [96] further restricts the output locations and can thus control and maintain the degree of sparsity throughout the processing, thus further improving

the efficiency. SSCN [97] then introduces such submanifold convolutions into point cloud segmentation and demonstrates its ability and efficiency.

Besides, 3D atrous convolutions are also explored for lightweight computation. VoxSegNet [98] designs to cover a dense cubic receptive fields without holes while only using 3D atrous convolutions.

2.4.2 Non-grid Voxelization

There are also other approaches towards voxelization apart from placing a uniform grid, such as using space-partitioning data structure, usually a tree, or transforming the Euclidean space into permutohedral lattice. These methods avoids the use of uniform grid to circumventing the constraints.

Space partitioning. Such approach addresses the problem of fixed resolution by allocating more voxels for space with. points and less for empty space. With such a dynamic resolution, it can then avoid wasting compute at vacancy and can afford better resolution and representation ability for space with dense points.

KD-Networks [99] uses the construction of K-D tree to perform dynamic voxelization, which helps reveal the spatial structure of the point cloud. It also proposes a smart 3D convolution that leverage the KD-tree structure to speed up the computation. OctNet [100] uses a hybrid grid-octree approach where a shallow octree is constructed for each cell of the 3D grid. With constraining the maximal depth of octree, it can afford deep network. Similarly, O-CNN [101] proposes to use octree to split the point cloud and designs an efficient convolution that utilizes the octree structure.

Thanks to the efficient allocation of representation, octree representation enables efficient attention that scales to large transformer backbones [102].

Permutohedral lattice. The construction of lattice can be regarded as projecting a voxelization with uniform grid to the hyperplane $H : \mathbf{p} \cdot \mathbf{1} = 0$, where $\mathbf{p}$ is the plane normal and $\mathbf{1} = (1, 1, 1)$ in Euclidean space.

Splatnet [103] proposes to use bilateral convolution and fuse with image feature by converting feature maps to the same lattice space. Later works [104, 105] improves by further projecting point cloud features to the lattice space.

2.5 Point-based Network

These networks directly process point cloud without any discretization, hence retaining the full detail to the network. At the same time, they need to handle the intrinsic challenges in point cloud, such as unorderedness and sparsity.

To overcome unorderedness, PointNet [18] sets up an efficient framework for direct processing on point cloud using point-wise MLPs. Further research then explores to better capture spatial structure described by points, which include constructing hierarchical processing and developing local-aware operation in point cloud. Besides, additional data can also be fused with point cloud to further improve the network ability.

Therefore, we first introduce the point-wise MLPs as a basis and then discuss point hierarchy, local operation and additional modality.

2.5.1 Point-wise MLP

Point-wise MLPs is the basic form of point-based networks, an efficient architecture to obtain global descriptor directly from point cloud without any projection or voxelization, which is proposed by the seminal work, PointNet [18]. Its essence is to apply to each point a series of shared MLPs to extract per-point feature, followed by a max pooling to aggregate into a global feature encoding the point cloud semantic. To bring the global context back to each point, the global feature is concatenated to per-point features and segmentation cam be performed with a few more shared MLPs. Due to max pooling, such architecture can base the prediction on a set of key points and is thus robust to outliers and missing data. A parallel work, DeepSet [106] reveals, from a theoretical view, that the key for point-based network is being input permutation invariant and equivariant. Concretely, point-based network can be

formulated as

$$f(\mathcal{P}) = \rho\left(\{\phi(p), p \in \mathcal{P}\}\right) \tag{2.1}$$

Here, $\mathcal{P}$ is the point cloud; ϕ is the operation applied to each point, such as MLPs ; and ρ is an permutation invariant operation, such as max pooling. The output $f(\mathcal{P})$ can be followed by other task-specific network, such as fully connected layers for classification and further point-wise operation for segmentation.

In the direct follow-up works, various more advanced features shows improvements and better generalization, such as the moment of point coordiantes [107] and rotation-invariant feature [108].

2.5.2 Point Hierarchy

Spatial hierarchy is found to be vital, because PointNet [18] uses a single max pooling to obtain the final descriptor from the entire point cloud and can suffer from insufficient exploitation in local and fine-grained pattern. Especially, the representability is restricted by the length of max-pooled descriptor and local structure is missing.

One explicit and popular way to address such problem is to perform hierarchical processing on point cloud. Compared with using only global feature, hierarchical processing extracts features at multiple scales and is able to process large scan taken in real-world scenario, where the point cloud describes complex scene with multiple objects.

Overall, point hierarchy usually takes a downsampling-upsampling structure. The global descriptor could be obtained at the end of the downsampling stage and upsampling part can recover the lost points for per-point prediction.

Despite there exists a natural hierarchy on 2D images by decreasing the resolution, hierarchy in point cloud is open to various construction and we find there are mainly two popular approaches, sampling-grouping and space partitioning.

Sampling and grouping. Such procedure is to sample a few representative points at each stage, each of which becomes the. center point and explores local context by considering its

neighbor points (grouping). PointNet++ [19] is a pioneer and uses fartherest point sampling (FPS) to sample a set of center points at each stage and creates local point sets via K-NN. It utilizes a simplified PointNet [18] to extract feature from local point set at various downsampling stage. For each upsampling stage, it uses interpolation and concatenates with the corresponding low-level point-wise feature via skip connection. The interpolation can be also replaced by other upsampling operation such as nearest upsampling.

RandLA-Net [109] proposes to use random sampling for much faster inference in large-scale point cloud, and designs novel local feature aggregation based on attention and residual connection. HPEIN [110] proposes an additional edge branch in the upsampling stage. It establishes a coarse graph on points of last downsampling stage and upsamples the edge features along with the points. It effectively upsamples into a larger and denser graph with both point and edge features. S-NET [111] proposes a learnable sampling module and incorporates a sampling loss for joint training with task-specific loss. AdaCoSeg [112] achieves adaptive shape co-segmentation based on PointNet++ [19]. With weak supervision, it overcomes inconsistent training labels by iteratively minimizing a group consistency loss defined over a set of shapes and their pre-segmented parts. DPAM [113] proposes an attention-based downsampling, where each sampled point is interpolated by all input points of current hierarchy. Similarly, PAT [114] proposes Gumbel Subset Sampling, which introduces Gumbel-Softmax and Gumbel reparameterization into the attention.

Meanwhile, the extracted features show effectiveness in deriving semantic clusters. SPGN [115], a pioneering work, novelly represents instance segmentation via predicting a similarity matrix that scores the similarity between points. Yet, the similarity matrix size grows quadratically with the point number, becoming a high memory burden. To improve efficiency, metric learning is introduced and utilized by clustering algorithms. ASIS [116] benefits instance and semantic segmentation from each other, by fusing both the features and the clustering results with each other. JSIS3D [117] explores a more explicit joint optimization, feeding per-point semantic and instance feature into a multi-value CRF to jointly perform semantic and instance segmentation. The seminal work, PointGroup [118], further advances the bottom-up

clustering paradigm by introducing a dual-set clustering algorithm, grouping points based on both their original and offset-shifted coordinates to better separate adjacent objects.

Combining with space partitioning. Voxelization can be viewed as an naive space partitioning approach to assign points into different groups and can deeply integrates voxel represnetation into point-based network.

On one hand, it provides an approximated yet fast access to local neighborhood. On the other hand, voxel can be interpreted as a point with feature locating at the voxel center, thus introducing 3D convolutions to perform hierarchical processing. A specific advantage of having such conceptual voxel is to explicitly account for empty space, while empty space is never presented to classic point-based network.

PVCNN [119] reduces large memory footprint of voxels and avoids irregular memory access for points. Concretely, at each stage, it fuses point features from two branches, one interpolated from voxel features after 3D convolutions and the other one extracted by point-wise MLPs. Part-A^2 Net [120] views voxel as a point at voxel center, so that it can use sparse convolution to extract features and PointRCNN [121] to generate proposals. It enable part awareness by exploring the relative locations of foreground points w.r.t. to their corresponding 3D boxes, and by RoI-aware pooling to aggregate features and capture the geometry, including empty space, of the proposal. Similarly, PV-RCNN [122] considers each non-emtpy voxel as a point at voxel center and samples a set of keypoints to aggregate the extracted voxel features. With proposal given by 3D CNN, 3D RoI-pooling also considers each 3D bin center as a point to aggregate neighboring keypoints. With rich context from 3D CNN and fine location from keypoints, it achieves leading performance.

2.5.3 Local Operators

Local operation is applied to a local point subset to explore local contextual information. As being local-aware, it can increase the receptive fields by stacked local operations in network. These operations can thus bring more powerful representation ability into the point-wise MLP and point hierarchy architecture.

With different formulation, the operation applied on the local point set can be viewed as point-based network, convolution and/or graph network. We provide separate discussion for each of these perspectives. Additionally, there are also methods focusing on neighborhood construction, instead of the widely used KNN and radius neighborhood.

Point networks. Small point-based network is the most general formulation of local operation, which takes only local. point as input and is shared across local point sets. It can be formulated as

$$f_p = \rho(\{\phi(p, q); q \in \mathcal{N}(p)\}) \tag{2.2}$$

Here, p is the chosen center point; $\mathcal{N}(p)$ is its neighbor point set; ϕ a shared operation (e.g. MLPs); and ρ a permutation invariant operation (e.g. pooling or summation) to obtain the output feature f_p. We note that K-NN and radius neighborhood are commonly used to construct $\mathcal{N}(p)$ due to their simplicity.

As similar to the formulation of point-based network in Eq. (2.1), a natural implementation is to use a simplified PointNet [18], e.g. MLPs with pooling, to perform local feature aggregation, as in PointNet++ [19]. DensePoint [123] densely concatenate all previous point features as an enhancement. LSANet [124] proposes to learn a spatial attention to weight the local neighbors of a point.

Convolutions. To adapt the success of CNNs to irregular point sets, significant effort has been made to parameterize local aggregation through learnable kernels. Under Eq. (2.2), a general convolution can be written as:

$$f_p = (f \circ g)(p) = \sum_{q \in \mathcal{N}(p)} g(p - q) f(q), \tag{2.3}$$

where p is the center point, $q \in \mathcal{N}(p)$ is a neighbor, $f(q)$ denotes its feature, and $g(\cdot)$ is a kernel function defined over relative coordinates. With this definition, two common instantiations emerge.

Typically, KPConv [22] structures kernel weights as a feature with local location, a kernel point. Given a local point set, it explores and aggregates the interaction between each pair

of input point and kernel point. Similarly, ConvPoint [125] associates location to kernel weights, but apart from correlating kernel weights with point features, it separately processes the spatial relation to provide dynamic spatial weights.

From another perspective, MC Conv [126] formulates convolution on points as a Monte-Carlo integration problem, where MLPs map local coordinate $q - p$ into kernel weights. It also weights the points by reversed point density to account for the underlying non-uniform distribution. Following such formulation, PCNN [127] improves by using Gaussian function as radial basis functions (RBF) for points. Similarly, RS-CNN [128] also performs a weighted sum in convolution, where weights are learned from relation between p and each neighbor q; PointConv [129] also learns the weights from point density and provides a more efficient implementation.

Stacking such convolutional operators increases the receptive field, enabling hierarchical and multi-scale context modeling. Recent conv-style backbones, such as PointNeXt [130], revisit these designs with modern training and scaling strategies, setting as a strong baseline.

Attentional operations. Attention complements kernelized convolutions with content-adaptive weighting of local neighborhoods. Early backbones, such as RandLA-Net [109], already employ attentive pooling inside local feature aggregation to emphasize informative neighbors while remaining efficient. Point transformer families elevate this principle to the backbone. Point Transformer (PT) [26] formulates vector attention over k-NN neighborhoods with relative position encoding. Successors, such as Stratified Transformer [131], PTv2 [132], and PTv3 [25], further improve efficiency and scalability via stratified sampling, grouped attention, patchification and serialization schemes, as well as lightweight positional encoding.

Despite their name, these models are still predominantly local, as full global attention over million points of a scene is prohibitive. Long-range context is accumulated through stacking, hierarchical downsampling, or window/partitioned attention [25, 131].

Ordering and serializations. There are attempts to establish order in g, by serializing neighbor points.

Some methods make the use of local surface to introduce 2D convolutions. [133] proposes tangent convolution for efficient segmentation, which projects neighbor points to the local tangent plane defined by the normal of center point and interpolates into a tangent image for 2D convolutions. Similarly, A-CNN [134] proposes annular convolution, using ordered and constrained K-NN to construct neighborhood. Specifically, it considers a ring-shape neighborhood and projects neighbors to the local surface plane, which are then processed in counter-clockwise order. FPConv [135] flatten the local patch of points onto a 2D grid plane by dynamically predicting projection weights, followed by regular 2D convolutions for efficient processing.

While some others directly establish order in 3D space by space partitioning. PointSift [136] splits point neighborhood into a $2 \times 2 \times 2$ cube based on octree and incorporates SIFT descripter, and ShellNet [137] divides radius neighborhood by a series of concentric spherical shells.

Additionally, point features can be used in ordering. SFCN [138] performs graph convolution with neighbor points sorted by their feature similarity with the center point, and uses balanced binary tree to determine nodes order in graph coarsening. PointCNN [24] proposes $\mathcal{X}$ transformation to transform local point set into ordered features to match the order in convolution weights, with the transformation matrix dynamically produced by MLPs based on point feature. Similarly, [139] applies MLPs on local point set to estimate a transformation of points from geometry space to discret kernel space.

More recently, PTv3 [25] establishes the state-of-the-art performance by serializing the whole point cloud based on space filling curves. It thus enables efficient patchification of unordered point cloud and can leverage the advancement in training large transformer models at scale. PTv3 further demonstrates the effectiveness of scaling with large-scale pre-training [17, 28, 29].

Graph model. Graph can effectively describe the local point sets, where each point in the local region is considered as. a vertex and the relation between center p and each neighbor

point q is explicitly modeled as an edge. Recently, graph network is also an emerging research field, introducing powerful operations such as graph convolution.

ECC [140] is the first to introduce graph convolution in point cloud classification, where the convolution weights of each neighbor is conditioned on its edge label. The seminal work DGCNN [20] learns the edge feature by MLPs on pair of points, which is then aggregated to be the feature for center point. Further improvements include adding shortcuts between different semantic level for multi-scale features [141], capturing richer spatial information from neighboring points [142], introducing attention mechanism with shape context [143], and creating densely connected graph for better point-wise correlation [144],

2.6 From 3D Modeling to Spatial Intelligence

Beyond the representation-centric taxonomy above, recent work increasingly scales point cloud backbones through self-supervised pre-training and connects 3D geometry to foundation models. These trends position point transformers as geometry-grounded spatial encoders. They produce queryable and compositional spatio-semantic representations, and increasingly support open-vocabulary recognition and language-conditioned interaction in embodied settings.

Large-scale pre-training for 3D point transformers. Recent work increasingly treats 3D backbones as scalable transformer encoders that benefit from self-supervised pre-training on large and diverse corpora. Early contrastive approaches such as CSC [145] and Point-Contrast [146] show that unsupervised pre-training can transfer across datasets and tasks for scene understanding. More recent masked modeling paradigms [52] further adapt the language/image pre-training recipe to irregular 3D inputs by predicting masked geometry or features. Several object-centric works include Point-BERT [147] with masked point modeling, point-MAE [148] with masked auto-encoding on points, as well as hierarchical extensions [149]. ReCon [150] further bridges generative and contrastive paradigms by using reconstruction-guided distillation to improve scalability and generalizability. Masked Scene Contrast [151] explores the combination of contrastive learning with masked point modeling

at the scene level. Sonata [17] explicitly mitigates geometric shortcuts during pre-training and scales both the data and model to obtain high-quality features that supports effective linear probing and efficient adaptation.

Multimodal pre-training and open-vocabulary alignment. Another direction aligns 3D features to stronger 2D vision-language spaces to inherit semantic priors. At object level, ULIP [152, 153] learns unified representations across language, images, and point clouds, while PointCLIP [154] and its successor [155] transfer CLIP knowledge via multi-view projections or distillation. For scene-level semantics, OpenScene [156] aligns dense 3D points with CLIP pixel/text embeddings to enable open-vocabulary queries, and OpenMask3D [157] extends this paradigm to open-vocabulary instance segmentation by lifting features of foundation models based on masks proposals from SAM [11]. Concerto [29] further improves Sonata by distilling rich 2D semantics from DINOv2 [37] into 3D representations with the help of feed-forward 3D reconstructions [1, 158, 159]. These approaches blur the boundary between representation learning and supervision. Features or tokens from large foundation models serve as the labels, while geometry provides the spatial scaffold to localize and aggregate them.

From point clouds to high-level and embodied understanding. Building on multimodal alignment, a new wave of 3D-capable LLM/VLM systems aims to accept point clouds (objects or scenes) as first-class inputs and to output language-conditioned interpretations suitable for interaction and robotics. Representative examples include PointLLM [160] and MiniGPT-3D [161] for instruction following at the level of 3D objects, as well as ShapeLLM [162] and its extension [163] for embodied interaction. Scene-oriented systems, such as SpatialLM [164], encode scenes into 3D tokens and couple with LLM reasoning to support grounding, captioning, question answering, and scene parsing via structured outputs like code. In parallel, recent efforts study how explicit 3D tokens can augment visual token streams in multimodal models [165]. A common design pattern is to first encode geometry with a pre-trained 3D backbone, then compress or tokenize 3D features, and finally align to LLMs with lightweight projectors or parameter-efficient tuning. This line of work provides a natural endpoint for geometry-grounded perception, where spatio-semantic representations

not only assign labels, but also support compositional querying, referencing, and reasoning in embodied settings.

Deployment under noise and shifts. Spatial intelligence is ultimately evaluated in deployment, where observations, label spaces, and environments evolve beyond the training distribution [46]. To be reliable beyond curated benchmarks, despite, such systems must transfer across sensors and environments, and adapt under limited or indirect supervision. While large-scale self-supervision and multimodal distillation demonstrates superior generalizability, 3D-related annotation remains limited, costly and can be inconsistent [12]. It makes the scaling of large-scale self-supervision and multimodal distillation particularly hard if relying on only curated datasets. As a result, it makes the learning under noisy or sparse supervision a primary concern [166, 167], and the transfer to new environments vital.

For the first concern, various label-efficient methods have been explored for 3D vision [75, 168–170]. For the latter problem, it is commonly formalized through domain adaptation [171–173]. During the era of large models, parameter-efficient fine-tuning emerges as an practical alternative [174–177]. Notably, dense labeling tasks such as semantic segmentation often serve as the first and most sensitive probe of these shifts, making robustness to shift and weak/indirect supervision a central requirement for geometry-grounded perception in the wild [16, 170].

2.7 Summary

In this chapter, we review common datasets and representative network families for point cloud understanding, with an emphasis on semantic segmentation. Multi-view methods exploit mature 2D backbones through projection; voxel methods discretize space and apply 3D, often sparse, convolutions for context modeling; and point-based methods operate directly on raw points via permutation-invariant mappings, hierarchical sampling, and local operators such as convolution and attention. Across these paradigms, a unifying theme is the pursuit of *structure* that renders irregular geometry semantically learnable. We also discussed the emerging scaling trends, including self-supervised pre-training, multimodal alignment, and point-cloud

interfaces to LLM/VLM systems, which increasingly position point transformers as geometry-grounded spatial encoders for embodied 3D understanding. Finally, we highlighted how supervision noise and distribution shifts constrain deployment in the wild, motivating the geometry-guided learning objectives, learning under imperfect supervision, and adaptation mechanisms developed in the subsequent chapters.

CHAPTER 3

Structuring Output via Semantic Boundaries

Following the thesis view that explicit 3D geometry serves as a source of structure, we begin with the structure of the model output. In scene segmentation, this structure is most explicit at semantic boundaries, which is thin yet difficult regions where geometry and semantic ambiguity meet, and where small local errors can distort the global partitions. In this chapter, we thus explore structural coherence in the model output. From a geometry-grounded view, scene segmentation is not only about improving average accuracy, but also about enforcing semantic structure in the model predictions, especially for 3D models.

Point cloud segmentation is fundamental in understanding 3D environments. However, current 3D point cloud segmentation methods usually perform poorly on scene boundaries, which degenerates the overall segmentation performance. In this chapter, we focus on the segmentation of scene boundaries. Accordingly, we first explore metrics to evaluate the segmentation performance on scene boundaries. To address the unsatisfactory performance on boundaries, we then propose a novel contrastive boundary learning (CBL) framework for point cloud segmentation. Specifically, the proposed CBL enhances feature discrimination between points across boundaries by contrasting their representations with the assistance of scene contexts at multiple scales. By applying CBL on three different baseline methods, we experimentally show that CBL consistently improves different baselines and assists them to achieve compelling performance on boundaries, as well as the overall performance, *e.g.* in mIoU. The experimental results demonstrate the effectiveness of our method and the importance of boundaries for 3D point cloud segmentation. Code and model will be made publicly available at https://github.com/LiyaoTang/contrastBoundary.

3.1 Introduction

3D point cloud semantic segmentation aims to assign semantic categories to each 3D data point, while robust 3D segmentation is very important for various applications [4, 9], including autonomous driving, unmanned aerial vehicles, and augmented reality.

However, despite that various point cloud segmentation methods have been developed, little attention has been put on boundaries in 3D point clouds. Accurate segmentation on scene boundaries can be of great importance. Firstly, a clean boundary estimation can be beneficial for overall segmentation performance. For example, in 2D image segmentation, accurate segmentation on boundary is the key to generate high-fidelity masks [178–180]. Secondly, compared to object categories that usually have a large portion of 3D points, such as buildings and trees, erroneous boundary segmentation could affect the recognition of object categories with much fewer points (*e.g.*, pedestrians and pillars) to a greater extent. This can be particularly hazardous for applications like autonomous driving, *e.g.*, crashing into curbs if boundaries are recognized inaccurately by a self-driving car.

Unfortunately, most previous 3D segmentation methods generally overlook the segmentation on scene boundaries. Though a few methods have considered boundaries, they still lack an explicit and comprehensive investigation to analyze the segmentation performance on boundary areas. They also perform unsatisfactorily on the overall segmentation performance.

Therefore, to deliver a more thorough study of the segmentation on boundaries, we first explore metrics to quantify the segmentation performance on scene boundaries. After revealing the unsatisfactory performance, we propose a novel Contrastive Boundary Learning (CBL) framework to help optimize the segmentation performance on boundaries particularly, which also consistently improves the overall performance for different baseline methods.

In particular, current popular segmentation metrics lack specific measurements on boundaries, making it difficult to reveal the boundary segmentation quality in existing methods. To make a clearer view on the performance on boundaries, we calculate the popular mean intersection-over-union (mIoU) for boundary areas and inner (non-boundary) areas separately.

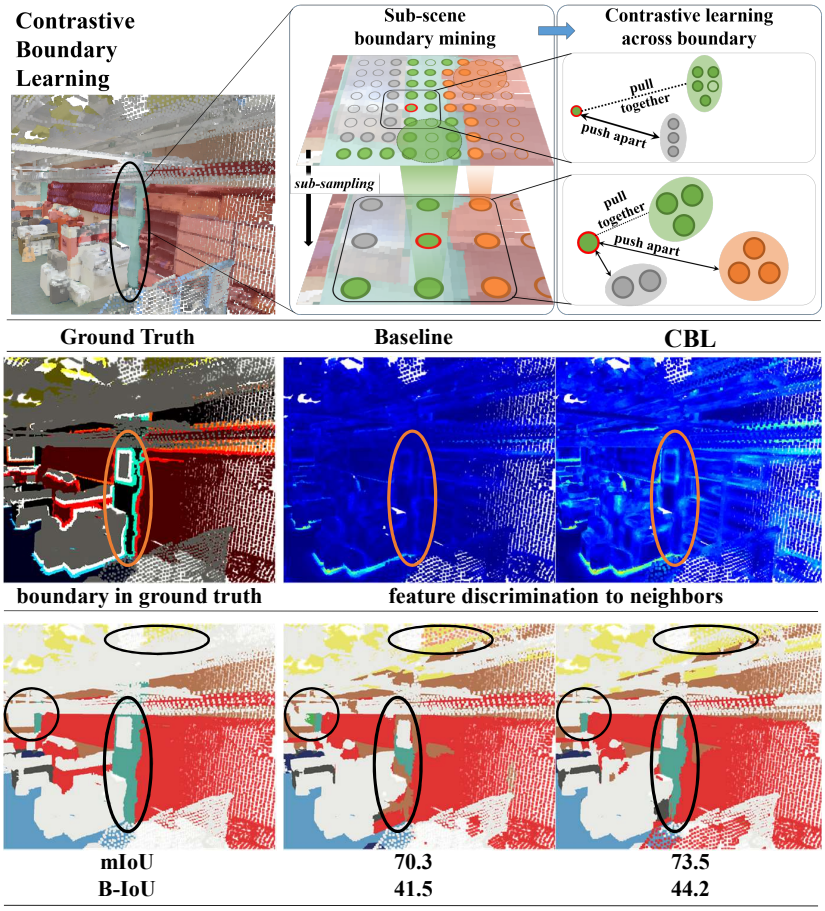


FIGURE 3.1. Contrastive Boundary Learning (top) discovers boundary from ground truth in each sub-sampled point cloud, *i.e.*, sub-scene, through the sub-sampling procedure. By imposing contrastive optimization on boundary areas at multiple scales, CBL enhances the feature discrimination across boundaries (middle). Without an explicit boundary prediction, CBL improves boundary segmentation and achieves better scene segmentation results (bottom). The visualization is conducted on S3DIS testset Area 5.

By comparing the performance on types of areas as well as the overall performance, the unsatisfactory performance on boundary areas can be directly revealed. Moreover, to describe the performance on boundaries more comprehensively, we consider the alignment between the boundary in the ground truth and the boundary in model segmentation results. Therefore, we introduce the popular boundary IoU [179] score (B-IoU) used in 2D instance segmentation for evaluation, which also gives a much lower score compared with the overall performance in mIoU.

After identifying the boundary segmentation difficulties, we further propose a novel contrastive boundary learning (CBL) framework to better align the boundaries of model predictions with ground-truth data's boundaries. As shown in Fig. 3.1, CBL optimizes a model on the feature representation of points in boundary areas, enhancing the feature discrimination across the scene boundaries. Furthermore, to make model better aware of the boundary areas at multiple semantic scales, we also develop a sub-scene boundary mining strategy, which leverages the sub-sampling procedure to discover boundary points in each sub-sampled point cloud, *i.e.*, sub-scene. Specifically, CBL operates across different sub-sampling stages and facilitates 3D segmentation methods to learn better feature representation around boundary areas.

Empirically, we experiment with three baselines across four datasets. We first present the unsatisfactory performance on boundary areas when using current point cloud segmentation methods and then show that CBL can assist baseline in achieving promising boundary and overall performance. For example, the proposed CBL helps RandLA-Net surpass current state-of-the-art methods on the Semantic3D dataset and enables a basic ConvNet to achieve leading performance on the S3DIS dataset.

The main contributions of this chapter are as follows:

- We explore the boundary problem in current 3D point cloud segmentation and quantify it with metrics that consider boundary area, *e.g.*, boundary IoU. The results reveal that current methods deliver much worse accuracy in boundary areas than their overall performance.
- We propose a novel Contrastive Boundary Learning (CBL) framework, which improves the feature representation by contrasting the point features across the scene boundaries. It thus improves the segmentation performance around boundary areas and subsequently the overall performance.
- We conduct extensive experiments and show that CBL can bring significant and consistent improvements on boundary area as well as overall performance across all baselines. These empirical results further demonstrate that CBL is effective for improving boundary segmentation performance, and accurate boundary segmentation is important for robust 3D segmentation.

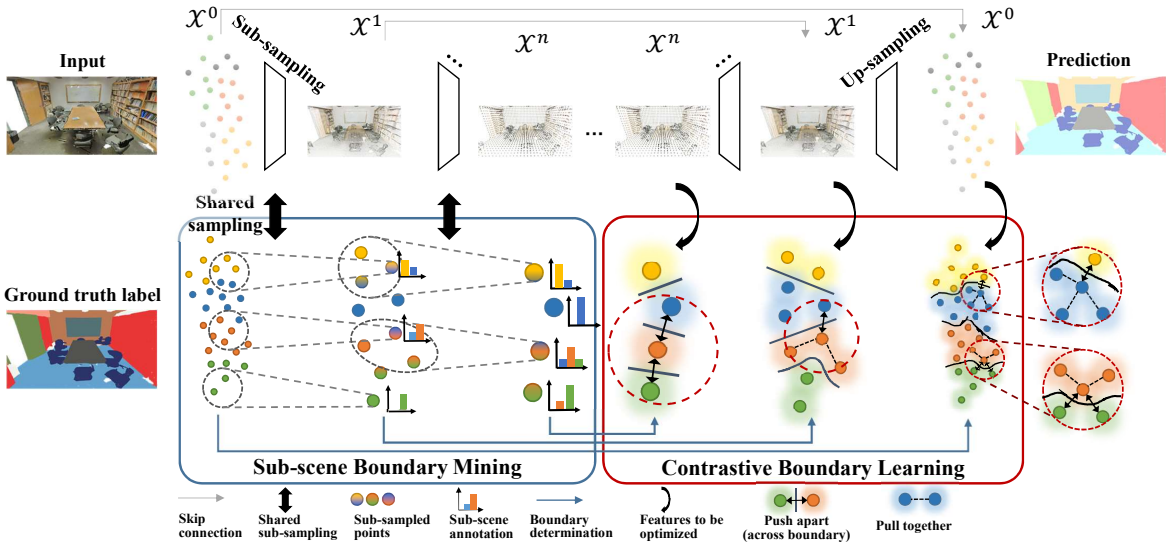


FIGURE 3.2. The detailed illustration of the Contrastive Boundary Learning.

3.2 Related Work

Point cloud segmentation. Point cloud semantic segmentation aims to assign semantic labels to each 3D point. Recent deep learning methods have taken over traditional methods [181, 182] that use hand-crafted features, which can be roughly divided into projection-based and point-based methods.

Projection-based methods project 3D points to grid-like structure, either 2D image [183–186] or 3D voxels [90, 92, 187, 188]. For the 2D image plane, we can make use of existing studies for 2D image processing. However, a complete 3D segmentation generally requires taking multiple viewpoints and re-projection [79, 80], which may result in surface occlusions. For 3D voxels, sparse convolutions [23, 98] are proposed to alleviate the resource consumption in voxel construction, considering the large emptiness in 3D space. In general, the voxel resolution incurs the trade-off between losing detail and being resource-demanding [55]. Point-based network directly operates on 3D points, while a pioneering work in this direction is PointNet [18], which uses point-wise MLPs to process per-point feature. Following this success, recent works adopt an encoder-decoder paradigm [19]. Various local aggregation modules are proposed to examine the local context in point clouds, including 3D convolution [22, 24, 129], attentional operations [26, 109, 189], and graph-based operation [20, 190]. To better process

unstructured point cloud, sub-sampling [111, 114, 191], up-sampling [192, 193], and post-processing modules [194, 195] are also considered to enhance point cloud representation. Despite these developments in different modules, the boundary in point cloud segmentation has rarely been explored.

Boundary in segmentation. Boundary problem has a long history in 2D image processing [178–180, 196], whereas only few works [195, 197] realize the significance of boundary in 3D point cloud segmentation. However, both works involve complex modules for explicit boundary prediction [195, 197] or local aggregation [195]. These operations largely increase the model complexity, yet yield limited performance gain for overall metric. Regarding segmentation performance on boundaries, they also only give qualitative results. In comparison, we present a contrastive learning framework that brings little overhead to the model and can improve upon various baselines with simple adaption. Additionally, we would like to note that, we for the first time, quantify the boundary quality with numeric metric, and demonstrate that boundary problem is indeed widely existing across current methods.

Contrastive learning. Contrastive learning [198–202] has shown promising performance in representation learning for computer vision tasks, ranging from unsupervised settings to supervised settings. In recent works, contrastive learning has also been introduced into 2D segmentation [203, 204] as well as unsupervised representation learning in point cloud processing [145, 146, 205]. Especially, PointContrast [146] conducts point-wise contrastive learning to overcome geometric transformation, such as rigid transformation. P4Contrast [205] suggests a more flexible contrasting strategy to promote multi-modal fusion between geometric and RGB information. In contrast, in our work, we take a supervised setting and demonstrate with CBL that contrastive learning is well-suited for improving segmentation quality on boundary areas. Additionally, unlike the above works that only use points at input point cloud, we utilize the sub-sampled point cloud to examine scene context at multiple scales.

3.3 Segmentation on Boundaries

Since most of the current works focus on the improvement of general metrics, such as mean intersection over union (mIoU), overall accuracy (OA), and mean average precision (mAP), the boundary quality in point cloud segmentation is usually overlooked. Unlike recent boundary-related works [195, 197] that give only qualitative results on boundaries, we are the first to quantify the quality of segmentation on boundaries. Particularly, we introduce a series of metrics for presentation, including mIoU@boundary, mIoU@inner and the boundary IoU (B-IoU) score from 2D instance segmentation tasks [179].

Based on ground-truths data, we consider a point as a boundary point if there exist points that have a different annotated label in its neighborhood. Similarly, for model predictions, we consider a point as a boundary point if there exist nearby points with a different predicted label. More formally, we note the point cloud as $\mathcal{X}$ and the i-th point as x_i, whose local neighborhood is $\mathcal{N}_i = \mathcal{N}(x_i)$, corresponding ground truth label is l_i, and the model predicted label is p_i. We further note the set of boundary points in ground-truth as $\mathcal{B}_l$ and those in predicted segmentation as $\mathcal{B}_p$, thus we have:

$$\begin{aligned} \mathcal{B}_l &= \{x_i \in \mathcal{X} \mid \exists\, x_j \in \mathcal{N}_i,\ l_j \neq l_i\}, \\ \mathcal{B}_p &= \{x_i \in \mathcal{X} \mid \exists\, x_j \in \mathcal{N}_i,\ p_j \neq p_i\}, \end{aligned} \tag{3.1}$$

where we set $\mathcal{N}_i$ to be the radius neighborhood with a radius of 0.1 following the common practice [22, 206].

To examine the boundary segmentation results, an intuitive way is to calculate the mIoU within the boundary area, *i.e.*, mIoU@boundary. To further compare the model performance in boundary and non-boundary (inner) area, we further calculate the mIoU in the inner area, *i.e.* mIoU@inner. Given that mIoU is calculated on the whole point cloud $\mathcal{X}$ as:

$$\text{mIoU}(\mathcal{X}) = \frac{1}{K}\sum_{k=1}^{K}\frac{\sum_{x_i \in \mathcal{X}} \mathbb{1}[p_i = k \wedge l_i = k]}{\sum_{x_j \in \mathcal{X}} \mathbb{1}[p_j = k \vee l_j = k]}, \tag{3.2}$$

where K is the total number of classes and $\mathbb{1}[\cdot]$ represents a boolean function that outputs 1 if the condition within $[\cdot]$ is true and 0 otherwise. We have the mIoU@boundary and

mIoU@inner defined as:

$$\begin{aligned} \text{mIoU@boundary} &= \text{mIoU}(\mathcal{B}_l), \\ \text{mIoU@inner} &= \text{mIoU}(\mathcal{X} - \mathcal{B}_l), \end{aligned} \tag{3.3}$$

where $\mathcal{X} - \mathcal{B}_l$ is the set of points in inner area.

However, the mIoU@boundary and mIoU@inner do not consider the false boundary in model predicted segmentation. Inspired by boundary IoU [179] for 2D instance segmentation, for better evaluation, we consider the alignment between boundary in segmentation predictions and boundary in ground truth data. It thus leads to the following B-IoU for evaluation:

$$\text{B-IoU} = \frac{|\mathcal{B}_l \cap \mathcal{B}_p|}{|\mathcal{B}_l \cup \mathcal{B}_p|}. \tag{3.4}$$

3.4 Method

In this section, we present our contrastive boundary learning (CBL) framework, shown in Fig. 3.2. It imposes contrastive learning to enhance the feature discrimination across boundaries. Then, to deeply augment the model performance on boundaries, we enable the CBL in sub-sampled point clouds, *i.e.*, sub-scene, through the sub-scene boundary mining.

Contrastive Boundary Learning. We follow the widely used InfoNCE loss [198] and its generalization [199, 207] to define the contrastive optimization goal on boundary points. In particular, for a boundary point $x_i \in \mathcal{B}_l$, we encourage learned representations more similar to its neighbor points from the same category and more distinguished from other neighbor points from different categories, *i.e.*,

$$L_{CBL} = \frac{-1}{|B_l|} \sum_{x_i \in B_l} \log \frac{\sum\limits_{x_j \in \mathcal{N}_i \wedge l_j = l_i} \exp(-d(f_i, f_j)/\tau)}{\sum\limits_{x_k \in \mathcal{N}_i} \exp(-d(f_i, f_k)/\tau)}, \tag{3.5}$$

where f_i is the feature of x_i, $d(\cdot, \cdot)$ is a distance measurement and τ is the temperature in contrastive learning. The contrastive learning described by Eq. (3.5) focuses on boundary points only (the dashed circles in red in Fig. 3.2). First, we consider all the boundary points

$\mathcal{B}_l$ from ground-truth data as defined in Eq. (3.1). Then, for each point $x_i \in \mathcal{B}_l$, we restrict the sampling of its positive and negative points to be within its local neighborhood $\mathcal{N}_i$. With such strong spatial restriction, we obtain positive pairs for x_i as $\{x_j \in \mathcal{N}_i \wedge l_j = l_i\}$, and other neighboring points, *i.e.* $\{x_j \in \mathcal{N}_i \wedge l_j \neq l_i\}$, are negative pairs. Therefore, the contrastive learning enhances the feature discrimination across scene boundaries, which is important for improving segmentation on boundary areas.

Sub-scene Boundary Mining. To better explore scene boundaries, we examine the boundaries in sub-sampled point clouds at multiple scales, which enables the contrastive boundary learning on different sub-sampling stages of a backbone model. Collecting boundary points from the input point cloud is straightforward with the ground truth label. However, after sub-sampling, it is difficult to obtain a proper definition of boundary point set following Eq. (3.1), due to the undefined label for sub-sampled points [208]. Therefore, to enable CBL in sub-sampled point cloud, we propose the sub-scene boundary mining that determines the set of ground-truth boundary points in each sub-sampling stage. Specifically, we use superscripts to denote stage. At the sub-sampling stage n, we represent its sub-sampled point cloud as $\mathcal{X}^n$. For input point cloud, we have $\mathcal{X}^0 = \mathcal{X}$. When collecting a set of boundary points $\mathcal{B}_l^n \in \mathcal{X}^n$ in stage n, it is required to determine the label l_i^n of a sub-sampled point $x_i^n \in \mathcal{X}^n$, *i.e.*, the sub-scene annotation. As each sub-sampled point $x_i^n \in \mathcal{X}^n$ is aggregated from a group of points in its previous point cloud $\mathcal{X}^{n-1}$; we thus utilize the sub-sampling procedure to determine the label iteratively. We take l_i^0 to be the one-hot label of ground truth label l_i for point $x_i^0 = x_i$, and have the following:

$$l_i^n = \text{AVG}(\{l_j^{n-1} | x_j^{n-1} \in \mathcal{N}^{n-1}(x_i^n)\}), \tag{3.6}$$

where $\mathcal{N}^{n-1}(x_i^n)$ denotes the local neighbors of x_i^n in previous stage (the dashed circles in grey in Fig. 3.2), *i.e.*, the group of points aggregated from $\mathcal{X}^{n-1}$ to be represented by the single point $x_i^n \in \mathcal{X}^n$ after sub-sampling procedure, and AVG is the average-pooling.

With Eq. (3.6) and ground-truth labels, we can iteratively obtain the sub-scene annotation l_i^n as a distribution, whose k-th location describes the proportion of k-th class in its corresponding

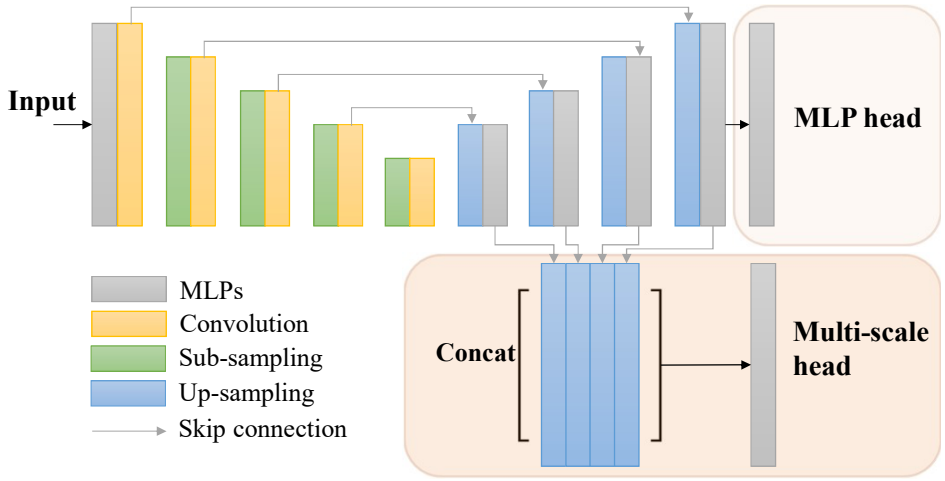


FIGURE 3.3. The architecture of the 3D ConvNet model, which follows the widely adopted encoder-decoder paradigm, with an optional multi-scale prediction head. More details are provided in the appendix.

group of points in the input point cloud. To determine the set of boundary points in sub-sampled point cloud $\mathcal{X}^n$, we simply take $\arg\max l_i^n$ to allow the evaluation of boundary point in Eq. (3.1)[1], and use the feature of sub-sampled point for the contrastive boundary optimization in Eq. (3.5). Finally, with sub-scene boundary mining, we have CBL applied at all stages and the final loss is

$$L = L_{\text{cross entropy}} + \lambda \sum_n L_{CBL}^n, \tag{3.7}$$

where L_{CBL}^n is the CBL loss at stage n and λ is the loss weight.

3.5 Implementation Details and Baselines

As 3D ConvNet has been a popular backbone model for point cloud processing, to present a generalized implementation, we illustrate with a ConvNet baseline (Fig. 3.3) as a case study for applying CBL in point cloud processing. Following [125, 126], we build the ConvNet

[1]We choose $\arg\max$ for its simplicity and non-parametric nature. We provide more analysis on this choice in the appendix.

with convolution in 3D continuous space:

$$f_i = (h \circ g)(x_i) = \sum_{x_j \in \mathcal{N}_i} g(x_i - x_j)h(x_j), \tag{3.8}$$

where $\circ$ denotes convolution operator and the continuous kernel $g(\cdot)$ is approximated by one-layer MLP and set $h(x_j) = f_j$ to simply use the feature of point x_j. We note that the 3D convolution in Eq. (3.8) is purely based on spatial location between the center point and its neighbors, compared to other advanced local aggregation modules that utilize the local context [26, 109].

To better utilize the boundary features optimized by CBL at multiple scales, we use a multi-scale head for prediction, which simply concatenates the point feature from each sub-sampled point cloud into the last output layer. As we would show in the ablation study (Sec. 3.6.3), such concatenation across multiple scales fails without the CBL. Note that CBL can be married to any other multi-stage backbone. Specifically, we also apply the CBL to two other popular baselines: the RandLA-Net [109] and CloserLook3D [206], to demonstrate the generalizability. RandLA-Net leverages random sampling and attentive local aggregation to handle the large-scale scene with fast processing; CloserLook3D proposes a parameter-free PosPool module that largely reduces model parameters and resources consumption, while achieving comparable performance against other methods with parametric aggregation module, such as KPConv [22]. Together with the ConvNet baseline, our experiments cover the backbone with most of the typical local aggregation methods for point cloud, ranging from convolution, attentional operation, to parameter-free operation. For training, we follow the setup of baseline and set the loss weight $\lambda = 0.1$. More details will be provided in the appendix.

3.6 Experiments

We first present the boundary problem with experiments. We then evaluate the benefits of the proposed CBL on multiple large-scale point cloud segmentation datasets, including in-door scenes (S3DIS [44], ScanNet [43]) and out-door scenes (Semantic3D [68], NPM3D [70]).

methods	mIoU overall	mIoU @boundary	mIoU @inner	B-IoU
pointnet [18]	41.1	30.2	53.4	35.6
KPConv [22]	67.3	50.5	71.1	58.9
JSE-Net [195]*	67.7	50.5	71.4	60.9
RandLA-Net [109]	62.6	44.1	65.8	45.4
CloserLook3D [206]	66.9	50.0	70.7	59.2
ConvNet	67.4	50.1	71.2	59.6
RandLA-Net + CBL	65.3	47.4	67.2	49.9
	+2.7	**+3.3**	**+1.4**	**+4.5**
CloserLook3D + CBL	67.5	50.6	71.0	60.4
	+0.6	**+0.6**	**+0.3**	**+1.2**
ConvNet + CBL	69.4	52.6	73.1	61.5
	+2.0	**+2.5**	**+1.9**	**+1.9**

TABLE 3.1. The results are obtained on the S3DIS datasets testset Area 5, following the instruction of the officially released code of each method. Method with * also consider boundaries.

3.6.1 The Boundary Problem in Experiment

We experimentally compare the score given by mIoU, mIoU@boundary, mIoU@inner as well as the B-IoU. As shown in Tab. 3.1, for recent 3D point cloud segmentation methods, the mIoU@boundary is much lower than the mIoU@inner. With the overall performance sitting between these two scores, it suggests that it is the boundary area that degenerates the overall segmentation performance. Similarly, B-IoU also agrees with the mIoU@boundary by giving a score that is far lagged behind the general performance of mIoU score. Hence, such observation indicates the unsatisfied segmentation quality on boundary areas. While with the proposed CBL, the improvement on both mIoU@boundary and B-IoU is larger than the improvement on overall mIoU as well as the mIoU@inner, across all three baselines. Due to the limited space, we provide more thorough studies in presenting the boundary problem in the appendix.

3.6.2 Performance Comparison

S3DIS Indoor Scene Segmentation. S3DIS [44] is a challenging point cloud dataset of indoor scenes. It contains 3D RGB point clouds of 6 indoor areas covering 272 rooms. Each

methods	mIoU	OA	mACC	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
PointNet [18]	41.1	-	49.0	88.8	97.3	69.8	0.1	3.9	46.3	10.8	59.0	52.6	5.9	40.3	26.4	33.2
SegCloud [90]	48.9	-	57.4	90.1	96.1	69.9	0.0	18.4	38.4	23.1	70.4	75.9	40.9	58.4	13.0	41.6
PointCNN [24]	57.3	85.9	63.9	92.3	98.2	79.4	0.0	17.6	22.8	62.1	74.4	80.6	31.7	66.7	62.1	56.7
SPGraph [190]	58.0	86.4	66.5	89.4	96.9	78.1	0.0	**42.8**	48.9	61.6	84.7	75.4	69.8	52.6	2.1	52.2
PCT [189]	61.3	-	67.7	92.5	98.4	80.6	0.0	19.4	61.6	48.0	76.6	85.2	46.2	67.7	67.9	52.3
HPEIN [110]	61.9	87.2	68.3	91.5	98.2	81.4	0.0	23.3	**65.3**	40.0	75.5	87.7	58.5	67.8	65.6	49.4
MinkowskiNet [92]	65.4	-	71.7	91.8	**98.7**	**86.2**	0.0	34.1	48.9	62.4	81.6	89.8	47.2	74.9	**74.4**	58.6
KPConv [22]	67.1	-	72.8	92.8	97.3	82.4	0.0	23.9	58.0	69.0	81.5	**91.0**	75.4	75.3	66.7	58.9
JSENet [195]*	67.7	-	-	93.8	97.0	83.0	0.0	23.2	61.3	71.6	89.9	79.8	75.6	72.3	72.7	60.4
CGA-Net [194]	68.6	-	-	94.5	98.3	83.0	0.0	25.3	59.6	71.0	**92.2**	82.6	76.4	**77.7**	69.5	61.5
RandLA-Net [109]	62.4	87.2	71.4	91.1	95.6	80.2	0.0	24.7	62.3	47.7	76.2	83.7	60.2	71.1	65.7	53.8
+ CBL	65.3	87.5	74.5	92.2	97.7	81.0	0.0	36.8	61.0	39.4	78.1	88.1	**81.4**	71.5	68.7	52.6
CloserLook3D [206]	66.9	90.0	72.1	94.8	98.4	82.5	0.0	25.5	51.3	70.9	92.1	81.9	76.7	70.1	64.5	61.2
+ CBL	67.5	90.2	72.7	**94.9**	98.4	83.1	0.0	27.3	55.0	71.2	91.9	82.9	75.9	71.3	63.5	60.4
ConvNet	67.4	90.1	72.9	94.1	98.1	83.1	0.0	24.9	53.5	**73.0**	91.7	82.3	76.5	72.3	66.9	60.8
+ CBL	**69.4**	**90.6**	**75.2**	93.9	98.4	84.2	0.0	37.0	57.7	71.9	91.7	81.8	77.8	75.6	69.1	**62.9**

TABLE 3.2. Quantitative results on S3DIS Area 5 dataset [44], showing the mean IoU (mIoU) overall accuracy (OA) and the mean accuracy (mACC). Method with * also consider boundaries in their design.

point is annotated with one of the 13 semantic categories, e.g., ceiling, floor, clutter. As shown in Tab. 3.2, our methods consistently improve across all three baselines, showing to be effective with different local aggregation modules. Notably, the improvements are much more significant in classes, such as column (+13 compared to ConvNet baseline), than in other classes with large areas, such as wall and ceiling. Such observation shows our effectiveness on boundary areas; and with the consistent improvement across different classes, it also suggests that the CBL is NOT trading off between scenes of major and minor classes, but is indeed separating them more clearly. With the benefit of a cleaner boundary, the ConvNet finally achieves a leading performance of 69.4 in mIoU.

We further demonstrate qualitatively in Fig. 3.4 that, the CBL effectively improves the overall performance by improving segmentation on boundary areas. Compared with JSENet [195] that also considers boundaries, we demonstrate our superiority by obtaining a much larger relative improvement to our baselines than that made by JSENet on its baseline, *i.e.*, KPConv [22], especially in classes that boundaries are important, *e.g.*, column, window, sofa, bookcase and clutter, as well as the overall performance. To avoid overfitting on S3DIS Area 5, we further conduct the 6-fold cross-validation, with the result reported in Tab. 3.3. A large improvement is also shown in column (+9.5), and consistent improvement is made across all classes except one (-0.2). Therefore, the proposed CBL can be indeed regarded as a general and effective method, achieving 73.1 in mIoU with a common ConvNet baseline.

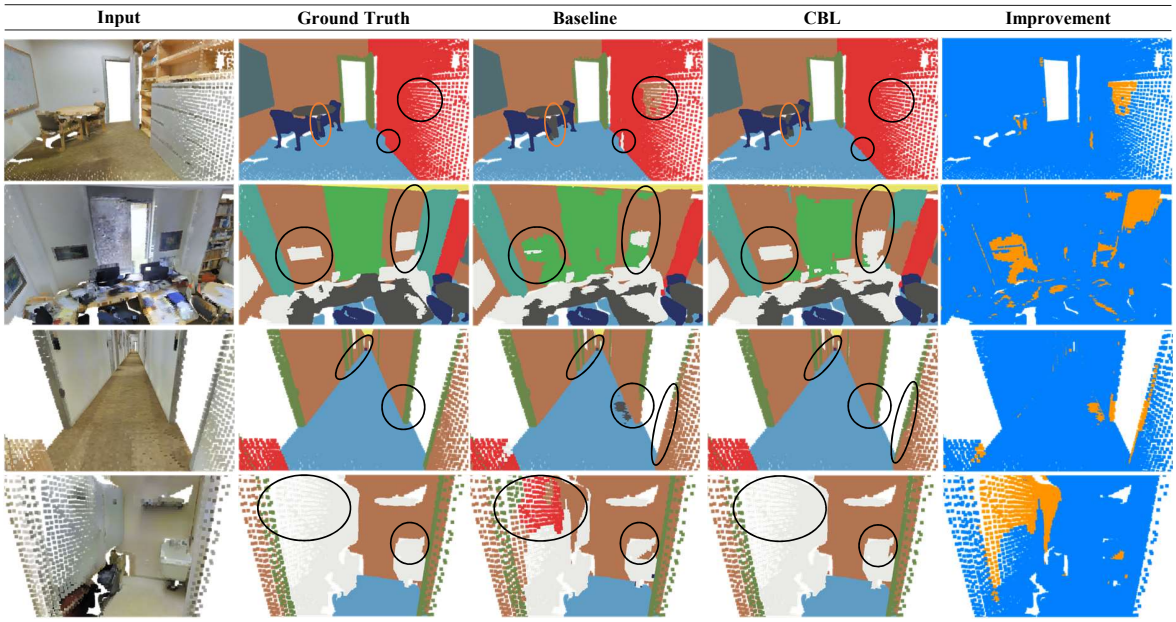


FIGURE 3.4. We compare the results of ConvNet baseline with CBL on several different scenes and show that the improvements are from boundaries. In offices (top 2), CBL can effectively improve the results on boundary areas, especially in a cluttered one (2nd row). In the last two rows (hallway and others), CBL avoids unnecessary boundaries, and repairs the missing boundary between walls and doors/objects at the right place. The visualization is done on S3DIS testset Area 5.

methods	mIoU	OA	mACC	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
PointNet [18]	47.6	78.6	66.2	88.0	88.7	69.3	42.4	23.1	47.5	51.6	54.1	42.0	9.6	38.2	29.4	35.2
RSNet [209]	56.5	-	66.5	92.5	92.8	78.6	32.8	34.4	51.6	68.1	59.7	60.1	16.4	50.2	44.9	52.0
SPG [190]	62.1	86.4	73.0	89.9	95.1	76.4	62.8	47.1	55.3	68.4	73.5	69.2	63.2	45.9	8.7	52.9
PointCNN [24]	65.4	88.1	75.6	**94.8**	**97.3**	75.8	63.3	51.7	58.4	57.2	71.6	69.1	39.1	61.2	52.2	58.6
PointWeb [144]	66.7	87.3	76.2	93.5	94.2	80.8	52.4	41.3	64.9	68.1	71.4	67.1	50.3	62.7	62.2	58.5
ShellNet [137]	66.8	87.1	-	90.2	93.6	79.9	60.4	44.1	64.9	52.9	71.6	**84.7**	53.8	64.6	48.6	59.4
RandLA-Net [109]	70.0	88.0	82.0	93.1	96.1	80.6	62.4	48.0	64.4	69.4	69.4	76.4	60.0	64.2	65.9	60.1
KPConv [22]	70.6	-	79.1	93.6	92.4	83.1	63.9	54.3	66.1	76.6	57.8	64.0	69.3	**74.9**	61.3	60.3
SCF-Net [210]	71.6	88.4	82.7	93.3	96.4	80.9	**64.9**	47.4	64.5	70.1	71.4	81.6	67.2	64.4	**67.5**	60.9
BAAF [192]	72.2	88.9	**83.1**	93.3	96.8	81.6	61.9	49.5	65.4	73.3	72.0	83.7	67.5	64.3	67.0	62.4
ConvNet	69.7	88.6	76.8	93.8	91.9	84.2	46.3	52.1	66.7	**78.5**	75.2	72.8	70.1	71.7	57.1	61.3
+ CBL	**73.1**	**89.6**	79.4	94.1	94.2	**85.5**	50.4	**58.8**	**70.3**	78.3	**75.7**	75.0	**71.8**	74.0	60.0	**62.4**

TABLE 3.3. Quantitative results on S3DIS [44] with 6-fold cross validation.

Semantic3D Outdoor Scene Segmentation. In addition to improvement on S3DIS [44], we demonstrate the generalizability across different types of scenes by evaluating CBL on point cloud collected at the outdoor environment, the Semantic3D [68] dataset. It is a large-scale dataset comprising over 4 billion points and provides 15 large point clouds for training, with each point annotated to one of the 8 classes, *e.g.*, cars, buildings. We use the reduced-8 benchmark and present the quantitative results in Tab. 3.4. We evaluate with both ConvNet and RandLA-Net [109] as baselines and observe consistent improvements.

methods	mIoU (%)	OA (%)	man-made.	natural.	high veg.	low veg.	buildings	hard scape	scanning art.	cars
SnapNet [79]	59.1	88.6	82.0	77.3	79.7	22.9	91.1	18.4	37.3	64.4
SEGCloud [90]	61.3	88.1	83.9	66.0	86.0	40.5	91.1	30.9	27.5	64.3
SPG [190]	73.2	94.0	97.4	92.6	87.9	44.0	83.2	31.0	63.5	76.2
RGNet [211]	74.7	94.5	**97.5**	**93.0**	**88.1**	48.1	94.6	36.2	72.0	68.0
KPConv [22]	74.6	92.9	90.9	82.2	84.2	47.9	94.9	40.0	**77.3**	**79.7**
RFCR [208]	77.8	94.3	94.2	89.1	85.7	54.4	95.0	43.8	76.2	83.7
SCF-Net [210]	77.6	94.7	97.1	91.8	86.3	51.2	95.3	50.5	67.9	80.7
ConvNet	72.8	92.6	92.2	79.9	84.4	41.3	95.2	41.2	62.6	85.6
+ CBL	75.0	94.0	96.2	90.1	84.0	47.5	94.7	36.0	64.8	86.3
RandLA-Net [109]	77.4	94.8	95.6	91.4	86.6	51.5	95.7	51.5	69.8	76.8
+ CBL	**78.4**	**95.0**	95.3	91.3	87.9	**55.6**	**96.3**	**56.2**	65.9	78.2

TABLE 3.4. Quantitative results on Semantic3D reduced-8 benchmark [68]. The metrics shown the mean IoU (mIoU) and overall accuracy (OA) obtained from benchmark site with only the recent published works included.

Especially, RandLA-Net has achieved state-of-the-art performance on multiple outdoor datasets and the improvement made on it can better demonstrate the effectiveness of our CBL. Notably, significant improvement is made in the high vegetation and low vegetation class, which are two classes that confuse most of the other methods. It is because the high/low vegetation usually co-exists at a near spatial distance and has a similar appearance, *e.g.*, trees surrounded by bushes/grass, which makes the separation of these two scenes challenging. The large improvement in both of these two classes demonstrates the effective improvement on scene boundaries. Lastly, with CBL, RandLA-Net obtains a leading performance of 78.4 in mIoU.

Further experiments on NPM3D and ScanNet. To further demonstrate the generalization of the proposed CBL, we report on another two popular dataset, the ScanNet [43] (indoor scene) and NPM3D [70] (outdoor scene). As shown in Tab. 3.5 and Tab. 3.6, our method achieves reasonable results and consistent improvement over the baseline. It thus shows that CBL is robust to different baselines, datasets, and types of scenes. Detailed results are available in the appendix.

3.6.3 Ablation Studies

We conduct ablation studies on the ScanNet validation set to evaluate the effectiveness of different components in the proposed CBL scheme.

methods	modality	mIoU (%)
DCM-Net [212]	3D + Mesh	65.8
VMNet [213]		74.6
SparseConvNet [23]	3D (voxel)	72.5
MinkowskiNet [92]		73.6
O-CNN [187]		76.2
OccuSeg [188]		76.4
Mix3D [214]		78.1
BA-GEM [197]*	3D (point)	63.5
PointConv [129]		66.6
PointASNL [191]		66.6
KP-Conv [22]		68.4
FusionNet [215]		68.8
JSENet [195]*		69.9
RFCR [208]		70.2
ConvNet	3D (point)	69.1
+ CBL		70.5

TABLE 3.5. Quantitative results on ScanNet [43] benchmark. Performance is taken from the official benchmark site by the time of submission. Methods with * also consider boundaries.

methods	mIoU (%)
HDGCN [216]	68.3
ConvPoint [125]	75.9
RandLANet [109]	78.5
KP-Conv [22]	82.0
FKAConv [217]	82.7
PyramidPoint [193]	82.9
ConvNet	76.2
+ CBL	78.6

TABLE 3.6. Quantitative results on Paris-Lille-3D of NPM3D [70] benchmark, results obtained from online benchmark site by the time of submission.

The Effectiveness of CBL. As shown in Tab. 3.7, the direct application of CBL on the input point cloud (without sub-scene boundary mining) can improve the performance, which demonstrates that boundary areas are worth more attention. By introducing sub-scene boundary mining, a more significant improvement is gained, as boundaries at multiple scales are identified and optimized in the CBL.

	CBL @input	CBL @sub-scenes	mIoU(%)	OA(%)
ConvNet			69.71 -	88.97 -
	✓		70.05 (+0.34)	89.01 (+0.04)
	✓	✓	70.98 (+1.27)	89.31 (+0.34)
ConvNet (multiscale head)			69.83 (+0.12)	88.88 (−0.09)
	✓	✓	71.33 (+1.62)	89.40 (+0.43)

TABLE 3.7. Results on validation set of ScanNet [43]. The CBL @input refers to only conduct contrastive boundary learning on the input point cloud (with point feature extracted from last upsampled stage), and @sub-scene refers to the CBL with sub-scene boundary mining. Default settings are marked in gray and relative improvements are also noted.

The Effect of Multi-scale Head. Comparing the ConvNet baseline with and without the multi-scale head, we find that a direct application of multi-scale head can even hurt the performance (-0.09 in OA). It shows that a direct concatenation across multiple scales can not bring much benefit. In contrast, with multi-scale head, ConvNet with CBL is further boosted to gain a larger improvement in both mIoU and OA. It shows that the main improvement is originated from the more discriminative features learned by CBL at different sub-sampled point clouds.

3.7 Summary

In this chapter, we comprehensively analyze the segmentation performance on scene boundaries for the current point cloud segmentation methods. We show that the current segmentation accuracy on boundaries is unsatisfactory and quantitatively present the boundary problem with metrics, including mIoU@boundary and B-IoU. We further propose Contrastive Boundary Learning (CBL) to explicitly optimize the feature on boundaries and improve the model performance on boundaries. The leading performance and consistent improvement across various baselines and datasets demonstrate the effectiveness of CBL and the importance of scene boundaries in 3D point cloud segmentation.

Limitation and future work. One of our limitation is that we mainly concentrate on the scene boundaries while ignoring the broad inner areas. Therefore, in the future, we would

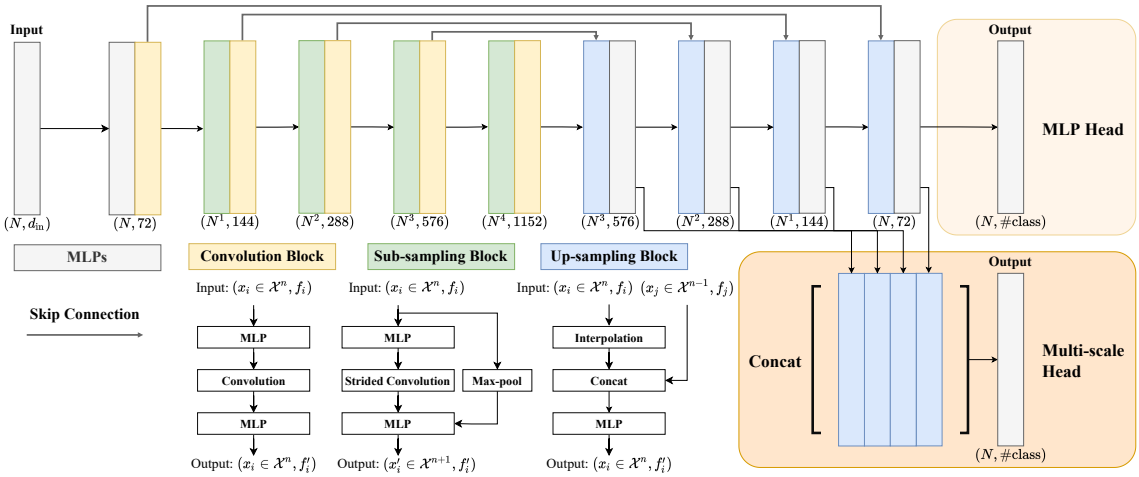


FIGURE 3.5. The detail architecture of ConvNet baseline.

like to further explore the role of boundary in point cloud segmentation and its relation with inner areas.

3.8 Appendix

In this supplementary material, we provide more details regarding baseline architecture (Sec. 3.8.1), the boundary problem Sec. 3.8.2, visualization results (Sec. 3.8.3), the training setup (Sec. 3.8.4), the effect of temperature (Sec. 3.8.5), the effect of design regarding sub-scene annotation (Sec. 3.8.6), and experiment results (Sec. 3.8.7).

Especially, CBL achieves a new stat-of-the-art on S3DIS with the newly released transformer model (Tab. 3.14).

3.8.1 Architecture of Baseline

We show the specific architecture of our ConvNet baseline in Fig. 3.5. With a consistent notation, $\mathcal{X}^n$ is the point cloud in sub-sampling stage n, f_i is the feature of point x_i, and $N^n = |\mathcal{X}^n|$ with $N = N^0$. We use the multi-scale head on all baselines when adapting the CBL.

3.8.2 Further Analysis on Boundary Problem

We further account for the type of areas and class-specific analysis for better exploring the boundary problem. Specifically, we provide per-class IoU score that is separately calculated on boundary area $\mathcal{B}_l$ and inner area $\mathcal{X} - \mathcal{B}_l$.

As shown in Tab. 3.9, we evaluate for all three baselines with and without the proposed CBL. We notice that, large improvements are made on small objects, *e.g.* column, which aligns with the observation in Tab. 3.2 in main chapter. We would like to add that, despite that CBL focuses only on boundaries, improvements are also made on inner area. We hypothesize the reason might be that the false boundary in model predicted segmentation is restrained, as features in inner area implicitly becomes more similar when the features across boundaries are optimized to be more distinctive by the CBL.

Moreover, for all three baselines, the improvement on boundary area is much more than that made on inner area, which is summarized in Tab. 3.8.

Therefore, with metrics separately calculated on boundary and inner area, we clearly see that the improvement brought by CBL is mainly from the boundary areas. Such observation further emphasizes the importance of clear scene boundaries in point cloud segmentation task.

3.8.3 More Visualizations

We provide more qualitative results as a support for the improvement made by CBL on boundaries. The visualization results include various scenes, including rooms (Fig. 3.7), cluttered space (Fig. 3.8), hallways (Fig. 3.9), and offices (Fig. 3.10). For each scene, we further attempt to visualize the features discrimination between center points and their corresponding neighbors and the results are presented in the every second row. Specifically, we calculate the normalized feature distance between the point feature f_i and features of its neighboring points $\{f_j \mid x_j \in \mathcal{N}_i\}$. We then take the mean distance for visualization.

According to the presented figures, it shows that the CBL significantly enhances the feature distances around the scene boundaries and improves the baseline to obtain a more detailed

baselines (+ CBL)	mIoU		OA		mACC	
	boundary	inner	boundary	inner	boundary	inner
RandLA-Net [109]	+3.3	+1.4	+4.1	-0.3	+3.4	+2.4
CloserLook3D [206]	+0.6	+0.2	+0.1	+0.2	+0.7	+0.4
ConvNet	+2.5	+2.0	+1.0	+0.7	+3.2	+2.8

TABLE 3.8. The improvement brought by CBL on different baselines and types of area (boundary / inner area).

methods	mIoU	OA	mACC	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
RandLA-Net [109]	44.1	67.1	59.1	65.5	69.4	52.2	0.0	21.4	28.6	55.0	55.0	56.0	41.1	41.2	45.8	42.1
+ CBL	47.4	71.2	62.5	78.2	85.9	56.0	0.0	30.3	25.7	42.6	58.4	60.9	50.0	42.5	52.2	44.2
CloserLook3D [206]	50.0	76.6	58.5	80.7	88.6	63.9	0.0	21.1	15.6	57.5	73.3	64.7	52.2	43.1	37.2	52.6
+ CBL	50.6	76.7	59.2	80.9	88.6	64.6	0.0	26.5	15.6	55.9	73.0	65.0	50.4	47.6	38.4	51.2
ConvNet	50.1	76.5	58.3	80.4	88.3	63.5	0.0	26.5	15.2	58.3	72.1	63.4	52.3	40.8	38.7	52.2
+ CBL	52.6	77.5	61.5	80.5	88.8	65.7	0.0	32.5	20.9	61.8	71.7	62.4	52.5	46.7	47.4	52.5

(A) The full metrics calculated on boundary points from ground truth (*i.e.*, $\mathcal{B}_l$) only.

methods	mIoU	OA	mACC	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
RandLA-Net [109]	65.8	89.6	73.0	93.3	98.6	84.6	0.0	25.9	65.7	46.5	81.1	88.9	65.4	75.5	71.9	58.2
+ CBL	67.2	89.3	75.4	93.0	99.1	84.6	0.0	37.3	64.1	39.4	82.7	91.5	79.3	75.9	73.9	56.0
CloserLook3D [206]	70.7	92.2	75.2	96.4	99.9	86.5	0.0	25.9	55.1	76.5	95.9	87.1	81.9	75.1	72.5	66.2
+ CBL	70.9	92.4	75.6	96.5	99.9	86.9	0.0	27.0	59.3	78.1	95.7	87.7	80.8	75.4	69.4	65.6
ConvNet	71.2	92.1	75.5	95.0	99.8	85.9	0.0	34.6	56.0	82.7	95.4	87.4	81.3	73.8	68.4	65.7
+ CBL	73.2	92.8	78.3	95.3	99.9	88.0	0.0	38.4	62.2	76.4	95.9	87.5	82.7	81.2	75.2	68.6

(B) The full metrics calculated on inner points from ground truth (*i.e.*, $\mathcal{X} - \mathcal{B}_p$) only.

TABLE 3.9. The consistent improvement CBL brought on baselines, separately calculated in boundary area (a) and inner area (b).

temperature	mIoU	OA	mACC
0.3	70.67	89.16	77.91
0.5	70.98	89.31	78.27
1	71.33	89.40	78.69
2	70.73	89.10	77.98
10	70.03	88.97	77.58

TABLE 3.10. The effect of temperature on CBL.

and cleaner boundary in prediction for different type of scenes. The visualization is done on S3DIS testset Area 5.

	mIoU (%)	Ground	Building	Pole	Bollard	Trash can	Barrier	Pedestrian	Car	Natural
HDGCN [216]	68.3	99.4	93.0	67.7	75.7	25.7	44.7	37.1	81.9	89.6
ConvPoint [125]	75.9	99.5	95.1	71.6	88.7	46.7	52.9	53.5	89.4	85.4
RandLANet [109]	78.5	99.5	97.0	71.0	86.7	50.5	65.5	49.1	95.3	91.7
KP-Conv [22]	82.0	99.5	94.0	71.3	83.1	78.7	47.7	78.2	94.4	91.4
FKAConv [217]	82.7	99.6	98.1	77.2	91.1	64.7	66.5	58.1	95.6	93.9
PyramidPoint [193]	82.9	99.6	97.1	74.6	84.3	56.0	65.9	79.1	95.1	93.9
ConvNet	76.2	99.5	96.3	68.5	67.4	41.4	41.5	80.6	96.3	94.1
+ CBL	78.6	99.5	96.7	72.1	72.6	46.2	60.4	70.1	97.2	93.2

TABLE 3.11. Quantitative results on Paris-Lille-3D of NPM3D [70] benchmark, results obtained from online benchmark site by the time of submission.

3.8.4 Training Setup in Details

For the RandLA-Net [109] and CloserLook3D [206] baselines, we follow their instructions of released code for training and evaluation, which are here (RandLA-Net) and here (CloserLook3D), respectively. Especially, in CloserLook3D [206], there are two non-parametric module, we use the one with sin/cos spatial embedding.

For the ConvNet baseline, we use the SGD optimizer to train for 600 epoch, with a weight decay of 0.001. We set the initial learning rate to 0.01 and use a momentum of 0.98 with a decay rate of $0.1^{1/200}$. It roughly takes 24 hours to train on 4 Nividia v100 GPUs, and we do not observe obvious increase in training time after applying the CBL.

3.8.5 Effect of Temperature in CBL

We conduct empirical study on ScanNet [43] validation set to analyze the effect of temperature τ in the CBL (Eq. (3.5)). We use the ConvNet baseline and train for 600 epoch on training set. As shown in Tab. 3.10, we find that the proper temperature for CBL is within $(0.5, 2)$, and we set the temperature to $\tau = 1$ by default.

3.8.6 Effect of Design of Sub-scene Annotation

While the sub-scene annotation is a distribution, we only use the simple $\arg\max$ when evaluating the boundary points. Therefore, it raises two particular question: 1) is it necessary

to maintain the distribution? 2) is there any better way in utilizing the sub-scene annotation than the $\arg\max$?

In this section, we explore other alternatives and answer to this two questions with a particular focus of how they affect the model performance on boundaries.

Necessities of maintaining distribution. There are two main reasons to leverage the average pooling on labels and maintain the distribution. First, current methods may not preserve the original input points after sub-sampling, *e.g*. grid sub-sampling in KPConv [22]. Therefore, the original label of a sub-sampled point is not presented and the sub-scene annotation is thus demanded. Although we may use the label of the nearest point for approximation, Tab. 3.12 shows that CBL (nearest) is sub-optimal. Second, despite that we only use the "argmax" result of the sub-scene annotation, maintaining distribution still preserves more information than just maintaining "argmax" result. As "argmax" discards the minor classes during sampling, such elimination of minority may further accumulate through more sub-sampling stages and leads to imprecise boundary, as depicted in Fig. 3.6. Experimentally, in Tab. 3.12, though CBL (argmax) improves boundary (B-IoU), it compromises overall performance.

Better treatment than Argmax. While "argmax" is straight forward, it introduces the problem of "label-flipping" when the distribution of sub-scene annotation is close to a uniform distribution, *i.e*., when the number of points of different classes are roughly the same.

To avoid this, we leverage the KL divergence as a measure of the semantic distance among sub-scene annotations. We then threshold on the KL-distance to determine if two sub-scene annotations belong to the same semantic class or not, which further enables us to determine the boundary points in sub-sampled point cloud. Specifically, we set the threshold to 0.5 and CBL (kl) can be bring a small improvement on overall performance, and a slightly larger boost on boundary performance, as in Tab. 3.12. Yet, as "thresholding KL distance" introduces extra hyper-parameters and complexity, we opt for "argmax" for simplicity in the main chapter.

Summary. Therefore, we summarize the reason for designing the sub-scene annotation as a distribution as it can preserve much more information and can be extended to a more robust boundary determination using KL-distance.

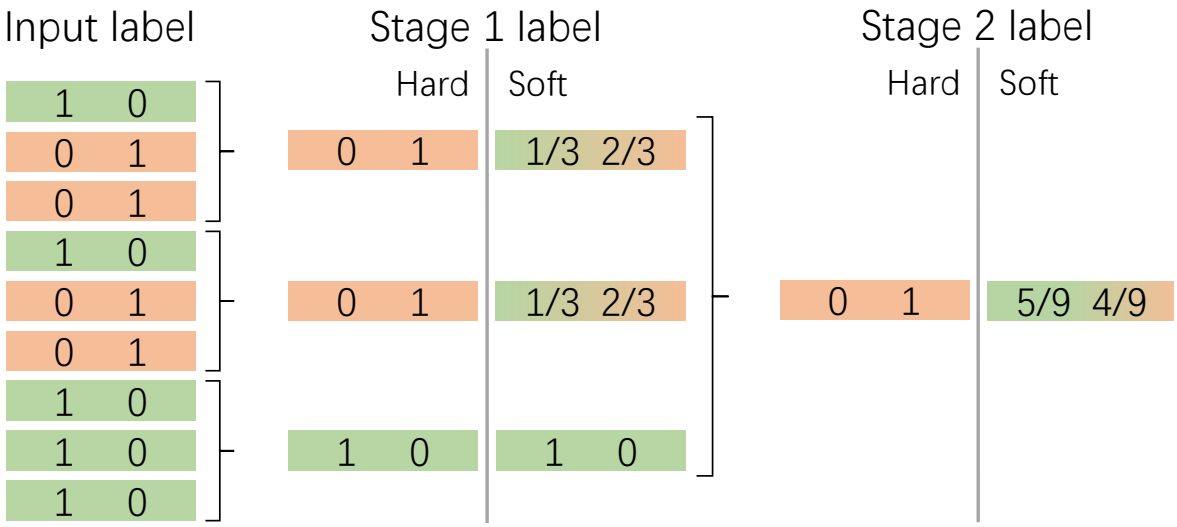


FIGURE 3.6. With every 3 points being sub-sampled into 1 in each stage, tracking distribution (soft label) describes original input faithfully, but hard label fails due to accumulated errors.

methods	mIoU overall	mIoU @boundary	mIoU @inner	B-IoU
ConvNet	67.4	50.1	71.2	59.6
ConvNet + CBL	69.4	52.6	73.1	61.5
ConvNet + CBL (nearest)	68.3	52.1	71.8	60.9
ConvNet + CBL (argmax)	66.8	50.6	70.4	60.6
ConvNet + CBL (kl)	69.5	52.5	73.2	62.0

TABLE 3.12. Same setting as in Tab. 3.1 in main chapter.

3.8.7 Further Experiments

Results on ScanNet and NPM3D datasets. We provide the detail results on ScanNet in Tab. 3.13; and the detail results on NPM3D in Tab. 3.11.

CBL with Transformer. We use the open-source code base (here) to re-produce the performance of newly released PointTransformer (PT) [26] on S3DIS [44] Area 5 dataset.

In Tab. 3.14, the same consistent improvement is made on classes such as column. CBL with better boundaries further boosts the overall performance to 71.0 in mIoU, achieving a new state-of-the-art performance.

Method	mIoU	bathtub	bed	books.	cabinet	chair	counter	curtain	desk	door	floor	other	pic	fridge	shower	sink	sofa	table	toilet	wall	wndw
DCM-Net [212]	65.8	77.8	70.2	80.6	61.9	81.3	46.8	69.3	49.4	52.4	94.1	44.9	29.8	51.0	82.1	67.5	72.7	56.8	82.6	80.3	63.7
VMNet [213]	74.6	87.0	83.8	85.8	72.9	85.0	50.1	87.4	58.7	65.8	95.6	56.4	29.9	76.5	90.0	71.6	81.2	63.1	93.9	85.8	70.9
SparseConvNet [23]	72.5	64.7	82.1	84.6	72.1	86.9	53.3	75.4	60.3	61.4	95.5	57.2	32.5	71.0	87.0	72.4	82.3	62.8	93.4	86.5	68.3
MinkowskiNet [92]	73.6	85.9	81.8	83.2	70.9	84.0	52.1	85.3	66.0	64.3	95.1	54.4	28.6	73.1	89.3	67.5	77.2	68.3	87.4	85.2	72.7
O-CNN [187]	76.4	75.8	79.6	83.9	74.6	90.7	56.2	85.0	68.0	67.2	97.8	61.0	33.5	77.7	81.9	84.7	83.0	69.1	97.2	88.5	72.7
OccuSeg [188]	76.2	92.4	82.3	84.4	77.0	85.2	57.7	84.7	71.1	64.0	95.8	59.2	21.7	76.2	88.8	75.8	81.3	72.6	93.2	86.8	74.4
Mix3D [214]	78.1	96.4	85.5	84.3	78.1	85.8	57.5	83.1	68.5	71.4	97.9	59.4	31.0	80.1	89.2	84.1	81.9	72.3	94.0	88.7	72.5
BA-GEM [197] *	63.5																				
PointConv [129]	66.6	78.1	75.9	69.9	64.4	82.2	47.5	77.9	56.4	50.4	95.3	42.8	20.3	58.6	75.4	66.1	75.3	58.8	90.2	81.3	64.2
PointASNL [191]	66.6	70.3	78.1	75.1	65.5	83.0	47.1	76.9	47.4	53.7	95.1	47.5	27.9	63.5	69.8	67.5	75.1	55.3	81.6	80.6	70.3
KP-Conv [22]	68.4	84.7	75.8	78.4	64.7	81.4	47.3	77.2	60.5	59.4	93.5	45.0	18.1	58.7	80.5	69.0	78.5	61.4	88.2	81.9	63.2
FusionNet [215]	68.8	70.4	74.1	75.4	65.6	82.9	50.1	74.1	60.9	54.8	95.0	52.2	37.1	63.3	75.6	71.5	77.1	62.3	86.1	81.4	65.8
JSENet [195]	69.9	88.1	76.2	82.1	66.7	80.0	52.2	79.2	61.3	60.7	93.5	49.2	20.5	57.6	85.3	69.1	75.8	65.2	87.2	82.8	64.9
RFCR [208]	70.2	88.9	74.5	81.3	67.2	81.8	49.3	81.5	62.3	61.0	94.7	47.0	24.9	59.4	84.8	70.5	77.9	64.6	89.2	82.3	61.1
ConvNet + CBL	70.5	76.9	77.5	80.9	68.7	82.0	43.9	81.2	66.1	59.1	94.5	51.5	17.1	63.3	85.6	72.0	79.6	66.8	88.9	84.7	68.9

TABLE 3.13. Quantitative results on ScanNet [43] benchmark, results obtained from online benchmark site by the time of submission. We group method by the 3D representation type, which is respectively, from top to down, 3D + mesh, 3D voxel and 3D point, and we also use 3D point. The empty line denotes no record of detailed performance found. The method with * also considers boundary.

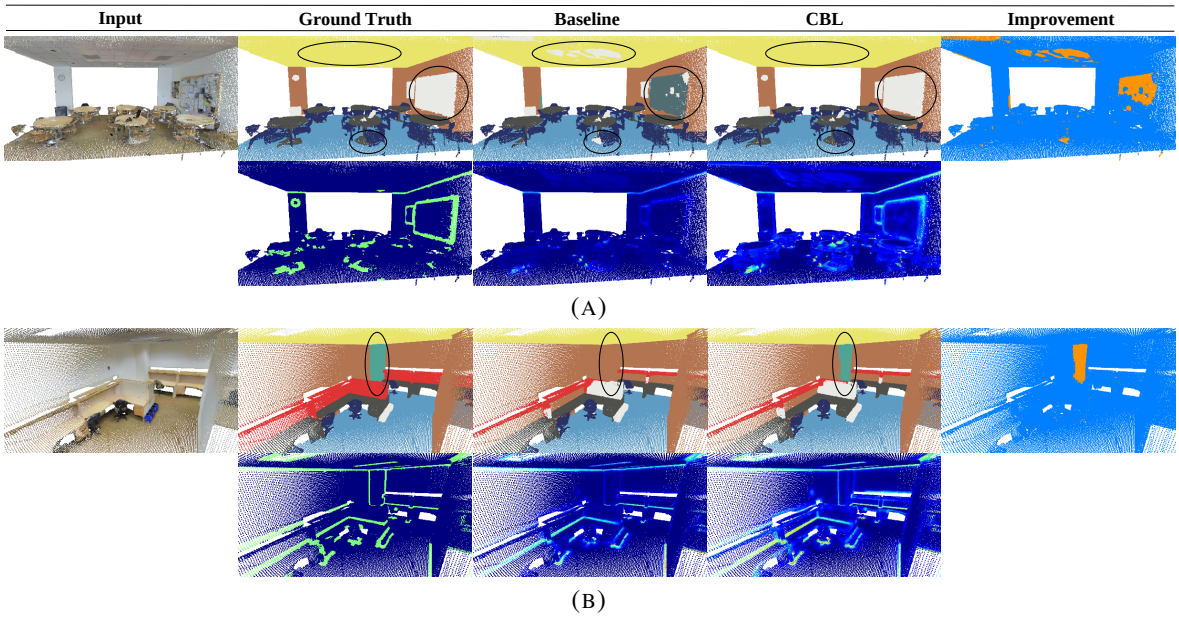


FIGURE 3.7. Large rooms. We compare the results of ConvNet baseline with CBL. On the every second row, we visualize the boundary points calculated from the ground truth label, and the feature discrimination among neighboring points for each model. The improvement on the first row and the enhanced feature discrimination on the second row show that CBL improves the features across boundaries to obtain a better segmentation quality on boundary areas. The visualization is done on S3DIS testset Area 5.

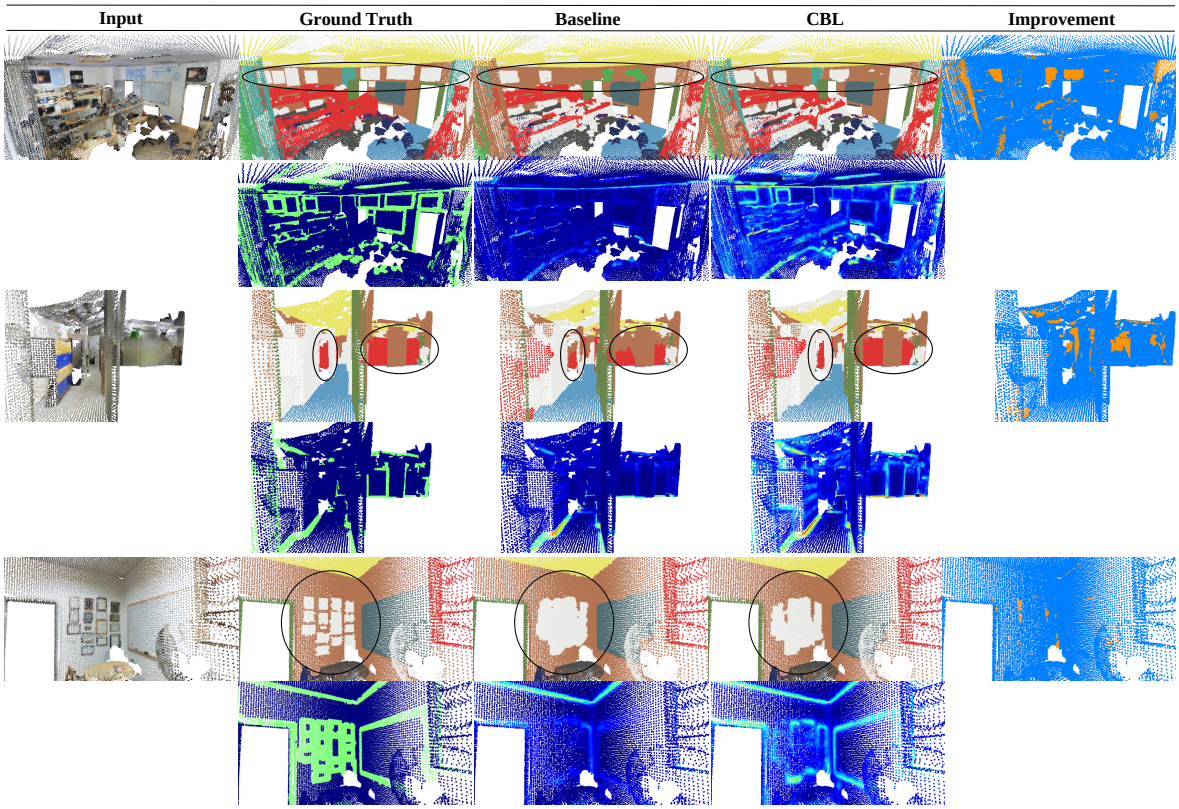
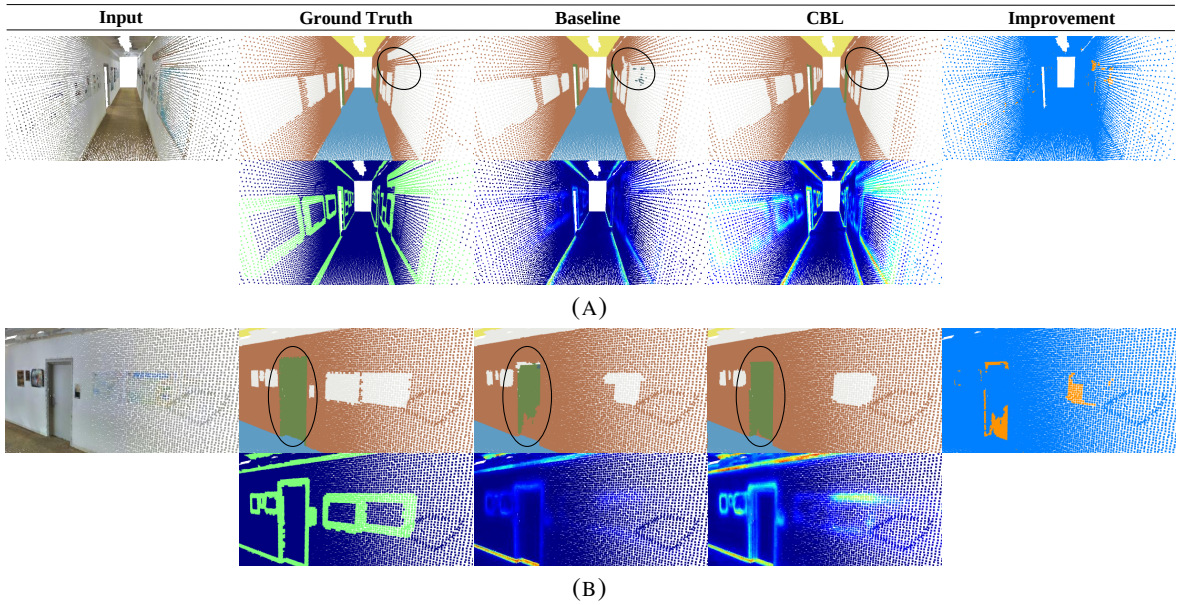


FIGURE 3.8. Cluttered space. Same as above (Fig. 3.7).



(A)

(B)

FIGURE 3.9. Hallways. Same as above (Fig. 3.7).

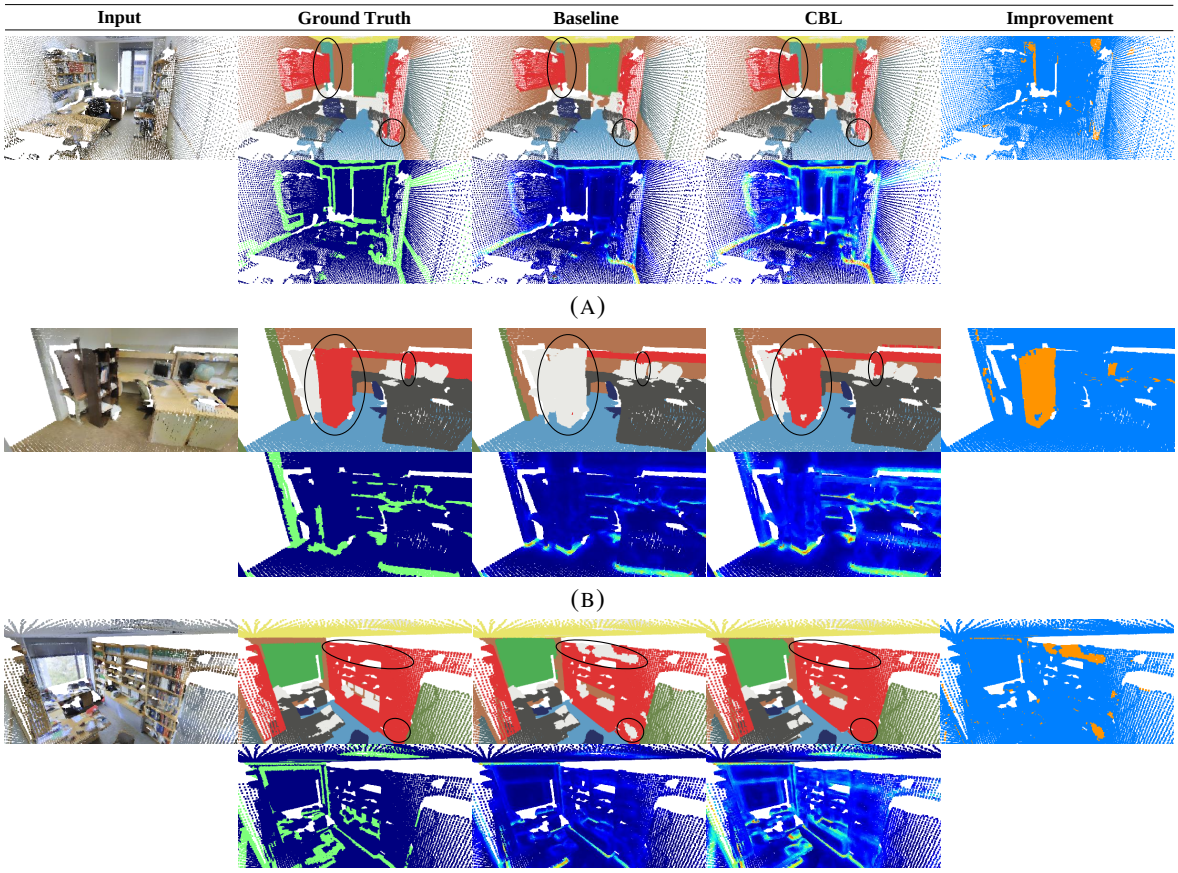


FIGURE 3.10. Offices. Same as above (Fig. 3.7).

methods	mIoU	OA	mACC	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
PT [26]*	70.4	90.8	76.5	94.0	98.5	86.3	0.0	38.0	63.4	74.3	89.1	82.4	74.3	80.2	76.0	59.3
PT [26]	70.0	90.5	76.5	95.2	98.6	85.1	0.0	36.7	62.5	75.9	81.5	91.0	75.1	71.9	76.4	60.2
+ CBL	71.0	90.9	77.5	94.3	98.3	87.4	0.0	42.1	64.0	78.5	82.5	88.9	75.1	71.1	81.3	59.6

TABLE 3.14. Quantitative results on S3DIS Area 5 dataset [44], showing the mean IoU (mIoU), overall accuracy (OA), mean accuracy (mACC), and per-class IoU scores. We include both performance reported in original paper (with *, the first row) and the re-produced performance (without *, the second row). We observe consistent improvement over both the re-produced PT, and the performance reported in original paper.

CHAPTER 4

Structuring Supervision via Entropy-Regularized Alignment

This chapter explores how to structure supervision for 3D segmentation models by explicitly handling noise in the learning signals. This problem is central to 3D point cloud segmentation, where extracting supervision from noisy and unstructured geometry can be unreliable, and we further reveal its importance for 2D images in overcoming significant geometric disturbances caused by strong data augmentations. We study label-efficient regimes in which supervision is synthesized from model predictions, and we ask how to regularize these signals so that learning remains stable across modalities and geometric perturbations.

Label-efficient segmentation aims to perform effective segmentation on input data using only sparse and limited ground-truth labels for training. This topic is widely studied in 3D point cloud segmentation due to the difficulty of annotating point clouds densely, while it is also essential for cost-effective segmentation on 2D images. Until recently, pseudo-labels have been widely employed to facilitate training with limited ground-truth labels, and promising progress has been witnessed in both the 2D and 3D segmentation. However, existing pseudo-labeling approaches could suffer heavily from the noises and variations in unlabelled data, which would result in significant discrepancies between generated pseudo-labels and current model predictions during training. We analyze that this can further confuse and affect the model learning process, which shows to be a shared problem in label-efficient learning across both 2D and 3D modalities. To address this issue, we propose a novel learning strategy to regularize the pseudo-labels generated for training, thus effectively narrowing the gaps between pseudo-labels and model predictions. More specifically, our method introduces an Entropy Regularization loss and a Distribution Alignment loss for label-efficient learning, resulting in an ERDA learning strategy. Interestingly, by using KL distance to formulate

the distribution alignment loss, ERDA reduces to a deceptively simple cross-entropy-based loss which optimizes both the pseudo-label generation module and the segmentation model simultaneously. In addition, we innovate in the pseudo-label generation to make our ERDA consistently effective across both 2D and 3D data modalities for segmentation. Enjoying simplicity and more modality-agnostic pseudo-label generation, our method has shown outstanding performance in fully utilizing all unlabeled data points for training across different label-efficient settings. This can be evidenced by promising improvement over other state-of-the-art approaches on 2D image segmentation and 3D point cloud segmentation. In some experiments, our method can even outperform fully supervised baselines using only 1% of true annotations, illustrating the importance of reducing noises and variations in labels during training. We believe these results can demonstrate that our approach represents a substantial step toward a modality-agnostic label-efficient segmentation solution. Code and model will be made publicly available at https://github.com/LiyaoTang/ERDA.

4.1 Introduction

Semantic segmentation is an important task for scene understanding, which assigns each data point a label of certain categories. Although cutting-edge fully-supervised segmentation approaches achieve promising performance, they heavily rely on large-scale densely annotated datasets that can be costly to obtain [16, 170, 218]. For example, when annotating 3D point cloud data, a single scan in ScanNet [43] dataset would require more than 20 minutes [219] of labeling work. Considering that the dataset comprises over 1500 scans with more than 10^5 points, the amount of labor time for annotating all the points in this frame would be overwhelming. Regarding 2D image data, while the annotation could be slightly easier, some downstream tasks such as medical imaging [220] demand expertise for labeling, which could still be difficult to access. Nevertheless, annotating 2D or 3D data usually involves multiple annotators for the same dataset, and the disagreement between annotators could introduce unnecessary noises to the dataset [12, 221, 222] and thus affect the model performance.

To avoid exhaustive annotation process, label-efficient learning has emerged as a promising alternative. It aims to achieve scene understanding using only limited and sparse annotations, such as semi-supervised learning and weakly-supervised learning. Effective label-efficient segmentation approaches can be advantageous for both 2D images and 3D point clouds. For 2D images, effective label-efficient segmentation can help segment image areas of interest without vast annotations, which is beneficial for downstream tasks like image inpainting [11, 223], image generation [224], and so on. For 3D point clouds, label-efficient segmentation enables more affordable use of 3D scene understanding techniques in various applications [4, 9], such as autonomous driving, unmanned aerial vehicles, and augmented reality.

Despite the benefits of label-efficient learning, one of the most significant challenges of using limited labels is that the training signals may not be sufficient to secure a robust model [168]. To tackle this problem, many leading label-efficient segmentation learning strategies [168, 225–227] attempt to generate pseudo-labels from model predictions on unlabelled data points, aiming to make the best use of unlabelled data for generating richer training signals. In the related experiments, these pseudo-label approaches have shown promising performance, but they are soon superseded by some recent consistency regularization methods [228, 229] that employ consistency constraints after randomly perturbed inputs. By analyzing the existing pseudo-labeling framework, we find that the current widely used label selection mechanism could compromise the benefits of pseudo-labels for training.

Typically, the current label selection mechanism is generally designed to select pseudo-labels with a confidence higher than some threshold, which utilizes only highly confident pseudo-labels for training and could result in under-explored unlabeled data. While if directly using less confident pseudo-labels, we find the model performance is negatively affected, which is also evidenced by studies that reveal the negative impacts of noises [170, 230] and potential unintended biases [231–233] from these low-confidence pseudo-labels. Accordingly, we hypothesize that assigning low-confidence pseudo-labels to unlabeled data introduces discrepancies between these pseudo-labels and the segmentation model outputs, which leads to unreliable and confusing training signals that hinder performance.

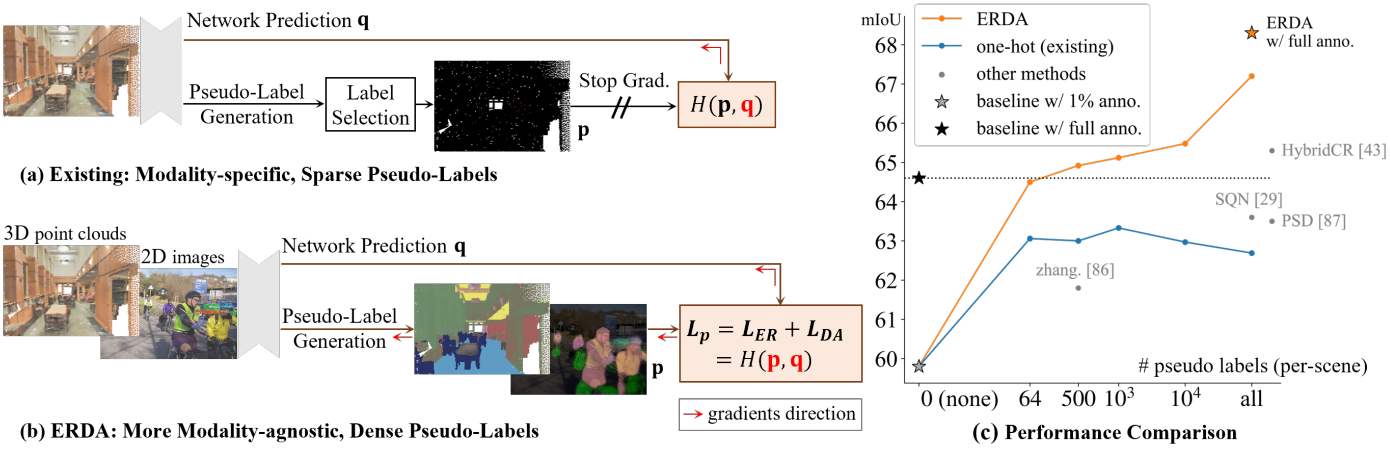


FIGURE 4.1. While existing pseudo-labels (a) are limited in the exploitation of unlabeled points, ERDA (b) simultaneously optimizes the pseudo-labels $\mathbf{p}$ and predictions $\mathbf{q}$ taking the same and simple form of cross-entropy. By reducing the noise via entropy regularization and bridging their distributional discrepancies, ERDA produces informative pseudo-labels that neglect the need for label selection. As the exemplar in (c) on 3D data, it thus enables the model to consistently benefit from more pseudo-labels, surpassing other methods and its fully-supervised baseline.

By addressing the above problem, we propose a novel learning-based pseudo-labeling framework for label-efficient segmentation, replacing the widely used hard thresholding label selection process and augmenting the segmentation performance significantly. Specifically, to reduce the level of noise in pseudo-labels and alleviate the confusion between pseudo-labels and segmentation model outputs, we introduce two learning objectives for label-efficient training with pseudo-labels. Firstly, we introduce an *entropy regularization* (ER) objective to reduce the noise and uncertainty in the pseudo-labels. This regularization promotes more informative, reliable, and confident pseudo-labels, which helps reduce the generation of noisy and uncertain pseudo-labels. Secondly, we propose a *distribution alignment* (DA) loss that minimizes statistical distances between pseudo-labels and model predictions. This ensures that the distribution of generated pseudo-labels remains close to the distribution of segmentation model predictions when regularizing their entropy, thus reducing confusion during training. In particular, we discover that formulating the distribution alignment loss using KL distance can transform our method into a deceptively simple cross-entropy-style learning objective that optimizes both the pseudo-label generator and the segmentation network simultaneously. This makes our method straightforward to implement and apply. By integrating the entropy

regularization and distribution alignment, we achieve the ERDA learning strategy, as shown in Fig. 4.1.

Furthermore, due to the significant difference in 2D and 3D data modality, the pseudo-label generation and training procedures are usually specifically designed to satisfy each modality. As a result, while both 2D and 3D modalities can suffer from similar problems as discussed above, it can still be hard to transfer methods derived from one modality to the other. For a specific instance, using 2D images, it is common to leverage rich and strong augmentations to supervise the student model with weak-to-strong pseudo-labels [227, 234] from the teacher model, while prototypical pseudo-labels [168, 225] are more common in 3D due to the insufficient augmentation methods for 3D data. Moreover, though the pseudo-label generation strategies are different, neither 2D nor 3D approaches could leverage the information of all unlabeled data due to the potential noise in pseudo-labels and the discrepancy between the distributions of pseudo-labels and model predictions. Therefore, to adapt our method to various data modalities and account for the potential rich augmentations in 2D data processing, we extend our framework by introducing a novel query-based pseudo-labeling method. Specifically, we introduce class queries with more stable embeddings to generate pseudo-labels that are more aligned under various augmentations together with the help of cross-attention. In this way, we are able to cope with the gap caused by the use of various modality-specific augmentations in data processing, and can thus enhance the ERDA learning for pseudo-label generation across modalities. We believe this can represent a substantial step towards a modality-agnostic label-efficient learning method for semantic segmentation.

Empirically, we comprehensively experiment with different label-efficient settings on both 3D and 2D datasets. Despite its concise design, our ERDA outperforms existing methods on large-scale point cloud datasets such as S3DIS [44], ScanNet [43], and SensatUrban [71], as well as 2D datasets such as Pascal [235] and Cityscapes [222]. Notably, our ERDA can surpass the fully supervised baselines using only 1% labels, demonstrating its significant effectiveness in leveraging pseudo-labels. Furthermore, we validate the scalability of our method by successfully generalizing it to other settings, which illustrates the benefits of utilizing dense pseudo-label supervision with ERDA.

This chapter develops ERDA as a noise-aware supervision strategy for semantic segmentation across 2D and 3D modalities. The main contributions of this chapter are as follow.

(1) A unified ERDA learning that couples entropy regularization with distribution alignment, yielding a deceptively simple cross-entropy-style objective that jointly improves pseudo-label generation and model learning.
(2) A query-based pseudo-labeling mechanism that improves robustness under modality-specific perturbations, notably strong 2D augmentations.
(3) Broad empirical validation across semi-supervised, sparse-label, medical-image, and unsupervised settings, supported by extensive ablations and analyses that isolate the impact of pseudo-label noise.

4.2 Related Work

Point cloud segmentation. Point cloud semantic segmentation aims to assign semantic labels to 3D points. The cutting-edge methods are deep-learning-based and can be classified into projection-based and point-based approaches. Projection-based methods project 3D points to grid-like structures, such as 2D image [79, 80, 183–186] or 3D voxels [23, 90, 92, 98, 187, 188]. Alternatively, point-based methods directly operate on 3D points [18, 19]. Recent efforts have focused on novel modules and backbones to enhance point features, such as 3D convolution [22, 24, 129, 130, 206, 217], attentions [26, 109, 131, 189, 236, 237], graph-based methods [20, 190], and other modules such as sampling [111, 114, 191] with additional supervision signals [8, 194, 195]. Although these methods have made significant progress, they rely on large-scale datasets with point-wise annotation and struggle with few labels [168]. To address the demanding requirement of point-wise annotation, our work explores weakly-supervised learning for 3D point cloud segmentation.

Weakly-supervised point cloud segmentation. Compared to weakly-supervised 2D image segmentation [227, 238–241], weakly-supervised 3D point cloud segmentation is less explored. In general, weakly-supervised 3D segmentation focuses on highly sparse labels: only a few scattered points are annotated in large point cloud scenes. Xu and Lee [168]

first propose to use 10x fewer labels to achieve performance on par with a fully-supervised point cloud segmentation model. Later studies have explored more advanced ways to exploit different forms of weak supervision [169, 219, 242, 243] and human annotations [75, 244]. Recent methods tend to introduce perturbed self-distillation [245], consistency regularization [168, 246–249], and leverage self-supervised learning [228, 229, 246, 250] based on contrastive learning [201, 251]. Pseudo-labels are another approach to leverage unlabeled data, with methods such as pre-training networks on colorization tasks [225], using iterative training [226, 252], employing separate networks to iterate between learning pseudo-labels and training 3D segmentation networks [244], or using super-point graph [190] with graph attentional module to propagate the limited labels over super-points [253]. However, these existing methods often require expensive training due to hand-crafted 3D data augmentations [229, 245, 247, 248], iterative training [226, 244, 252], or additional modules [226, 229], complicating the adaptation of backbone models from fully-supervised to weakly-supervised learning. In contrast, our work aims to achieve weakly-supervised learning with straightforward motivations and simple implementation, which is effective not only for 3D point clouds but also for other modalities such as 2D images.

Label-efficient image segmentation. Among various label-efficient settings, semi-supervised segmentation aims to learn segmentation models with only a small set of labeled images that have per-pixel annotations and an additional set of unlabeled images. With the general development of semi-supervised learning [254], there are generally two paradigms for semi-supervised segmentation, the entropy minimization [15, 45, 255–257] that leverages model prediction on unlabeled data as pseudo-labels for training, and the consistency regularization [258–265] that encourages model prediction to be invariant to the perturbations and noise on the unlabeled data. FixMatch [227] proposes to cast the model prediction on strongly perturbed unlabeled data as pseudo-labels to supervise the model prediction on weakly perturbed unlabeled data. Such weak-to-strong pseudo-labels combine the benefits from two schemes into one and have popularized in semi-supervised segmentation. Recent follow-up methods improves by proposing new pseudo-label selection criteria [240, 266–269] as well as stronger and more diverse augmentations [234, 270, 271].

Additionally, some methods incorporate other regularizations to better regulate the segmentation model, such as prototypical prediction heads [272] and contrastive learning [262]. In particular, ReCo [273] proposes to learn with sparse labels, where each image has only very few labeled pixels. Apart from weak-to-strong pseudo-labels, it samples unlabeled pixels based on model confidence to perform contrastive learning, which effectively leverages the scarce labeled pixels.

These existing works mostly focus on better exploring the supervision signals from the augmented 2D data, which largely relies on the specific processing and augmentation techniques such as mixups [265, 274, 275] and cut-outs [276]. In comparison, our method views pseudo-label generation as a unique learning target and introduces query-based pseudo-labels for end-to-end optimization together with the segmentation task to achieve a more modality-agnostic pseudo-labeling method.

Unsupervised segmentation. Unsupervised semantic segmentation aims to partition images into coherent regions without labels. Classical CRF models [41, 277] impose spatial smoothness over connected pixel grids and rely on low-level cues. Modern methods instead learn pixel-wise semantics with self-supervision, optimizing mutual information between cluster assignments [278] and enforcing consistency to stabilize pixel-level grouping [279–281] With the advent of self-supervised pre-training, many methods operate in the feature space of pre-trained backbones [36, 37], discovering prototypes or regualrizations for iterative self-distillation [282–285]. These methods typically require additional training on unlabeled images and, by construction, produce pseudo-labels for pixel-level supervision signals. Our approach is thus orthogonal and complementary, as ERDA can further learn to refine these pseudo-labels, thereby improving semantic clustering results.

Pseudo-label refinement. Pseudo-labeling [45], a versatile method for entropy minimization [15], has been extensively studied in various tasks, including semi-supervised 2D classification [231, 256], segmentation [227, 234], and domain adaptation [286, 287]. To generate high-quality supervision, various label selection strategies have been proposed based on learning status [266, 267, 288], label uncertainty [231, 286, 287, 289], class balancing [232], and data augmentations [227, 232, 234]. Our method is most closely related to the works

addressing bias and noise in supervision, where mutual learning [288, 290, 291] and distribution alignment [232, 292, 293] have been discussed. However, these works typically focus on class imbalance [232, 292] and rely on iterative training [288, 290, 291, 293], label selection [240, 269, 288, 292], and strong data augmentations [232, 292], which might not be directly applicable to 3D point clouds. For instance, common image augmentations [227] like cropping and resizing may translate to point cloud upsampling [294], which remains an open question in the related research area. Rather than introducing complicated mechanisms, we argue that proper regularization on pseudo-labels and its alignment with model prediction can provide significant benefits using a very concise learning approach designed for the weakly-supervised 3D point cloud segmentation task.

In addition, it is shown that the data augmentations and repeated training in mutual learning [166, 290] are important to avoid the feature collapse, *i.e.* the resulting pseudo-labels being uniform or the same as model predictions. We suspect the cause may originate from the entropy term in their use of raw statistical distance by empirical results, which potentially matches the pseudo-labels to noisy and confusing model prediction, as would be discussed in Sec. 4.3.3. Moreover, in self-supervised learning based on clustering [295] and distillation [36], it has also been shown that it would lead to feature collapse if matching to a cluster assignment or teacher output of a close-uniform distribution with high entropy, which agrees with the intuition in our ER term.

4.3 Methodology

In this section, we first present details about the formulation of the proposed ERDA. Subsequently, we discuss how our method can be extended to multiple modalities with the help of a query-based pseudo-labeling method.

4.3.1 Formulation of ERDA

As previously mentioned, we propose the ERDA approach to alleviate noise in the generated pseudo-labels and reduce the distribution gaps between them and the segmentation network

predictions. In general, our ERDA introduces two loss functions, including the entropy regularization loss and the distribution alignment loss for the learning on pseudo-labels. We denote the two loss functions as L_{ER} and L_{DA}, respectively. Then, we have the overall loss of ERDA as follows:

$$L_p = \lambda L_{ER} + L_{DA}, \tag{4.1}$$

where the $\lambda > 0$ modulates the entropy regularization which is similar to the studies [15, 45].

Before detailing the formulation of L_{ER} and L_{DA}, we first introduce the notation. While the losses are calculated over all unlabeled points, we focus on one single unlabeled point for ease of discussion. We denote the pseudo-label assigned to this unlabeled point as $\mathbf{p}$ and the corresponding segmentation network prediction as $\mathbf{q}$. Each $\mathbf{p}$ and $\mathbf{q}$ is a 1D vector representing the probability over classes.

Entropy Regularization loss. We hypothesize that the quality of pseudo-labels can be hindered by noise, which in turn affects model learning. Specifically, we consider that the pseudo-label could be more susceptible to containing noise when it fails to provide a confident pseudo-labeling result, which leads to the presence of a high-entropy distribution in $\mathbf{p}$.

To mitigate this, for the $\mathbf{p}$, we propose to reduce its noise level by minimizing its Shannon entropy, which also encourages a more informative labeling result [296]. Therefore, we have:

$$L_{ER} = H(\mathbf{p}), \tag{4.2}$$

where $H(\mathbf{p}) = \sum_i -p_i \log p_i$ and i iterates over the vector. By minimizing the entropy of the pseudo-label as defined above, we promote more confident labeling results to help resist noise in the labeling process[1].

Distribution Alignment loss. In addition to the noise in pseudo-labels, we propose that significant discrepancies between the pseudo-labels and the segmentation network predictions could also confuse the learning process and lead to unreliable segmentation results. In general, the discrepancies can stem from multiple sources, including the noise-induced unreliability of

[1]We note that our entropy regularization aims for entropy *minimization* on pseudo-labels, and we consider noise as the uncertain predictions by the pseudo-labels instead of incorrect predictions.

pseudo-labels, differences between labeled and unlabeled data [232], and variations in pseudo-labeling methods and segmentation methods [288, 291]. Although entropy regularization could mitigate the impact of noise in pseudo-labels, significant discrepancies may still persist between the pseudo-labels and the predictions of the segmentation network. To mitigate this issue, we propose that the pseudo-labels and network can be jointly optimized to narrow such discrepancies, making generated pseudo-labels not diverge too far from the segmentation predictions. Therefore, we introduce the distribution alignment loss.

To properly define the distribution alignment loss (L_{DA}), we measure the KL divergence between the pseudo-labels ($\mathbf{p}$) and the segmentation network predictions ($\mathbf{q}$) and aim to minimize this divergence. Specifically, we define the distribution alignment loss as follows:

$$L_{DA} = KL(\mathbf{p}||\mathbf{q}), \tag{4.3}$$

where $KL(\mathbf{p}||\mathbf{q})$ refers to the KL divergence. Using the above formulation has several benefits. For example, the KL divergence can simplify the overall loss L_p into a deceptively simple form that demonstrates desirable properties and also performs better than other distance measurements. More details will be presented in the following sections.

Simplified ERDA. With the L_{ER} and L_{DA} formulated as above, given that $KL(\mathbf{p}||\mathbf{q}) = H(\mathbf{p},\mathbf{q}) - H(\mathbf{p})$ where $H(\mathbf{p},\mathbf{q})$ is the cross entropy between $\mathbf{p}$ and $\mathbf{q}$, we can have a simplified ERDA formulation as:

$$L_p = H(\mathbf{p},\mathbf{q}) + (\lambda - 1)H(\mathbf{p}). \tag{4.4}$$

In particular, when $\lambda = 1$, we obtain the final ERDA loss[2]:

$$L_p = H(\mathbf{p},\mathbf{q}) = \sum_i -p_i \log q_i \tag{4.5}$$

The above simplified ERDA loss describes that the entropy regularization loss and distribution alignment loss can be represented by a single cross-entropy-based loss that optimizes both $\mathbf{p}$ and $\mathbf{q}$.

[2]We would justify the choice of λ in the following Sec. 4.3.3 as well as Sec. 4.4.4

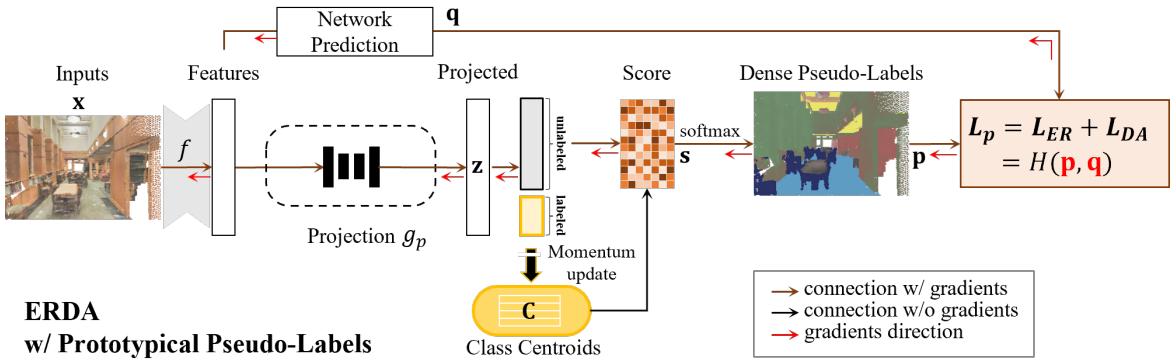


FIGURE 4.2. Detailed illustration of our ERDA with the prototypical pseudo-label generation process, which is prevalently used for 3D point cloud.

We would like to emphasize that Eq. (4.5) is distinct from the conventional cross-entropy loss. The conventional cross-entropy loss utilizes a fixed label and only optimizes the term within the logarithm function, whereas the proposed loss in Eq. (4.5) optimizes both $\mathbf{p}$ and $\mathbf{q}$ simultaneously.

4.3.2 Towards Modality-agnostic Pseudo-labeling with ERDA

Pseudo-labels are widely used in the segmentation for both 2D images and 3D point clouds when rich ground-truth labels are not available. As discussed earlier, we propose that the generated pseudo-labels would suffer from noises and variations in labels on both data modalities. As a result, the proposed ERDA method is supposed to be effective in alleviating the pseudo-label noises on both 2D and 3D data.

However, one of the most challenging problems when adapting ERDA to pseudo-labels on different modalities is that the ERDA needs to cope well with the diverse pseudo-labeling methods, which are designed for specific modalities and rise to be a unique problem when developing more modality-agnostic label-efficient learning.

For label-efficient learning on segmentation, current pseudo-labeling methods rely on high-quality teacher predictions to train the student model on unlabeled data. The production of such teacher predictions could however be modality-specific, which is rooted in the modality-specific data processing and augmentation techniques. For 2D images, strong augmentations

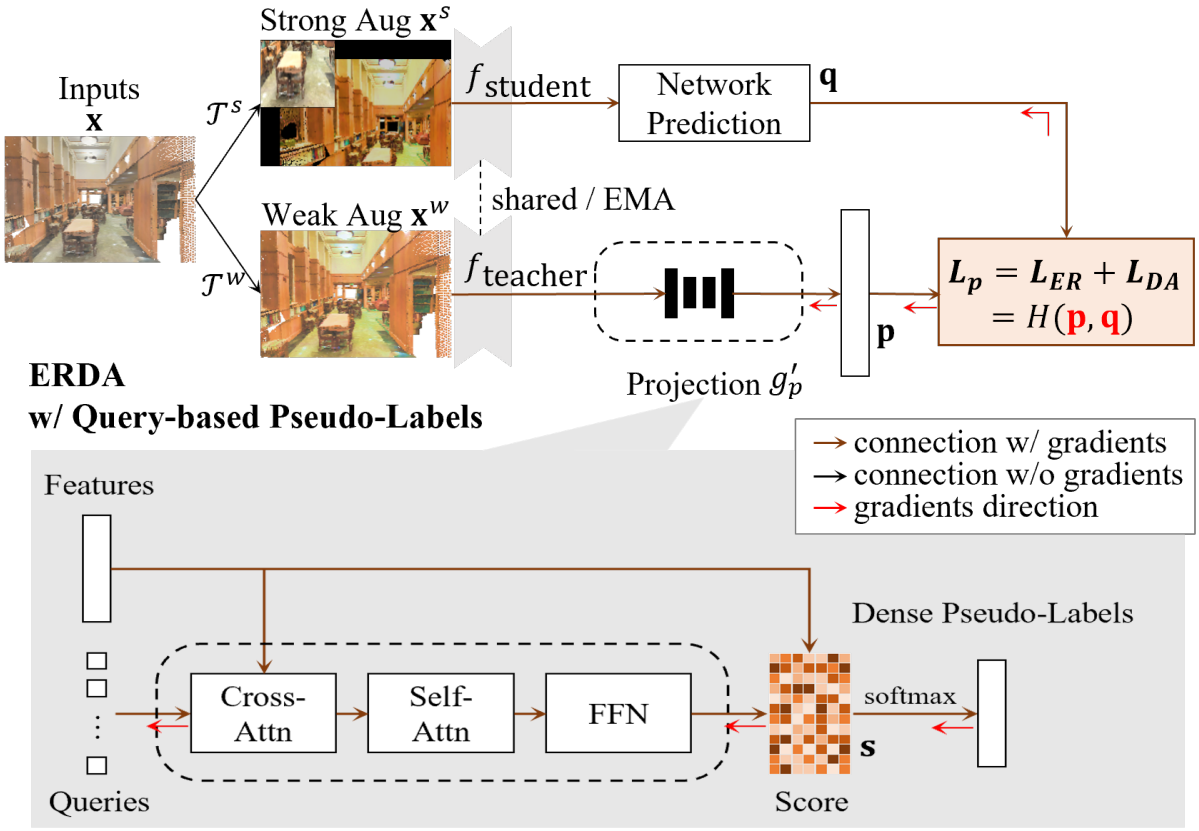


FIGURE 4.3. Illustration of our ERDA with our query-based pseudo-label generation process under the weak-to-strong framework, which are widely adopted in 2D label-efficient segmentation. The teacher model could be either shared with the student [227, 234] or an EMA-updated version of it [36, 273].

like mixups [265, 274, 275] and cut-outs [276] have been shown to be beneficial for model generalization. Researchers thus generally employ strong augmentations on the student model but weak augmentations on the teacher model to produce predictions as pseudo-labels, leading to the weak-to-strong pseudo-labeling strategy. In contrast, in 3D point clouds, strong augmentations like mixups and cut-outs are not usually used because they could change the spatial relation and structural information that are of great importance in 3D modeling. Weak augmentations are thus uniformly applied on both student and teacher models, and prototypical features are commonly used for pseudo-label generation. We illustrate different pseudo-labeling procedures in Fig. 4.2 and Fig. 4.3, and provide more discussion and details for the common practice of augmentations across data modalities, as well as how various augmentations are employed in the training with pseudo-labels, in the supplementary Sec. 4.6.1.

When applying ERDA on 3D data, noises or confusions in pseudo-labels induced by augmentation are limited and can thus be easily dealt with. Instead, when applying ERDA on the 2D data, the disturbances from strong data augmentation might act like large noises in the pseudo-labels. Directly using ERDA to diminish the noises from pseudo-labels on 2D data may thus offset the benefits brought by both data augmentation and ERDA, as we expect the ERDA learning to specifically reduce the noise within pseudo-labels but not the noise brought by different level of data augmentations[3]. Due to these differences in data processing and augmentation strategies, current research towards modality-agnostic pseudo-labeling methods is very limited. By addressing this, we proposed to enhance the data augmentation awareness for ERDA to make it better fit to the 2D images, making the proposed ERDA a much more modality-agnostic strategy that can significantly enhance label-efficient segmentation on both 2D and 3D data.

In this study, we propose to specifically account for the gaps between pseudo-labels and student network predictions caused by augmentations in pseudo-label generation, so that ERDA can better address the non-augmentation noises, without affecting the benefits of data augmentations. In the following, we will discuss in detail how we adapt ERDA to different data modalities for pseudo-label learning, making our overall ERDA approach an innovative and more modality-agnostic approach.

Prototypical pseudo-labeling with augmentations. For our ERDA, we derive pseudo-labels based on prototypes due to simplicity and effectiveness [168, 225, 297] for 3D point clouds as well as 2D images. Specifically, as shown in Fig. 4.2, prototypes [298] denote the class centroids in the feature space, which are calculated based on the limited labeled data, and pseudo-labels can be estimated based on the feature distances between unlabeled points and the prototypical class centroids.

[3]We provide more empirical analysis in Sec. 4.4.4 and Tab. 4.11b.

That is, supposing that $\mathcal{T}$ denotes the common augmentations applied on input data $\mathbf{x}$, we could then generate pseudo-label $\mathbf{p}$ and network prediction $\mathbf{q}$ based on:

$$\begin{cases} \mathbf{p} = \mathit{SoftMax}\big(cos[g_p(\mathcal{T}(\mathbf{x})), \mathbf{C}]\big), \\ \mathbf{q} = g_q(\mathcal{T}(\mathbf{x}))), \end{cases} \tag{4.6}$$

where g_p and g_q represent a network that maps input data to a feature for pseudo-label generation and network prediction, respectively, $cos[\cdot, \cdot]$ is cosine similarity measurement, and $\mathbf{C} \in \mathbb{R}^{d \times K}$ is the collection of class centroids, *i.e.* prototypes. d is the number of feature dimensions and K is the number of classes.

To avoid expensive computational costs and compromised representations for each semantic class [225, 228, 229], momentum update is utilized as an approximation to obtain global class centroids. Given $\mathcal{X}^l$ as the collection of labeled data and $\mathcal{X}^u$ as the collection of unlabelled data, the momentum-based prototype update for a specific class centroid $C_k \in \mathbf{C}$ is calculated by:

$$C_k \leftarrow mC_k + (1-m)\hat{C}_k, \;\; \hat{C}_k = \frac{1}{N_k} \sum_{\mathbf{x} \in \mathcal{X}^l \wedge y = k} g_p(\mathcal{T}(\mathbf{x}))) \tag{4.7}$$

where N_k is the number of labeled points of the k-th class.

We note that, in Eq. (4.6), both pseudo-labels and model predictions are generated through weakly augmented 3D point cloud data. However, when using 2D data, g_p and g_q are provided with different augmentations $\mathcal{T}$, *e.g.* weak augmentations $\mathcal{T}^{\mathrm{w}}$ for g_p and strong augmentations $\mathcal{T}^{\mathrm{s}}$ for g_q in a weak-to-strong strategy [227, 273]. Such practice naturally and intentionally introduces gaps between $\mathbf{p}$ and $\mathbf{q}$, which are shown to be useful for model robustness. To this end, we propose to innovate g_p to make it aware of the augmentation-related gaps and focus more on the noise within pseudo-labels.

Query-based pseudo-label for strong augmentations. As illustrated in Fig. 4.3, we attempt to make the pseudo-label generation aware of the gaps between pseudo-labels and student network predictions caused by augmentations. To achieve this, we first leverage query embeddings. Since the query embeddings are not dependent on the input, they can be less sensitive to the gaps caused by diverse data augmentations used on 2D images when generating

pseudo-labels. Using cross-attention, we can then associate these more "augmentation-insensitive" embeddings with the current backbone features, which would account for the irrelevant noise induced by the use of strong data augmentation. With such a process, we can generate pseudo-labels that would be more sensitive to noises related to the pseudo-labels rather than the augmentations, making our ERDA more helpful for better learning.

We propose the query-based pseudo-labels:

$$\mathbf{p} = \mathcal{A}(\mathbf{C}^Q, g_p^K(\mathbf{x}))g_p^V(\mathbf{x}), \tag{4.8}$$

where we directly generate pseudo-labels by conditioning the queries $\mathbf{C}^Q$ on the current data through cross-attention $\mathcal{A}$. We omit the transformation $\mathcal{T}$ for clarity. g_p^K and g_p^V are the functions to extract key and value features, respectively, and generally leverage teacher model as shown in Fig. 4.3. Here, the query, key, and value align with the transformer decoder [299], leading to a pseudo-label generation process based on the queries $\mathbf{C}^Q$, which is implemented as learnable embeddings in $\mathbb{R}^{d\times K}$ to be optimized by Eq. (4.1).

Discussion. Although we introduce the transformer decoder with query embeddings and cross-attention, we would like to emphasize that our approach is novel and different from existing formulations. On the one hand, we re-formulate the decoding process for our pseudo-label generation by using the query embeddings to encode class information that is more augmentation-insensitive. On the other hand, the employed query embeddings replace the typical class-centroids in prototypical pseudo-labels defined in Eq. (4.7). Compared with conservative updates such as momentum updates, the cross-attention between queries and current features makes the pseudo-label generation process aware of the use of various augmentations on the current input.

Our novel formulation enables our method to be effective in handling various data processing and augmentation methods for pseudo-label generation, such as the weak-to-strong approach on 2D images. Weak-to-strong approach combines the pseudo-labeling approaches with consistency regularizations through the use of rich data augmentations, especially the augmentations on image data. Typically, to enhance such regularization, existing works [227, 273] use the teacher model to generate pseudo-labels on images with weak augmentations, and

L_{DA}	$KL(\mathbf{p}\|\|\mathbf{q})$	$KL(\mathbf{q}\|\|\mathbf{p})$	$JS(\mathbf{p},\mathbf{q})$	$MSE(\mathbf{p},\mathbf{q})$
L_p	$H(\mathbf{p},\mathbf{q})-(1-\lambda)H(\mathbf{p})$	$H(\mathbf{q},\mathbf{p})-H(\mathbf{q})+\lambda H(\mathbf{p})$	$H(\frac{\mathbf{p}+\mathbf{q}}{2})-(\frac{1}{2}-\lambda)H(\mathbf{p})-\frac{1}{2}H(\mathbf{q})$	$\frac{1}{2}\sum_i(p_i-q_i)^2+\lambda H(\mathbf{p})$
S1	0	$q_i-\mathbb{1}_{k=i}$	0	0
S2	$(\lambda-1)p_i\sum_j p_j\log\frac{p_i}{p_j}$	$\frac{1}{K}-p_i+\lambda p_i\sum_j p_j\log\frac{p_i}{p_j}$	$p_i\sum_{j\neq i}p_j(\frac{1}{2}\log\frac{Kp_i+1}{Kp_j+1}+(\lambda-\frac{1}{2})\log\frac{p_i}{p_j})$	$-p_i^2+p_i\sum_j p_j^2+\lambda p_i\sum_j p_j\log\frac{p_i}{p_j}$

TABLE 4.1. The formulation of L_p using different functions to formulate L_{DA}. We study the gradient update on s_i, *i.e.* $-\frac{\partial L_p}{\partial s_i}$ under different situations. **S1**: update given confident pseudo-label, $\mathbf{p}$ being one-hot with $\exists p_k \in \mathbf{p}, p_k \to 1$. **S2**: update given confusing prediction, $\mathbf{q}$ being uniform with $q_1 = ... = q_K = \frac{1}{K}$. More analysis as well as visualization can be found in the Sec. 4.3.3 and the supplementary Sec. 4.6.2.

train the student model on images with strong augmentations. Thanks to the well-studied augmentation techniques [276, 300] on 2D images, such a weak-to-strong approach has been widely adopted in 2D label-efficient segmentation tasks [227, 234, 273]. Nonetheless, the use of different augmentations consistently induces large gaps between generated pseudo-labels and the model predictions, which could overwhelm and hinder the ERDA learning. In comparison, with the query-based pseudo-label, the decoder can be optimized to generate better and more aligned pseudo-labels for the student model under strong augmentations, and makes ERDA learning focus better on the non-augmentation noise within pseudo-labels.

From the perspective of pseudo-label generation, query-based pseudo-labels could be viewed as an effective way of integrating prototypical pseudo-labels and weak-to-strong approach to enjoy both compact representation and strong augmentations. In this view, we marry the ERDA learning to the weak-to-strong approach with a typical choice of g_p being transformer decoder as in Eq. (4.8) and producing pseudo-labels as Eq. (4.6). More comparison and analysis can be found in the ablation (Sec. 4.4.4) and supplementary.

4.3.3 Delving into the Benefits of ERDA

To formulate the distribution alignment loss, different functions can be employed to measure the differences between $\mathbf{p}$ and $\mathbf{q}$. In addition to the KL divergence, there are other distance measurements like mean squared error (MSE) or Jensen-Shannon (JS) divergence for replacement. Although many mutual learning methods [166, 288, 291, 293] have proven the effectiveness of KL divergence, a detailed comparison of KL divergence against other

measurements is currently lacking in the literature. In this section, under the proposed ERDA learning framework, we show by comparison that $KL(\mathbf{p}||\mathbf{q})$ is a better choice, and ER is necessary for label-efficient segmentation.

To examine the characteristics of different distance measurements, including $KL(\mathbf{p}||\mathbf{q})$, $KL(\mathbf{q}||\mathbf{p})$, $JS(\mathbf{p}||\mathbf{q})$, and $MSE(\mathbf{p}||\mathbf{q})$, we investigate the form of our ERDA loss L_p and its impact on the learning for pseudo-label generation network given two situations during training.

More formally, we shall assume a total of K classes and define that a pseudo-label $\mathbf{p} = [p_1, ..., p_K]$ is based on the confidence scores $\mathbf{s} = [s_1, ..., s_K]$, and that $\mathbf{p} = \text{softmax}(\mathbf{s})$. Similarly, we have a segmentation network prediction $\mathbf{q} = [q_1, ..., q_K]$ for the same point. We re-write the ERDA loss L_p in various forms and investigate the learning from the perspective of gradient update, as in Tab. 4.1.

Situation 1: Gradient update given confident pseudo-label $\mathbf{p}$. We first specifically study the case when $\mathbf{p}$ is very certain and confident, *i.e.* $\mathbf{p}$ approaching a one-hot vector. As in Tab. 4.1, most distances yield the desired zero gradients, which thus retain the information of a confident and reliable $\mathbf{p}$. In this situation, however, the $KL(\mathbf{q}||\mathbf{p})$, rather than $KL(\mathbf{p}||\mathbf{q})$ in our method, produces non-zero gradients that would actually increase the noise among pseudo-labels during its learning, which is not favorable according to our motivation.

Situation 2: Gradient update given confusing prediction $\mathbf{q}$. In addition, we are also interested in how different choices of distance and λ would impact the learning on pseudo-label if the segmentation model produces confusing outputs, *i.e.* $\mathbf{q}$ tends to be uniform. In line with the motivation of ERDA learning, we aim to regularize the pseudo-labels to mitigate potential noise and bias, while discouraging uncertain labels with little information. However, as in Tab. 4.1, most implementations yield non-zero gradient updates to the pseudo-label generation network. This update would make $\mathbf{p}$ closer to the confused $\mathbf{q}$, thus increasing the noise and degrading the training performance. Conversely, only $KL(\mathbf{p}||\mathbf{q})$ can produce a zero gradient when integrated with the entropy regularization with $\lambda = 1$. That is, only ERDA in Eq. (4.5) would not update the pseudo-label generation network when $\mathbf{q}$ is not reliable, which

avoids confusing the $\mathbf{p}$. Furthermore, when $\mathbf{q}$ is less noisy but still close to a uniform vector, it is indicated that there is a large close-zero plateau on the gradient surface of ERDA, which benefits the learning on $\mathbf{p}$ by resisting the influence of noise in $\mathbf{q}$.

In addition to the above cases, the gradients of ERDA in Eq. (4.5) could be generally regarded as being aware of the noise level and the confidence of both pseudo-label $\mathbf{p}$ and the corresponding prediction $\mathbf{q}$. Especially, ERDA produces larger gradient updates on noisy pseudo-labels, while smaller updates on confident and reliable pseudo-labels or given noisy segmentation prediction. Therefore, our formulation demonstrates its superiority in fulfilling our motivation of simultaneous noise reduction and distribution alignment, where both L_{ER} and KL-based L_{DA} are necessary. We provide more empirical studies in ablation (Sec. 4.4.4) and detailed analysis in the supplementary.

4.3.4 Overall Objective

Finally, with ERDA learning in Eq. (4.5), we minimize the same loss for both labeled and unlabeled points, segmentation task, and pseudo-label generation, where we allow the gradient to back-propagate through the (pseudo-)labels. The final loss is given as

$$L = \frac{1}{N^l} \sum_{\mathbf{x} \in \mathcal{X}^l} L_{ce}(\mathbf{q}, y) + \alpha \frac{1}{N^u} \sum_{\mathbf{x} \in \mathcal{X}^u} L_p(\mathbf{q}, \mathbf{p}), \tag{4.9}$$

where $L_p(\mathbf{q}, \mathbf{p}) = L_{ce}(\mathbf{q}, \mathbf{p}) = H(\mathbf{q}, \mathbf{p})$ is the typical cross-entropy loss used for point cloud segmentation, N^l and N^u are the numbers of labeled and unlabeled points, and α is the loss weight.

Aligned with our motivation, we do not introduce thresholding-based label selection or one-hot conversion [45, 225] to process generated pseudo-labels. Due to the simplicity of ERDA, we are able to follow the setup of the baselines for training, which enables straightforward implementation and easy adaptation on various backbone models and supervision settings with little overhead. More details are in the supplementary.

settings	methods	mIoU	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
Fully	PointNet [18]	41.1	88.8	97.3	69.8	0.1	3.9	46.3	10.8	59.0	52.6	5.9	40.3	26.4	33.2
	MinkowskiNet [92]	65.4	91.8	**98.7**	86.2	0.0	34.1	48.9	62.4	81.6	**89.8**	47.2	74.9	74.4	58.6
	KPConv [22]	65.4	92.6	97.3	81.4	0.0	16.5	54.5	69.5	90.1	80.2	74.6	66.4	63.7	58.1
	SQN [226]	63.7	92.8	96.9	81.8	0.0	25.9	50.5	65.9	79.5	85.3	55.7	72.5	65.8	55.9
	HybridCR [228]	65.8	93.6	98.1	82.3	0.0	24.4	59.5	66.9	79.6	87.9	67.1	73.0	66.8	55.7
	RandLA-Net [109]	64.6	92.4	96.8	80.8	0.0	18.6	57.2	54.1	87.9	79.8	74.5	70.2	66.2	59.3
	+ ERDA	68.4	93.9	98.5	83.4	0.0	28.9	62.6	70.0	89.4	82.7	75.5	69.5	75.3	58.7
	CloserLook [206]	66.2	94.2	98.1	82.7	0.0	22.2	57.6	70.4	91.2	81.2	75.3	61.7	65.8	60.4
	+ ERDA	69.6	94.5	98.5	85.2	0.0	31.1	57.3	72.2	91.7	83.6	77.6	74.8	75.8	62.1
	PT [26]	70.4	94.0	98.5	86.3	0.0	38.0	**63.4**	74.3	89.1	82.4	74.3	80.2	**76.0**	59.3
	+ ERDA	**72.6**	**95.8**	98.6	**86.4**	0.0	**43.9**	61.2	**81.3**	**93.0**	84.5	**77.7**	**81.5**	74.5	**64.9**
0.02% (1pt)	zhang *et al.* [225]	45.8	-	-	-	-	-	-	-	-	-	-	-	-	-
	PSD [245]	48.2	**87.9**	96.0	62.1	0.0	20.6	49.3	40.9	55.1	61.9	43.9	50.7	27.3	31.1
	MIL-Trans [229]	51.4	86.6	93.2	**75.0**	0.0	**29.3**	45.3	**46.7**	60.5	62.3	56.5	47.5	33.7	32.2
	HybridCR [228]	51.5	85.4	91.9	65.9	0.0	18.0	**51.4**	34.2	63.8	**78.3**	52.4	**59.6**	29.9	**39.0**
	RandLA-Net [109]	40.6	84.0	94.2	59.0	0.0	5.4	40.4	16.9	52.8	51.4	52.2	16.9	27.8	27.0
	+ ERDA	48.4	87.3	96.3	61.9	0.0	11.3	45.9	31.7	73.1	65.1	**57.8**	26.1	**36.0**	36.4
	CloserLook [206]	34.6	33.6	40.5	52.4	0.0	21.1	25.4	35.5	48.9	48.9	53.9	23.8	35.3	30.1
	+ ERDA	**52.0**	90.0	96.7	70.2	0.0	21.5	45.8	41.9	**76.0**	65.5	56.1	51.5	30.6	30.9
	PT [26]	2.2	0.0	0.0	29.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	+ ERDA	26.2	86.8	**96.9**	63.2	0.0	0.0	0.0	15.1	29.6	26.3	0.0	0.0	0.0	22.8
1%	zhang *et al.* [225]	61.8	91.5	96.9	80.6	0.0	18.2	58.1	47.2	75.8	85.7	65.2	68.9	65.0	50.2
	PSD [245]	63.5	92.3	97.7	80.7	0.0	27.8	56.2	62.5	78.7	84.1	63.1	70.4	58.9	53.2
	SQN [226]	63.6	92.0	96.4	81.3	0.0	21.4	53.7	73.2	77.8	**86.0**	56.7	69.9	66.6	52.5
	HybridCR [228]	65.3	92.5	93.9	82.6	0.0	24.2	**64.4**	63.2	78.3	81.7	69.0	74.4	68.2	56.5
	RandLA-Net [109]	59.8	92.3	97.5	77.0	0.1	15.9	48.7	38.0	83.2	78.0	68.4	62.4	64.9	50.6
	+ ERDA	67.2	94.2	97.5	82.3	0.0	27.3	60.7	68.8	88.0	80.6	76.0	70.5	68.7	58.4
	CloserLook [206]	59.9	95.3	**98.4**	78.7	0.0	14.5	44.4	38.1	84.9	79.0	69.5	67.8	53.9	54.1
	+ ERDA	68.2	94.0	98.2	83.8	0.0	30.2	56.7	62.7	**91.0**	80.8	75.4	80.2	74.5	58.3
	PT [26]	65.8	94.2	98.2	83.0	0.0	**44.2**	50.4	68.8	88.1	83.0	75.2	47.4	64.3	59.0
	+ ERDA	**70.4**	**95.5**	98.1	**85.5**	0.0	30.5	61.7	**73.3**	90.1	82.6	**77.6**	**80.6**	**76.0**	**63.1**
10%	Xu and Lee [168]	48.0	90.9	97.3	74.8	0.0	8.4	49.3	27.3	69.0	71.7	16.5	53.2	23.3	42.8
	Semi-sup [250]	57.7	-	-	-	-	-	-	-	-	-	-	-	-	-
	zhang *et al.* [225]	64.0	-	-	-	-	-	-	-	-	-	-	-	-	-
	SQN [226]	64.7	93.0	97.5	81.5	0.0	28.0	55.8	68.7	80.1	**87.7**	55.2	72.3	63.9	57.0
	RandLA-Net [109]	61.7	91.7	97.8	79.4	0.0	28.4	50.8	45.5	85.2	81.3	70.3	57.1	63.8	51.8
	+ ERDA	67.9	94.3	98.4	83.2	0.0	30.5	60.7	67.4	88.8	83.2	74.5	68.8	72.4	60.4
	CloserLook [206]	55.5	93.0	98.2	73.6	0.0	12.6	25.6	33.3	87.5	72.9	65.1	73.1	36.0	51.1
	+ ERDA	69.1	**94.7**	**98.5**	83.2	0.0	28.8	53.8	70.9	**91.5**	82.5	75.8	**82.1**	**75.3**	61.6
	PT [26]	66.0	93.7	98.3	83.7	0.0	35.0	48.1	70.9	88.3	81.9	73.2	60.3	67.3	57.2
	+ ERDA	**71.7**	94.6	**98.5**	**86.5**	0.0	**49.7**	**61.3**	**82.4**	89.8	84.3	**78.0**	70.5	74.1	**62.4**

TABLE 4.2. The results are obtained on the S3DIS datasets Area 5. For all baseline methods, we follow their official instructions in evaluation. The **bold** denotes the best performance in each setting.

4.4 Experiments

We present the benefits of our proposed ERDA by experimenting with multiple large-scale datasets on both 2D and 3D modalities. We also provide ablation studies for better investigation.

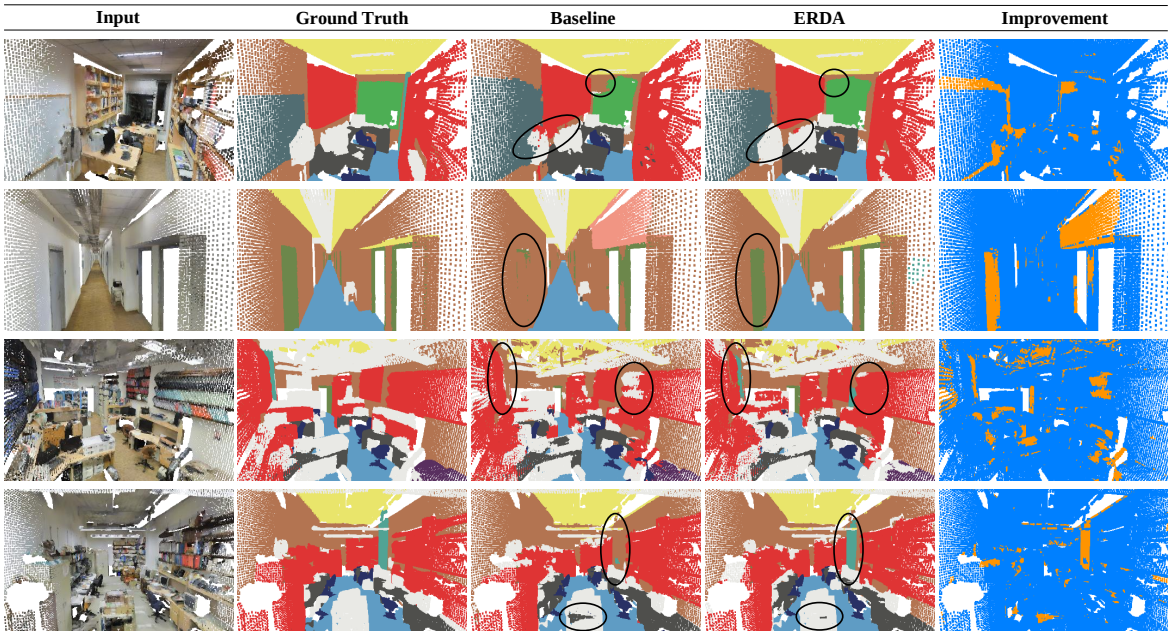


FIGURE 4.4. We show obvious improvement of our ERDA over baseline (RandLA-Net) on different scenes from S3DIS Area 5. In the office and hallway (top 2), ERDA produces more detailed and complete segmentation for windows and doors, and avoids over-expansion of the board and bookcase on the wall, thanks to the informative pseudo-labels. In more cluttered scenes (bottom 2), ERDA tends to make cleaner predictions by avoiding improper situations such as desk inside clutter and preserving important semantic classes such as columns.

4.4.1 Experimental Setup

3D point cloud segmentation. We choose RandLA-Net [109] and CloserLook [206] as our primary baselines following previous works. Additionally, while transformer models [301, 302] have revolutionized the field of computer vision as well as 3D point cloud segmentation [26, 131], none of the existing works have addressed the training of transformer for point cloud segmentation with weak supervision, even though these models are known to be data-hungry [301]. We thus further incorporate the PointTransformer (PT) [26] as our baseline to study the amount of supervision demanded for effective training of transformer on 3D point cloud.

For training, we follow the setup of the baselines and set the loss weight $\alpha = 0.1$. For a fair comparison, we follow previous works [225, 226, 245] and experiment with different settings,

settings	methods	mIoU
Fully	PointCNN [24]	45.8
	RandLA-Net [109]	64.5
	KPConv [22]	68.4
	HybridCR [228]	59.9
	CloserLook + **ERDA**	70.4
20pts	MIL-Trans [229]	54.4
	CloserLook + **ERDA**	57.0
0.1%	SQN [226]	56.9
	RandLA-Net + **ERDA**	62.0
1%	zhang *et al*. [225]	51.1
	PSD [245]	54.7
	HybridCR [228]	56.8
	RandLA-Net + **ERDA**	63.0

TABLE 4.3. Results on ScanNet test set.

settings	methods	cat. mIoU	ins. mIoU
Fully	KPConv [22]	85.0	86.2
	CloserLook [206]	84.3	86.0
	PT [26]	83.7	86.6
	CloserLook + **ERDA**	85.9	86.6
	PT + **ERDA**	85.2	86.7
1pt	CloserLook [206]	74.7	80.6
	PT [26]	76.7	81.5
	CloserLook + **ERDA**	78.3	81.9
	PT + **ERDA**	78.5	82.5
1%	CloserLook [206]	81.9	84.2
	PT [26]	82.2	85.0
	CloserLook + **ERDA**	83.4	85.2
	PT + **ERDA**	83.3	85.6

TABLE 4.4. Results on ShapeNetPart dataset.

methods	1-pixel	1%	5%	25%
Supervised	60.3	66.2	69.2	73.8
FixMatch [227]	63.7	71.0	72.9	75.8
+ ReCo [273]	66.1	72.7	74.1	76.0
+ ERDA	70.2	74.1	75.1	76.4

TABLE 4.5. Sparse-label results on Pascal.

methods	1 case	3 cases	7 cases
UA-MT [304]	-	61.0	81.5
CNN&Trans [305]	-	65.6	86.4
UniMatch [234]	85.4	88.9	89.9
+ ERDA	86.7	89.9	91.1

TABLE 4.6. Results on ACDC medical images.

including the 0.02% (1pt), 1% and 10% settings, where the available labels are randomly sampled according to the ratio[4]. More details are given in the supplementary.

2D image segmentation. To systematically analyze the effectiveness of ERDA learning across the modalities, we consider several important 2D label-efficient settings [234, 267, 268, 303], including semi-supervised and sparse-label settings, and also generalize to medical images as well as unsupervised settings.

More specifically, for semi-supervised settings, we follow the standard evaluation protocols from FixMatch [227, 234], which split the training set into two subsets, which are labeled and unlabeled images. We also follow the choice of baseline to use DeepLabv3+ [31] with ResNet-101 [54] as the backbone network. For sparse-label, we follow ReCo [273] that allocates annotations in the form of the number/ratio of labeled pixels per image. It also follows the FixMatch [227] in training and generating the pseudo-labels. For medical images, we follow the Unimatch [234] in using FixMatch with enhanced augmentations on medical image dataset. For unsupervised learning, we build upon the state-of-art method, SmooSeg [303], which utilizes the recent large vision transformer [36, 301] as the strong backbone. By default, we implement the query-based pseudo-labeling method using a transformer decoder with one transformer layer, which keeps our method simple and lightwight. More details are given in the supplementary.

[4]Some super-voxel-based approaches, such as OTOC [244], additionally leverage the super-voxel partition from the dataset annotations [226]. We thus treat them as a different setting and avoid direct comparison.

settings	methods	mIoU
Fully	PointNet [18]	23.7
	PointNet++ [19]	32.9
	RandLA-Net [109]	52.7
	KPConv [22]	57.6
	LCPFormer [306]	63.4
	RandLA-Net + **ERDA**	64.7
0.1%	SQN [226]	54.0
	RandLA-Net + **ERDA**	56.4

TABLE 4.7. Results on SensatUrban test.

methods	92	183	366	732	1464
Supervised	45.1	55.3	64.8	69.7	73.5
CPS [270]	64.1	67.4	71.7	75.9	-
ST++ [269]	65.2	71.0	74.6	77.3	79.1
PS-MT [271]	65.8	69.6	76.6	78.4	80.0
U^{2}PL [240]	68.0	69.2	73.7	76.2	79.5
PCR [272]	70.1	74.7	77.2	78.5	80.7
FixMatch [227]	63.9	73.0	75.5	77.8	79.2
+ **ERDA**	74.9	76.6	78.1	78.7	80.4

TABLE 4.8. Semi-supervised results on Pascal.

methods	backbone	Acc.	mIoU
IIC [278]	ResNet+FPN	47.9	6.4
MDC [279]	ResNet+FPN	40.7	7.1
PiCIE [279]	ResNet+FPN	65.5	12.3
DINO [36]	ViT-S/8	40.5	13.7
+ TransFGU [283]	ViT-S/8	77.9	16.8
+ STEGO [284]	ViT-S/8	69.8	17.6
+ SmooSeg [303]	ViT-S/8	82.8	18.4
+ **ERDA**	ViT-S/8	82.3	20.5

TABLE 4.9. Unsupervised results on Cityscapes.

4.4.2 Performance Comparison on 3D Segmentation

Results on S3DIS. S3DIS [44] is a large-scale point cloud segmentation dataset that covers 6 large indoor areas with 272 rooms and 13 semantic categories. As shown in Tab. 4.2, ERDA significantly improves over different baselines on all settings and almost all classes. In particular, for confusing classes such as column, window, door, and board, our method provides noticeable and consistent improvements in all weak supervision settings. We also note that PT suffers from severe over-fitting and feature collapsing under the supervision of extremely sparse labels of the "1pt" setting; whereas it is alleviated with ERDA, though not achieving a satisfactory performance. Such observation agrees with the understanding that the transformer is data-hungry [301].

Impressively, ERDA yields competitive performance against most supervised methods. For instance, with only 1% of labels, it achieves performance better than its stand-alone baselines with full supervision. Such result indicates that the ERDA is more successful than expected in alleviating the lack of training signals, as also demonstrated qualitatively in Fig. 4.4.

Therefore, we further extend the proposed method to fully-supervised training, *i.e.* in setting "Fully" in Tab. 4.2. More specifically, we generate pseudo-labels for all points and regard the ERDA as an auxiliary loss for fully-supervised learning. Surprisingly, we observe non-trivial improvements (+3.7 for RandLA-Net and +3.4 for CloserLook) and achieve the state-of-the-art performance of 72.6 (+2.2) in mIoU with PT. We suggest that the improvements are due to the noise-aware learning from ERDA, which gradually reduces the noise during the model learning and demonstrates to be generally effective. Moreover, considering that the ground-truth labels could suffer from the problem of label noise [12, 166, 167], we also

hypothesize that pseudo-labels from ERDA learning could stabilize fully-supervised learning and provide unexpected benefits.

We also conduct the 6-fold cross-validation, as reported in Tab. 4.13 in Sec. 4.6.3. In general, we find our method achieves a leading performance among both weakly-supervised and fully-supervised methods, which validates the effectiveness of our method.

Results on ScanNet. ScanNet [43] is an indoor point cloud dataset that covers 1513 training scenes and 100 test scenes with 20 classes. In addition to the common settings, *e.g.* 1% labels, it also provides official data efficient settings, such as 20 points, where for each scene there are a pre-defined set of 20 points with the ground truth label. We evaluate both settings and report the results in Tab. 4.3. We largely improve the performance under 0.1% and 1% labels. In the 20pts setting, we also employ the convolutional baseline, CloserLook [206], for a fair comparison. With no modification on the model, we surpass MIL-transformer [229] that additionally augments the backbone with transformer modules and multi-scale inference. Besides, we apply ERDA to baseline under fully-supervised setting and achieve competitive performance. These results also validate the ability of ERDA in providing effective supervision signals.

Results on SensatUrban. SensatUrban [71] is an urban-scale outdoor point cloud dataset that covers the landscape from three UK cities. In Tab. 4.7, ERDA surpasses SQN [226] under the same 0.1% setting as well as its fully-supervised baseline, and also largely improves under full supervision. It suggests that our method can be robust to different types of datasets and effectively exploits the limited annotations as well as the unlabeled points.

Generalizing to ShapeNet. Apart from real-world 3D scenes, synthetic 3D shapes, such as CAD models, are also important in 3D processing. For detailed shape analysis, 3D shapes are usually partitioned into parts, which leads to the task of part segmentation [58, 307]. ShapNet [307] comprises a large collection of 3D shapes ($> 16,000$) of 16 categories, each with 2-6 part annotations, which leads to a total of 50 classes. As in Tab. 4.4, we show consistent improvement on different baselines and settings. Especially compared with the improvement on instance mIoU, which averages the performance over 3D shapes, we acquire

	ER	DA	mIoU	ent.
baseline			59.8	-
+ pseudo-labels			63.3	2.49
	✓		65.1	2.26
+ ERDA		✓	66.1	2.44
	✓	✓	67.2	2.40

(A) **ERDA** improves the results and reduces the entropy (ent.), individually and jointly.

$L_{\text{DA}} \setminus \lambda$	0	1	2
-	-	65.1	66.3
$KL(\mathbf{p}\|\|\mathbf{q})$	66.1	67.2	66.6
$KL(\mathbf{q}\|\|\mathbf{p})$	66.1	65.9	65.2
JS	65.2	65.4	65.1
MSE	66.0	66.2	66.1

(B) **ER and DA** provide better results when taking $KL(\mathbf{p}||\mathbf{q})$ with $\lambda = 1$.

k	one-hot	soft	ERDA
64	63.1	62.8	64.5
500	63.0	62.3	64.1
1e3	63.3	63.5	65.5
1e4	63.0	62.9	65.6
dense	62.7	62.6	67.2

(C) **ERDA** consistently benefits the model with more pseudo-labels (k).

TABLE 4.10. Ablations on ERDA. If not specified, the model is RandLA-Net trained with ERDA as well as dense pseudo-labels on S3DIS under the 1% setting and reports in mIoU. Default settings are marked in gray.

larger gains on category mIoU, which is more sensitive to the per-category performance. It then hints that our dense and informative pseudo-labels could better assist the recognition of hard and minor classes.

4.4.3 Performance Comparison on 2D Segmentation

Semi-supervised results on Pascal. We follow FixMatch [227] in using DeepLabv3+ [31] with ResNet-101 [54] as baseline. As in Tab. 4.8, we show that ERDA brings consistent improvement from low to high data regime. It thus indicates the strong generalization of our method. We also see that the improvement is less significant than the 3D scenes. It might be because 2D data are more structured (*e.g.* pixels on a 2D grid) and are thus less noisy than the 3D point cloud from real-world scenes.

Sparse-label results on Pascal. We follow ReCo [273] to experiment with different annotation budgets in the form of pixel ratio in each image, which are similar the weakly-supervised settings for 3D point clouds. As shown in Tab. 4.5, ERDA surpasses the previous state-of-the-art method under the sparse-label setting. While noticeable improvements are achieved under all label ratios, ERDA is shown to be especially effective when given very few labels, such as the 1-pixel setting. Under confusing student output and strong data augmentations on 2D images, ERDA still learns an effective pseudo-labeling method, which could indicate the generalization of ERDA across modalities.

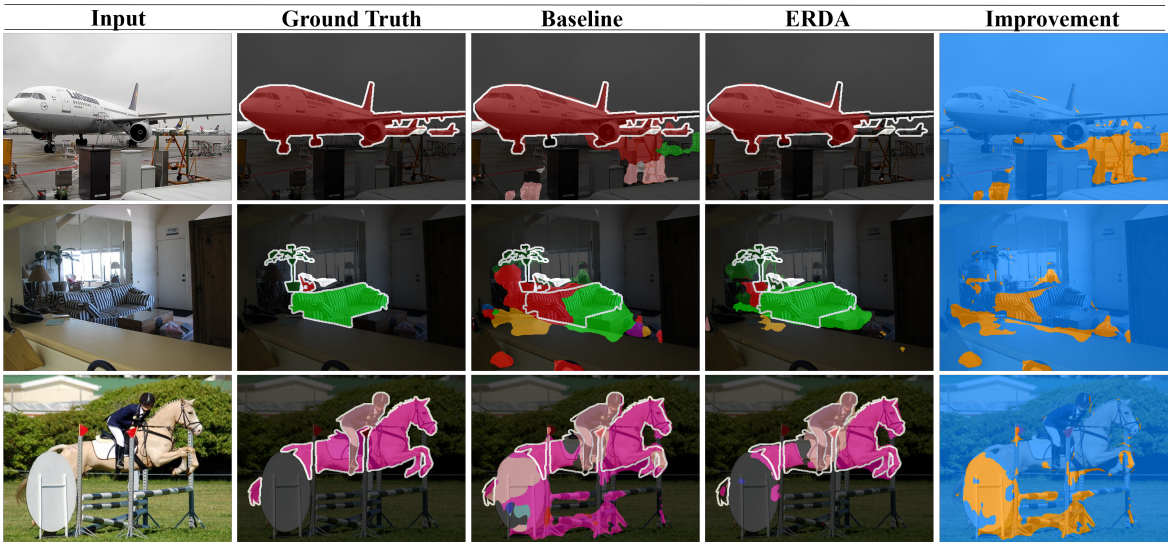


FIGURE 4.5. We show a clear benefit of our ERDA with query-based pseudo-labeling over baseline (FixMatch) on Pascal validation. Similar to 3D cases, ERDA provides cleaner predictions with better separations between different semantic groups, in both outdoor and indoor scenes.

As shown in Fig. 4.5, we further qualitatively compare ERDA with query-based pseudo-labels to our FixMatch baseline that utilizes strong augmentations. Under the the extremely scarce labels of the 1-pixel settings, we find ERDA could still strive to produce cleaner semantic groups with less noise. This could then imply that the model may still suffer from noise even if it has been trained to overcome the induced strong augmentations; and could also demonstrate the importance of our ERDA learning with query-based pseudo-labels to specifically address the noise within pseudo-labels.

Results on medical images. Following recent works [234, 304, 305] that analyze label-efficient method on medical images for practical use, we investigate ERDA on the ACDC [308] dataset. Specifically, medical datasets collect training data from different patients (cases). ACDC contains 70 cases for training and results are mainly reported in Dice Similarity Coefficient (DSC) averaged on 3 classes. As shown in Tab. 4.6, ERDA improves over the current state-of-the-art methods on all training scenarios with limited cases.

Generalizing to unsupervised segmentation. While semi-supervised and sparse-label settings still rely on few given labeled pixels, unsupervised segmentation aims to discover

PL	3D			2D			
	1pt	1%	10%	1-pixel	1%	5%	25%
sup.	40.6	59.8	61.7	60.3	66.2	69.2	73.8
proto	41.5	63.3	64.0	59.4	68.1	70.3	74.1
w2s	41.0	62.5	63.9	63.7	71.0	72.9	75.8
query	49.1	67.0	68.1	70.2	74.1	75.1	76.4

(A) **Query-based** pseudo-labels achieve the best generalizability.

g_p	1-pixel	1%	5%	25%
-	63.7	71.0	72.9	75.8
MLPs	64.5	72.0	73.5	75.6
query	70.2	74.1	75.1	76.4

(B) **Query-based** pseudo-labeling adapts better to the rich 2D augmentations.

g_p	1pt	1%	10%
-	2.2	65.8	66.0
MLPs	26.2	70.4	71.7
query	28.7	71.6	72.7

(C) **Query-based** pseudo-labeling further improves transformers on 3D data.

TABLE 4.11. Ablations on ERDA with query-based pseudo-labels for cross-modality generalization. For the "PL" column, "sup." denotes the supervised baseline, "proto" the prototypical pseudo-labels, "w2s" the weak-to-strong pseudo-labels, and "query" the proposed query-based pseudo-labels. If not specified, the models follow the settings in Tab. 4.10 and Tab. 4.5 on 3D and 2D modalities and report in mIoU.

semantically meaningful groups without any manual annotations [278, 279]. Recent advancement [284, 303] in unsupervised segmentation combines vision transformer [36, 37] with self-supervised training for better feature clustering and categories discovery, which usually utilize weak-to-strong pseudo-labels.

Since we have only unlabeled data points in the unsupervised setting, we could view the weights in the final classifier [283, 303] as the queries and incorporate it with our ERDA learning. As shown in Tab. 4.9, we gain clear improvements without bells and whistles. Such results further show that ERDA could provide semantically meaningful cues in its pseudo-labels to learn a more compact features and better clustering results for unsupervised segmentation.

4.4.4 Ablations and Analysis

We mainly consider the 1% setting and ablates in Tab. 4.10 to better investigate ERDA and make a more thorough comparison with the current pseudo-label generation paradigm. For more studies on hyper-parameters, please refer to the supplementary.

Individual effectiveness of ER and DA. To validate our initial hypothesis, we study the individual effectiveness of L_{ER} and L_{DA} in Tab. 4.10a. While the pseudo-labels essentially improve the baseline performance, we remove its label selection and one-hot conversion when adding the ER or DA term. We find that using ER alone can already be superior

to the common pseudo-labels and largely reduce the entropy of pseudo-labels (ent.) as expected, which verifies that the pseudo-labels are noisy, and reducing these noises could be beneficial. The improvement with the DA term alone is even more significant, indicating that a large discrepancy is indeed existing between the pseudo-labels and model prediction and is hindering the model training. Lastly, by combining the two terms, we obtain the ERDA that reaches the best performance but with the entropy of its pseudo-labels larger than ER only and smaller than DA only. It thus also verifies that the DA term could be biased to uniform distribution and that the ER term is necessary.

Different choices of ER and DA. Aside from the analysis in Sec. 4.3.3, we empirically compare the results under different choices of distance for L_{DA} and λ for L_{ER}. As in Tab. 4.10b, the outstanding result justifies the choice of $KL(\mathbf{q}||\mathbf{p})$ with $\lambda = 1$. Additionally, all different choices and combinations of ER and DA terms improve over the common pseudo-labels (63.3), which also validates the general motivation for ERDA.

Ablating label selection. We explore in more detail how the model performance is influenced by the amount of exploitation on unlabeled points, as ERDA learning aims to enable full utilization of the unlabeled points. In particular, we consider three pseudo-labels types: common one-hot pseudo-labels, soft pseudo-labels ($\mathbf{p}$), and soft pseudo-labels with ERDA learning. To reduce the number of pseudo-labels, we select sparse but high-confidence pseudo-labels following a common top-k strategy [225, 244] with various values of k to study its influence. As in Tab. 4.10c, ERDA learning significantly improves the model performance under all cases, enables the model to consistently benefit from more pseudo-labels, and thus neglects the need for label selection such as top-k strategy, as also revealed in Fig. 4.1. Besides, using soft pseudo-labels alone can not improve but generally hinders the model performance, as one-hot conversion may also reduce the noise in pseudo-labels, which is also not required with ERDA.

Effect on the training process. To analyze how ERDA helps with the model learning, we visualize the evolution of DA and ER terms along with the pseudo-labels as the training proceeds (on the training set) in Fig. 4.6. We observe that starting from a random stage, both the model and pseudo-labels could overfit on simple concepts such as walls in an early

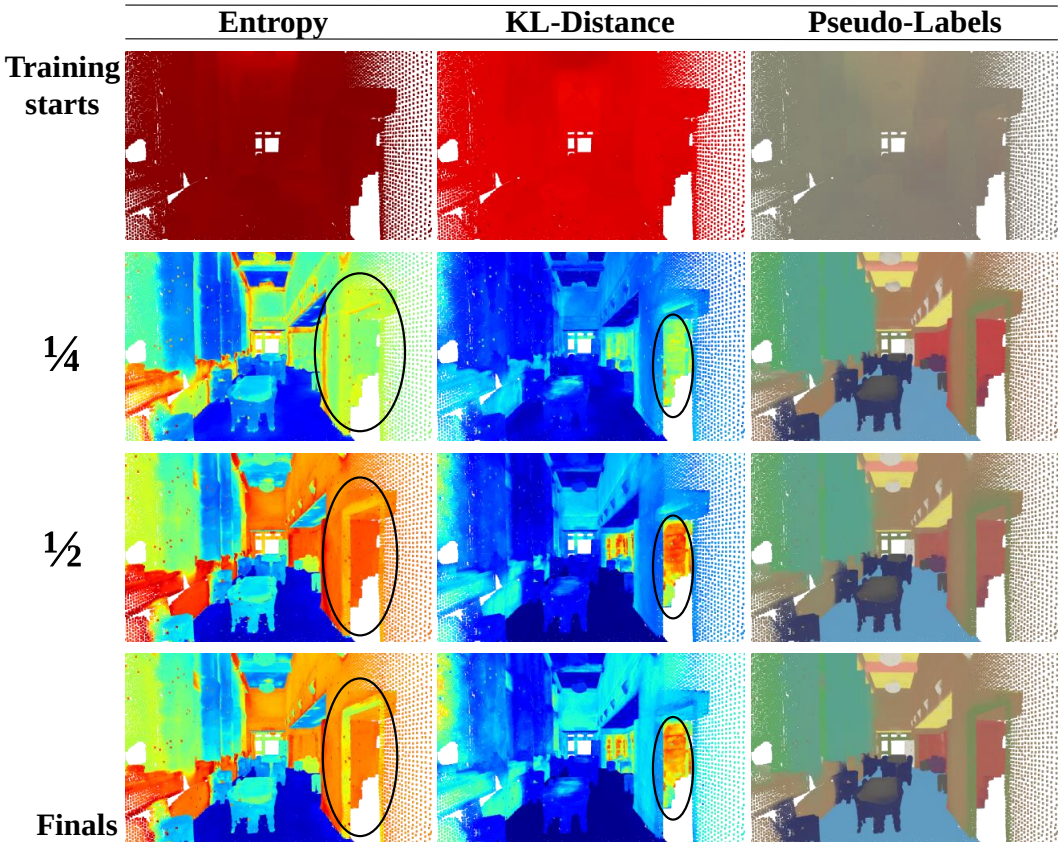


FIGURE 4.6. Visualization of ER and DA throughout the training process.

stage ($\frac{1}{4}$), which is denoted by the low entropy and KL-distance to the model prediction in most of the scenes. As the training proceeds, thanks to the noise-aware gradient of ERDA, pseudo-labels start to diverge from the model prediction and start to take up more complex concepts in the later stage ($\frac{1}{2}$), which is shown by an increased entropy and KL-distance in the area of the bookcase. It finally stabilizes with the ability to offer estimation with different levels of uncertainty of the semantic classes, which is denoted by the clearer edges across boundaries in the entropy map, such as the separation of the door and wall. Additionally, in the final stage, the KL-Distance to the model predictions is still relatively high in complex and cluttered areas, which can indicate that the pseudo-labels could capture some different semantic cues that guide and regularize the model learning. More analysis for the effect of ERDA on training dynamics are also provided in Sec. 4.6.4.

Modality-agnostic ability of query-based pseudo-label. To study the generalizability of query-based pseudo-labels across modalities, we compare it to the existing commonly used

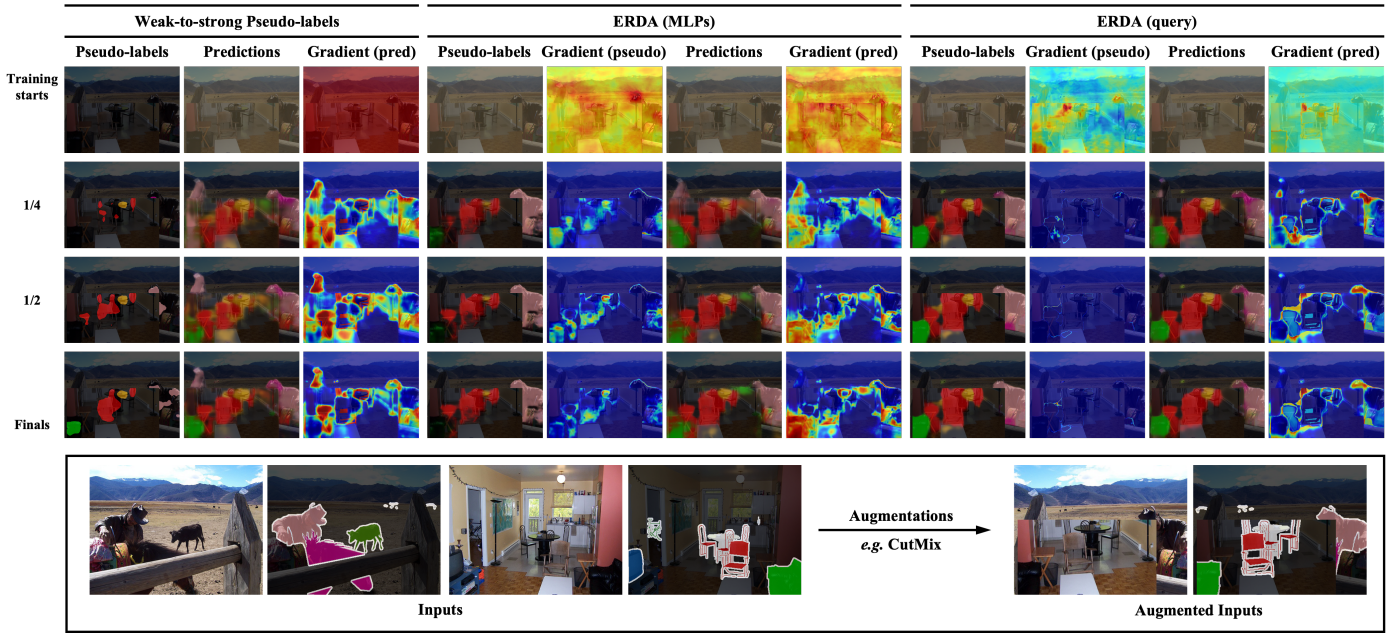


FIGURE 4.7. We show a clear benefit of our ERDA with query-based pseudo-labeling under strong augmentations [274, 275], where it produce consistent pseudo-labels with rich semantics to guide the student models towards clean and complete segmentations. Best viewed in color and zoom-in.

pseudo-labels on both 3D and 2D modalities, as in Tab. 4.11. We show that, though the prototypical pseudo-labels are effective on 3D point cloud, it could however degenerate the model performance on 2D images, especially when given limited annotations such as the 1-pixel setting, as it can hardly benefit from the rich 2D augmentations. Similarly, compared to prototypical pseudo-labels, the weak-to-strong pseudo-labels are sub-optimal for 3D point cloud, which might be due to the lack of effective 3D data augmentation techniques. Such empirical results could also reflect the motivation and challenge of achieving a more modality-agnostic pseudo-labeling method. In contrast, the proposed query-based pseudo-labels gain notable improvements on both 3D and 2D modalities, which indicates that the problem of the noise within pseudo-labels is indeed shared across modalities. We also notice that query-based pseudo-labels improve the most under low-data regime for both modalities, further demonstrating its effectiveness across modalities.

The effectiveness of query-based pseudo-labels on 2D. As the query-based pseudo-labels are primarily proposed to handle the noise brought by rich augmentations during the data processing on 2D images, we experiment with other choices of the projection g_p in Eq. (4.8).

As shown in Tab. 4.11b, we consider MLPs, as a typical replacement of transformer decoder, to directly apply ERDA learning on the weak-to-strong pseudo-labels. This direct application yields only a marginal gain over the baseline in the 1-pixel setting (+0.8 mIoU), in stark contrast to ERDA applied on query-based pseudo-labels (+6.5 mIoU). With more ground-truth supervision, such as the 25% setting, it can even hurt the performance (-0.2 mIoU). This observation thus demonstrates the negative effect of such weak-to-strong augmentations on the pseudo-label learning methods, including ERDA learning, which are attracted by the dominant gaps from $\mathcal{T}^{s}$ and overlook the noise within pseudo-labels, thus not fully focusing on improving the quality of pseudo-labels. On the contrary, with query-based pseudo-labels, the potential of ERDA could be fully released and bring clear improvements to the model.

The effectiveness of query-based pseudo-labels on 3D. While query-based pseudo-labels have shown clear benefits for label-efficient segmentation on 2D images, we propose that it could also further improve the 3D counterparts. Since data augmentations have been shown as an effective way to alleviate the lack of training data and avoid overfitting [274–276] in training large models such as transformers [52, 301], we then explore the potential of the proposed query-based pseudo-labels on recent 3D transformer models (PT) that are more prone to overfitting, as also shown in Sec. 4.4.2. As shown in Tab. 4.11c, ERDA learning with query-based pseudo-labels further improves over a direct application of ERDA learning with prototypical pseudo-labels (denoted as MLPs). These promising results could suggest that the proposed query-based pseudo-labeling does not hamper the overall generalizability of our method. However, we also notice that, ERDA can also be effective and performant when strong augmentations are absent and can match the performance of query-based pseudo-labels on other convolutional baselines. We consider the effectiveness of the pseudo-labeling methods can be limited by the capability of the backbones and provide more analysis in Sec. 4.6.3.

The effect of query-based pseudo-labels under augmentations. We further study the negative effects of strong augmentations on the learning of pseudo-labeling, which motivates the proposed query-based pseudo-labels.

In Fig. 4.7, we compare the training statistics of common weak-to-strong pseudo-labels, a direct application of ERDA, and the ERDA with query-based pseudo-labels. We notice that the weak-to-strong pseudo-labels can be sparse and irregular, leading to significant gradients w.r.t. student predictions even at the final stage of the training. We hypothesize that this can be the result of the use of label selection, where the spatial distribution of high-confidence predictions can vary as the training proceeds.

While a direct application of learning-based pseudo-labels improves the amount of effective pseudo-labels, we notice it can also be unstable during the model training, as indicated by the high gradients w.r.t. pseudo-labels themselves. Even though ERDA can avoid the noisy predictions from the student models, we find that the excessive use of augmentations makes the prediction of student model very noisy, especially at the segmentation boundaries. As a result, ERDA can only learn from the majority classes that are more stable under the strong augmentations, such as people and background class. It thus fails to pick up the features for all semantic classes and produces pseudo-labels only representing the majority class. As shown in the gradient for student model, such a naive application of learning-based pseudo-labels can even prevent the student model from predicting other minor classes, *e.g.* producing large gradients w.r.t. student prediction to suppress its prediction on "sofa".

These observations motivate our design of query-based pseudo-labels, where we introduce dedicated query embeddings to account for all semantic classes. As a result, query-based ERDA quickly stabilizes pseudo-labels and, more importantly, preserves the rich semantic cues for student model training. We provide

4.5 Summary

In this chapter, we approach the modality-agnostic label-efficient segmentation, which imposes the challenge of insufficient supervision signals for training and the various data processing techniques across modalities. Though pseudo-labels are widely used, label selection is commonly employed to overcome the noise, but it also prevents the full utilization of unlabeled data. By addressing this, we propose a new learning scheme on pseudo-labels,

ERDA, that reduces the noise, aligns to the model prediction, and thus enables comprehensive exploitation of unlabeled data for effective training. Moreover, as the problem of noise within pseudo-labels is shared across both 2D and 3D data, we further propose query-based pseudo-labels to overcome the modality-specific data processing and augmentations, leading to a more modality-agnostic pseudo-labeling method. We demonstrate that ERDA reduces to the deceptively simple cross-entropy-based loss, which leads to straightforward implementation and easy adaptation on various backbone models with little training overhead. Experimental results show that ERDA outperforms previous methods in various settings and datasets of both 3D and 2D modalities. Notably, it surpasses its fully-supervised baselines and can be further generalized to medical images and unsupervised settings.

Limitation. Despite promising results, our method, like other label-efficient approaches, assumes complete coverage of semantic classes in the available labels, which may not always hold in real-world cases. Combining label-efficient method with recent large foundation models for open-world scenarios could be explored as an important future direction.

4.6 Appendix

In this supplementary material, we provide more details regarding implementation details and discussion on augmentations in Sec. 4.6.1, more analysis of ERDA in Sec. 4.6.2, full experimental results in Sec. 4.6.3, studies on parameters in Sec. 4.6.4, and more visualization in Sec. 4.6.5.

4.6.1 Implementation and Pseudo-labeling Details

Baselines details. For the RandLA-Net [109] and CloserLook3D [206] baselines, we follow the instructions in their released code for training and evaluation, which are here (RandLA-Net) and here (CloserLook3D), respectively. Especially, in CloserLook3D[206], there are several local aggregation operations and we use the "Pseudo Grid" (KPConv-like) one, which provides a neat re-implementation of the popular KPConv [22] network (rigid version). For point transformer (PT) [26], we follow their paper and the instructions in the code base that

claims to have the official code (here). For FixMatch [227], we use the publicly available implementation here.

While the FixMatch baseline [227] is originally proposed for object recognition, several works [234, 273] that adapts the original baseline to the tasks of semi-supervised and weakly supervised image segmentations. We follow these implementations and compare against them in Tabs. 4.5 and 4.6. We discuss the training and pseudo-labeling details in the following paragraphs.

Our code and pre-trained models will be released.

Augmentations in training. We provide detailed setup of the typical augmentations employed in 3D point cloud and 2D images.

For 3D point cloud data, the augmentations $\mathcal{T}$ are mainly perturbations on point positions and augmentations on color space. For 2D data, the weak augmentations $\mathcal{T}^{\text{w}}$ are the basic resize-and-crop with minimal chromatic and spatial augmentations, and the strong augmentations incorporate more aggressive perturbations on both structure, texture, and semantics, such as mixups [265, 274, 275].

For example, Fig. 4.3 demonstrates a typical illustration of weak and strong 2D augmentations, where the strong augmentations mix up two images into one, significantly altering the input of the student network. In practice, image crops for mixing come from different images, and we use the same image for mixups only for a cleaner demonstration.

For clarity, we further provide typical samples of these augmentations in Listing 1 for 3D point cloud data, Listing 2 for weak augmentations on 2D data, and Listing 3 for strong augmentations on 2D data.

Pseudo-labeling with augmentations. We also provide details on how the pseudo-labeling methods are employed for training under different levels of augmentations, such as the weak-to-strong pseudo-labels.

Especially, we discuss the application of spatial augmentations on 2D images, which apply potentially different spatial augmentations that can lead to spatial misalignment.

As outlined above, the weak augmentations $\mathcal{T}^{\mathrm{w}}$ and strong augmentations $\mathcal{T}^{\mathrm{s}}$ indeed share the basic spatial augmentation of resize-and-crop, thus still sharing the same spatial content and ground-truth mask at this stage.

However, while further spatial augmentations in $\mathcal{T}^{\mathrm{s}}$ would result in different ground-truth masks for the student and teacher network, we would like to note that this spatial transform is tractable. The ground-truth masks, together with the generated masks of pseudo-labels, would then be transformed to align with the student network.

In particular, we provide pseudo-codes in Listing 4 to demonstrate the overview training procedures with the pseudo-labeling methods, including the alignment process, the application of various augmentations, and the proposed ERDA learning.

As denoted in Listing 4, though p and q are spatially aligned for the loss calculation, the p is inferred on the cleaner input images with weak augmentations $\mathcal{T}^{\mathrm{w}}$ while q is inferred on a much noiser image with strong augmentations $\mathcal{T}^{\mathrm{s}}$. Therefore, ERDA would notice a significant gap between the p and q, not because there is noise within pseudo-labels, but because this gap is intentionally introduced via the T^{s}.

As a result, we argue that the augmentations cause large gaps between the pseudo-labels p and the student predictions q, in addition to the noise in pseudo-labels p. It thus prevents a direct and easy application of ERDA on 2D images, and leads to our proposed query-based pseudo-labeling that accommodates the existence of such noise from augmentations.

4.6.2 Delving into ERDA with More Analysis

Following the discussion in Sec. 4.3, we study the impact of entropy regularization as well as different distance measurements from the perspective of gradient updates.

Algorithm 1 Typical sample of augmentations on 3D point cloud.

```
transform=[
  RandomDropout(dropout_ratio=0.2),                    # random dropout
   on points
  RandomRotate(angle=[-1, 1], axis="z", p=0.5),        # random rotation
    (+/- pi) - around z-axis
  RandomRotate(angle=[-1/64, 1/64], axis="x", p=0.5), # random rotation
    (+/- pi/64) - around x-axis
  RandomRotate(angle=[-1/64, 1/64], axis="y", p=0.5), # random rotation
    (+/- pi/64) - around y-axis
  RandomScale(scale=[0.9, 1.1]),                       # random scaling
   on each axis (+/-0.1)
  RandomFlip(p=0.5),                                   # random flipping
   , along xy-axes only
  RandomJitter(sigma=0.005, clip=0.02),                # random Gaussian
    noise on per-point xyz
  ChromaticAutoContrast(p=0.2),                        # enhance
   contrast
  ChromaticTranslation(p=0.95, ratio=0.05),            # random uniform
   shift on rgb
  ChromaticJitter(p=0.95, std=0.05),                   # random Gaussian
    noise on per-point rgb
]
```

Algorithm 2 Typical weak augmentations on 2D data.

```
transform = [
    Resize(ratio_range=(0.5, 2.0)),    # random resize
    Crop(size=(321, 321)),             # crop to desired size
    Hflip(p=0.5),                      # horizontal flipping
]
```

Algorithm 3 Typical strong augmentations on 2D data.

```
transform = [
    Resize(ratio_range=(0.5, 2.0)),    # random resize
    Crop(size=(321, 321)),             # crop to desired size
    Hflip(p=0.5),                      # horizontal flipping
    # - strong augmentation:
    ColorJitter(0.5, 0.5, 0.5, 0.25), # largely change brightness,
   contrast, saturation and hue
    RandomGrayscale(p=0.2),            # convert to gray-scale image
    Blur(p=0.5),                       # add Gaussian blurs
    CutMix(p=0.5),                     # cutout & mixup with crops from
   other images
]
```

Algorithm 4 Overview of weak-to-strong pseudo-labeling approach under spatial augmentations.

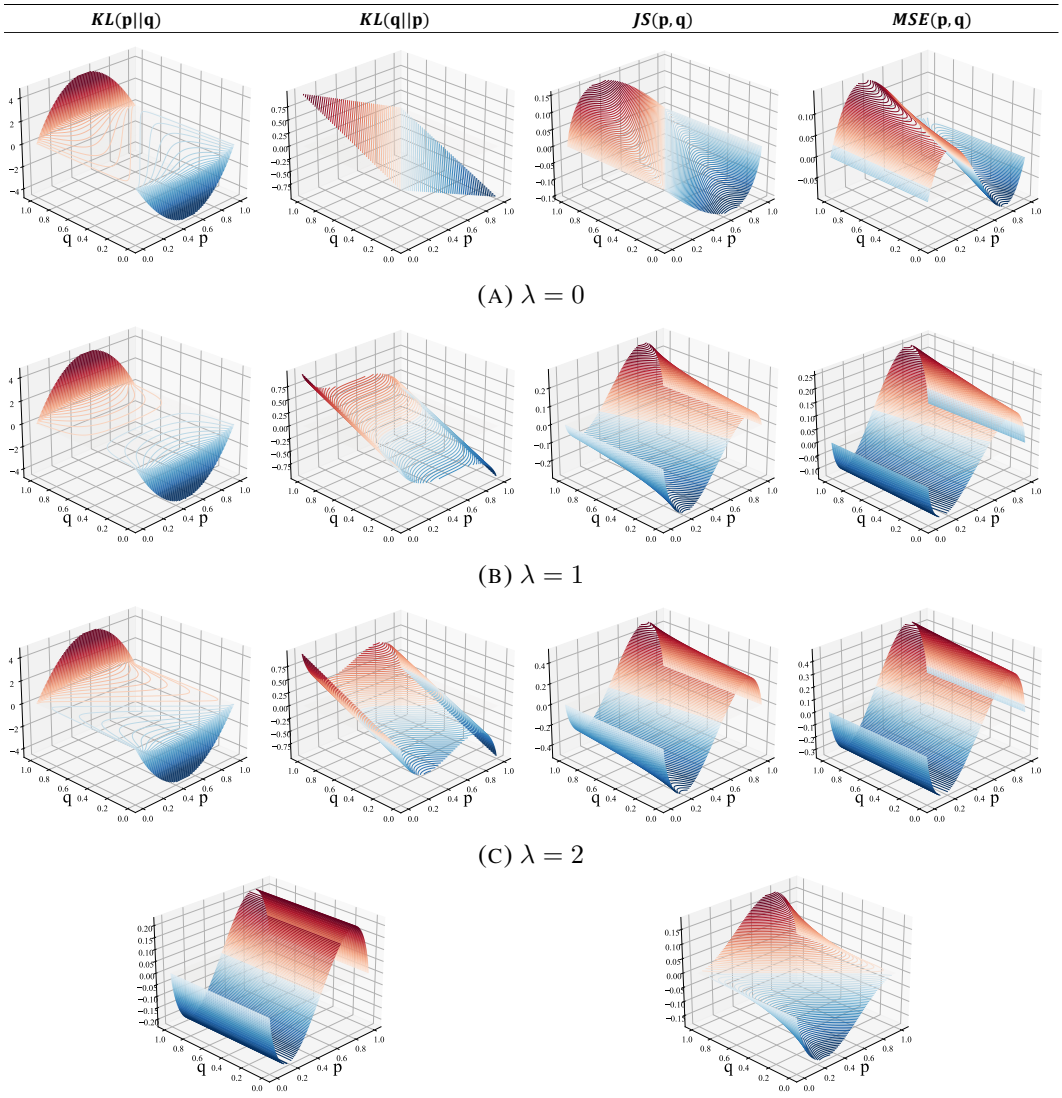
```
for img, gt_mask in data_loader:
  # img: [B, 3, H, W]
  # gt_mask: [B, H, W], filled with invalid values for unlabeled images

  # - transforms sampled and recorded for this batch
  T_w, T_s, T_spatial = get_transforms()

  # - inference
  logits = student(T_s(img))        # logits predicted by student, with
   strong-aug
  pseudo_mask = teacher(T_w(img))   # pseudo-labels predicted by teacher
   , with weak-aug

  # - alignemnt
  gt_mask = T_spatial(gt_mask)
  pseudo_mask = T_spatial(pseudo_mask)

  # - loss calculation
  L_ce = cross_entropy(logits, gt_mask)
  L_p = H(logits, pseudo_mask)  # Eq. (5)
  L = L_ce + alpha * L_p        # Eq. (9)

  L.backward()
```

`T_w`: weak augmentation; `T_s`: strong augmentation; `T_spatial`: the recorded spatial transforms in `T_s`; ~~`alpha`: α in Eq. (4.9)~~

L_{DA}	$KL(\mathbf{p}\|\|\mathbf{q})$	$KL(\mathbf{q}\|\|\mathbf{p})$	$JS(\mathbf{p},\mathbf{q})$	$MSE(\mathbf{p},\mathbf{q})$
L_p	$H(\mathbf{p},\mathbf{q})-(1-\lambda)H(\mathbf{p})$	$H(\mathbf{q},\mathbf{p})-H(\mathbf{q})+\lambda H(\mathbf{p})$	$H(\frac{\mathbf{p}+\mathbf{q}}{2})-(\frac{1}{2}-\lambda)H(\mathbf{p})-\frac{1}{2}H(\mathbf{q})$	$\frac{1}{2}\sum_i(p_i-q_i)^2+\lambda H(\mathbf{p})$
g_i	$p_i\sum_j p_j(-\log\frac{q_i}{q_j}+(1-\lambda)\log\frac{p_i}{p_j})$	$p_i-q_i-\lambda p_i\sum_j p_j\log\frac{p_i}{p_j}$	$p_i\sum_j p_j(\frac{-1}{2}\log\frac{p_i+q_i}{p_j+q_j}+(\frac{1}{2}-\lambda)\log\frac{p_i}{p_j})$	$p_i(p_i-q_i)-p_i\sum_j p_j(p_j-q_j)-\lambda p_i\sum_j p_j\log\frac{p_i}{p_j}$
Situations	$\Delta=-g_i$			
$p_k\to 1$	0	$q_i-\mathbb{1}_{k=i}$	0	0
$q_1=\ldots=q_K$	$(\lambda-1)p_i\sum_j p_j\log\frac{p_i}{p_j}$	$\frac{1}{K}-p_i+\lambda p_i\sum_j p_j\log\frac{p_i}{p_j}$	$p_i\sum_{j\neq i}p_j(\frac{1}{2}\log\frac{Kp_i+1}{Kp_j+1}+(\lambda-\frac{1}{2})\log\frac{p_i}{p_j})$	$-p_i^2+p_i\sum_j p_j^2+\lambda p_i\sum_j p_j\log\frac{p_i}{p_j}$
$q_i\to 1$	$+\inf$	$1-p_i+\lambda p_i\sum_j p_j\log\frac{p_i}{p_j}$	$p_i\sum_{j\neq i}p_j(\frac{1}{2}\log\frac{p_i+1}{p_j}+(\lambda-\frac{1}{2})\log\frac{p_i}{p_j})$	$-p_i^2+p_i(1-p_i)+p_i\sum_j p_j^2+\lambda p_i\sum_j p_j\log\frac{p_i}{p_j}$
$q_{k\neq i}\to 1$	$-\inf$	$-p_i+\lambda p_i\sum_j p_j\log\frac{p_i}{p_j}$	$p_i\sum_{j\neq i}p_j(\frac{1}{2}\log\frac{p_i}{p_j+\mathbb{1}_{j=k}}+(\lambda-\frac{1}{2})\log\frac{p_i}{p_j})$	$-p_i^2-p_ip_k+p_i\sum_j p_j^2+\lambda p_i\sum_j p_j\log\frac{p_i}{p_j}$

TABLE 4.12. The formulation of L_p using different functions to formulate L_{DA}. We present the gradients $g_i=\frac{\partial L_p}{\partial s_i}$, and the corresponding update $\Delta=-g_i$ under different situations. Analysis can be found in the Sec. 4.3.3 and Sec. 4.6.2.

In particular, we study the gradient on the score of the i-th class *i.e.* s_i, and denote it as $g_i=\frac{\partial L_p}{\partial s_i}$. Given that $\frac{\partial p_j}{\partial s_i}=\mathbb{1}_{(i=j)}p_i-p_ip_j$, we have $g_i=p_i\sum_j p_j(\frac{\partial L_p}{\partial p_i}-\frac{\partial L_p}{\partial p_j})$. As shown in Tab. 4.12, we demonstrate the gradient update $\Delta=-g_i$ under different situations.



(A) $\lambda = 0$

(B) $\lambda = 1$

(C) $\lambda = 2$

(D) Gradient updates of L_{ER}

(E) Gradient updates of $JS(\mathbf{p}, \mathbf{q})$ with $\lambda = 0.5$

FIGURE 4.8. Contour visualization of the gradient update with binary classes for better understanding. For a clearer view, we use red for positive updates and blue for negative updates, the darker indicates larger absolute values and the lighter indicates smaller absolute values.

In addition to the analysis in Sec. 4.3.3, we find that, when $\mathbf{q}$ is certain, *i.e.* $\mathbf{q}$ approaching a one-hot vector, the update of our choice $KL(\mathbf{p}||\mathbf{q})$ would approach the infinity. We note that this could be hardly encountered since $\mathbf{q}$ is typically also the output of a softmax function. Instead, we would rather regard it as a benefit because it would generate effective supervision on those model predictions with high certainty, and the problem of gradient explosion could also be prevented by common operations such as gradient clipping.

In Fig. 4.8, we provide visualization for a more intuitive understanding on the impact of different formulations for L_{DA} as well as their combination with L_{ER}. Specifically, we consider a simplified case of binary classification and visualize their gradient updates when λ takes different values. We also visualize the gradient updates of L_{ER}. By comparing the gradient updates, we observe that only $KL(\mathbf{p}||\mathbf{q})$ with $\lambda = 1$ can achieve small updates when $\mathbf{q}$ is close to uniform ($q = \frac{1}{2}$ under the binary case), and that there is a close-0 plateau as indicated by the sparse contour lines.

When $\lambda = 0$, from the visualization, we see that all measurements would tend to have positive updates when $q_i > p_i$ and negative updates when $q_i < p_i$, which align with our intuition that the distribution alignment may be biased to align the $\mathbf{p}$ to a uniform distribution. Such intuition could also be revealed by the (negative) entropy term in the raw measurements of $KL(\mathbf{p}||\mathbf{q})$ and $JS(\mathbf{p}, \mathbf{q})$ as in Tab. 4.12.

Additionally, we also find that, when increasing the λ, all distances, except the $KL(\mathbf{p}||\mathbf{q})$, are modulated to be similar to the updates of having L_{ER} alone; whereas $KL(\mathbf{p}||\mathbf{q})$ can still produce effective updates, which may indicate that $KL(\mathbf{p}||\mathbf{q})$ is more robust to the λ.

Besides, we find that $\lambda = \frac{1}{2}$ is a special case for $JS(\mathbf{p}, \mathbf{q})$, which could help it overcome the bias by removing the negative entropy term, as shown in Tab. 4.12 and visualized in Fig. 4.8e. Experimentally, it achieves a performance of 66.2 in mIoU, which surpasses its performance with other choices of λ as in Tab. 4.10b. This also demonstrates the benefits of mitigating the entropy term in distance measurements. Nonetheless, it is still sub-optimal compared with our ERDA.

4.6.3 More Results

Full results. We provide full results for the experiments reported in the main chapter. For S3DIS [44], we provide the results of S3DIS with 6-fold cross-validation in Tab. 4.13 and its full results in Tab. 4.17. For ScanNet [43] and SensatUrban [71], we evaluate on their online test servers, which are here and here, and provide the full results in Tab. 4.18 and Tab. 4.19,

respectively. For Pascal [235], we provide the full results of different settings in Tab. 4.20. For Cityscapes [222], we provide the full reuslts of our method in Tab. 4.21.

Query-based pseudo-labeling with 3D convolutional baselines.

4.6.4 Further Ablations and Analysis

On prototypical pseudo-labels. We first study the hyper-parameters involved in the implementation of ERDA with the typical prototypical pseudo-label generation, including loss weight α, momentum coefficient m, and the use of projection network. As shown in Tab. 4.15, the proposed method acquires decent performance (mIoUs are all > 65 and mostly > 66) in a wide range of different hyper-parameter settings, compared with its fully-supervised baseline (64.7 mIoU) and previous state-of-the-art performance (65.3 mIoU by HybridCR [228]).

Additionally, we suggest that the projection network could be effective in facilitating the ERDA learning, which can be learned to specialize in the pseudo-label generation task. On the contrary, without the projection network, it may be too demanding to optimize a shared feature with drastically different gradients and expect it to be optimal on both segmentation and pseudo-label generation tasks. This could also be related to the advances in contrastive learning. Many works [251, 309, 310] suggest that a further projection on feature representation can largely boost the performance because such projection decouples the learned features from the pretext task. We share a similar motivation in decoupling the features for ERDA learning on the pseudo-label generation task from the features for the segmentation task.

To validate this, we sample and evaluate the statistical properties of the projection network that is optimized by ERDA. Specifically, for a trained network (RandLA-Net + ERDA), we sample for more than 10^7 points (~ 1000 cloud samples) to calculate the mean and the standard deviation for the backbone features, *i.e.* representation by $f(\mathbf{x})$, and projection network features, *i.e.* projection by $g \circ f(\mathbf{x})$, as shown in Fig. 4.9. The drastically different statistics of these two types of features could indicate that they are decoupled due to the projection network.

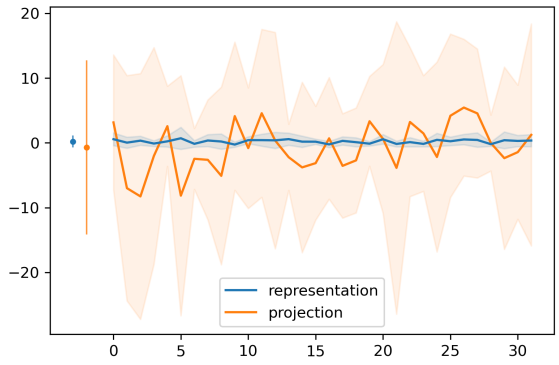


FIGURE 4.9. Statistical difference between projection features "projection" and backbone features "representation". The overall mean and standard deviation are shown as the dots and the vertical lines on the left, and the channel-wise mean and standard deviation are denoted as lines and shadings on the right.

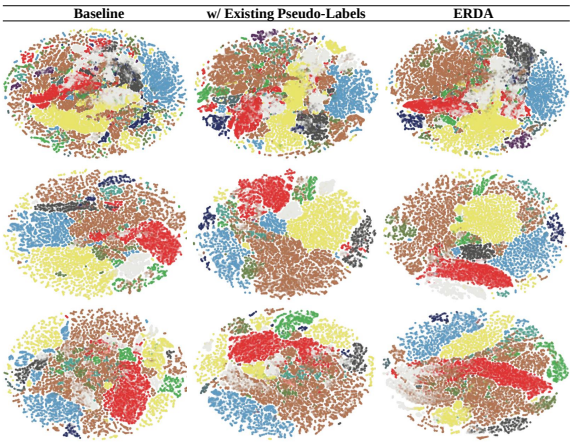


FIGURE 4.10. t-SNE visualization on different scenes sampled from S3DIS Area 5 with RandLA-Net as baseline. Visualizations in the same row share the same scene.

settings	methods	mIoU
Fully	PointNet [18]	47.6
	RandLA-Net [109]	70.0
	KPConv [22]	70.6
	HybridCR [228]	70.7
	PT [26]	73.5
	PointNeXt - XL [130]	74.9
	RandLA-Net + **ERDA**	71.0
	CloserLook3D + **ERDA**	73.7
	PT + **ERDA**	76.3
1%	zhang *et al.* [225]	65.9
	PSD [245]	68.0
	HybridCR [228]	69.2
	RandLA-Net + **ERDA**	69.4
	CloserLook3D + **ERDA**	72.3
	PT + **ERDA**	73.5

TABLE 4.13. Results on S3DIS 6-fold.

On query-based pseudo-labels. Furthermore, we study several important design choices for the implementation of our query-based pseudo-labeling method, which are the feature dimensions used for the transformer decoder, the number of transformer layers, and the number of attention heads for multi-head attention operations. As shown in Tab. 4.16, the

query-based pseudo labels demonstrate strong performance across a range of commonly used transformer settings. We note that, as in Tab. 4.16b, our method could petentially achieve even better results than the reported performance by stacking more transformer layers to the commonly used 6-layer transformer. Yet we opt for less layers to keep our method being simple and lightwight, while achieving comparable performance (both mIoUs > 70). These results indicate that the proposed query-based pseudo-labels could be an robust pseudo-labeling method under the optimization of ERDA learning.

On 3D pseudo-labeling with ERDA. Compared with 2D images, label-efficient methods on 3D data are underexplored. Therefore, we would like to expand the ablations of various pseudo-labels on 3D data to further study the challenges of 3D data from the perspective of pseudo-labeling. As shown in Tab. 4.14, we extend ERDA based on both the prototypical pseudo-labels and weak-to-strong pseudo-labels. Specifically, given weak-to-strong pseudo-labels, naive application of ERDA can even hurt the model performance, as the learning on pseudo-labels can compromise the benefit of the constructed strong augmentations without specially designed training recipes, such as iterative training [288, 290, 291, 293]. In comparison, query-based pseudo-labels can still improve based on the weak-to-strong labels, which is consistent with our motivation and their effectiveness on 2D images.

Moreover, in normal cases without strong augmentations, such as the use case of prototypical pseudo-labels, we find ERDA is already performant, making the query-based pseudo-labels hard to stand out. Such observation also indicates that the ERDA learning is an effective component for 3D label-efficient segmentation tasks.

Furthermore, we note that the weak-to-strong pseudo-labels lag behind the prototypical pseudo-labels, indicating that augmentations for 3D point cloud demands further study to serve as effective pseudo-labeling method, which can be an interesting future direction.

On the learning dynamics of ERDA. To better explore how pseudo-labels with ERDA affect the model training, we visualize the loss curves to demonstrate the learning dynamics of models under different pseudo-labels. As shown in Fig. 4.11, we find that ERDA can effectively regularize the model training to enjoy a smoother optimization on the intended

PL	g_p	RandLA-Net [109]			CloserLook [206]		
		1pt	1%	10%	1pt	1%	10%
sup.		40.6	59.8	61.7	34.6	59.9	55.5
proto	-	41.5	63.3	64.0	44.8	59.6	57.0
	ERDA (MLPs)	48.4	67.2	67.9	52.0	68.2	69.1
	ERDA (query)	49.1	67.0	68.1	52.5	68.0	68.9
w2s	-	41.0	62.5	63.9	48.2	63.2	64.7
	ERDA (MLPs)	39.7	53.2	53.6	50.1	61.3	66.7
	ERDA (query)	46.1	63.8	64.6	52.0	65.3	68.1

TABLE 4.14. More ablations of pseudo-labeling methods on 3D data. To better study different variants, we further decouple the pseudo-labeling methods and the application of ERDA. If not specified, the models follow the settings in Tab. 4.11 and report in mIoU.

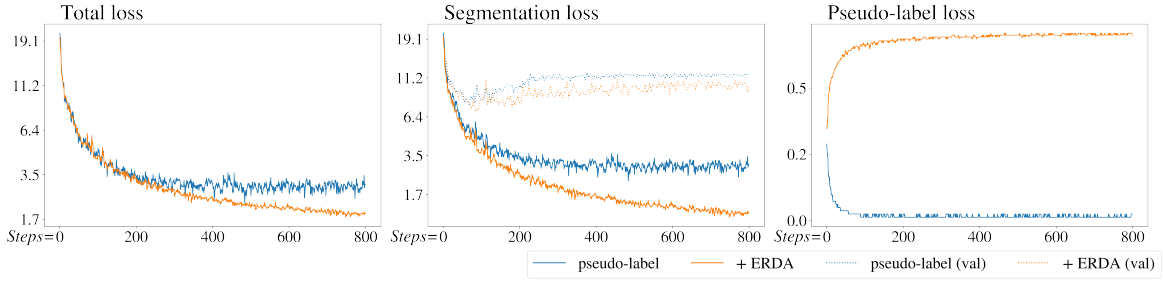


FIGURE 4.11. We visualize the loss curves, following the setting of Fig. 4.1c and Tab. 4.10. The total loss is the sum of segmentation loss and pseudo-label loss.

task of segmentation. We note that, model with existing pseudo-labels saturates on the optimization much earlier, roughly around the same time when the pseudo-label produces close-0 losses. We consider this as evidence of the underexplored unlabeled data by the existing pseudo-labels, where pseudo-labels also saturate and cannot provide effective signals to keep the model training forward. In comparison, we find ERDA maintains a higher and non-zero pseudo-label loss towards the end of model training. In this regard, ERDA learning stays effective in the model training. As ERDA performs noise-aware learning on the pseudo-labels, it can thus explore on all unlabeled data to provide effective signals throughout the model training.

4.6.5 Analysis with Visualizations

We provide more qualitative results in demonstrating the effectiveness of the proposed ERDA, together with the generated pseudo-labels.

m	mIoU
0.9	66.19
0.99	66.80
0.999	67.18
0.9999	66.22

(A) **Momentum update**.

projection	mIoU
-	65.90
linear	66.55
2-layer MLPs	67.18
3-layer MLPs	66.31

(B) **Projection network**.

α	mIoU
0.001	65.25
0.01	66.01
0.1	67.18
1	65.95

(C) **Loss weight**.

TABLE 4.15. Parameter study on ERDA. If not specified, the model is RandLA-Net with ERDA trained with loss weight $\alpha = 0.1$, momentum $m = 0.999$, and 2-layer MLPs as projection networks under 1% setting on S3DIS. Default settings are marked in gray.

feature dim.	mIoU
32	69.80
64	70.19
128	68.96

(A) **Feature dimensions**.

# layers	mIoU
1	70.19
3	69.75
6	70.25
12	69.96

(B) **Transformer layers**.

# heads	mIoU
1	69.88
4	70.16
8	70.19
16	69.93

(C) **Multi-heads**.

TABLE 4.16. Parameter study on ERDA with query-based pseudo-labels. If not specified, the model is FixMatch with ERDA trained with feature dimensions 64 for attention, 1-layer transformer, and 8-heads multi-head attention under 1-pixel setting on Pascal. Default settings are marked in gray.

Feature visualizations. In Fig. 4.10, we provide t-SNE visualization of the learned features when training the model with different strategies under the same 1% setting. An interesting observation is that, while using the existing pseudo-labels (one-hot labels with label selection) can already provide cleaner features than the baseline, ERDA leads the features of the same semantic class to be even closer together. In general, ERDA leads to more distinguishable and compact features for different semantic classes. This suggests that ERDA could facilitate the learning of more discriminative features from sparse ground-truth annotations.

Visualizations with pseudo-labels. We further plot the pseudo-labels by blending the color according to the class likelihood estimation. A simple example would be a binary 0-1 classification, where the class 0 uses the color of $[255, 0, 0]$ and class 1 $[0, 0, 255]$. Given a likelihood estimation of [0.3, 0.7], we blend its color to be $[255, 0, 0] * 0.3 + [0, 0, 255] * 0.7 = [77, 0, 178]$.

The visualization results include various scenes, including rooms and cluttered space (Fig. 4.12a), hallways (Fig. 4.12b), and offices (Fig. 4.12c). According to the presented figures, ERDA

assists the baseline with dense and informative pseudo-labels, and thus enables the model to capture more details and produce cleaner segmentation in different types of scenes.

Visualization on 2D images. Similar to visualization on 3D, we include various scenes for better investigation, ranging from indoor rooms (Fig. 4.13a), animals (Fig. 4.13a), to outdoor views (Fig. 4.13c).

settings	methods	mIoU	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
Fully	RandLA-Net + ERDA	71.0	94.0	96.1	83.7	59.2	48.3	62.7	73.6	65.6	78.6	71.5	66.8	65.4	57.9
	CloserLook3D + ERDA	73.7	94.1	93.6	85.8	65.5	50.2	58.7	79.2	71.8	79.6	74.8	73.0	72.0	59.5
	PT + ERDA	76.3	94.9	97.8	86.2	65.4	55.2	64.1	80.9	84.8	79.3	74.0	74.0	69.3	66.2
1%	RandLA-Net + ERDA	69.4	93.8	92.5	81.7	60.9	43.0	60.6	70.8	65.1	76.4	71.1	65.3	65.3	55.0
	CloserLook3D + ERDA	72.3	94.2	97.5	84.1	62.9	46.2	59.2	73.0	71.5	77.0	73.6	71.0	67.7	61.2
	PT + ERDA	73.5	94.9	97.7	85.3	66.7	53.2	60.9	80.8	69.2	78.4	73.3	67.7	65.9	62.1

TABLE 4.17. The full results of ERDA with different baselines on S3DIS 6-fold cross-validation.

settings	methods	mIoU	bathtub	bed	books.	cabinet	chair	counter	curtain	desk	door	floor	other	pic	fridge	shower	sink	sofa	table	toilet	wall	wndw
Fully	CloserLook3D + ERDA	70.4	75.9	76.2	77.0	68.2	84.3	48.1	81.3	62.1	61.4	94.7	52.7	19.9	57.1	88.0	75.9	79.9	64.7	89.2	84.2	66.6
20pts	CloserLook3D + ERDA	57.0	75.1	62.5	63.1	46.0	77.7	30.0	64.9	46.1	43.6	93.3	36.0	15.4	38.0	73.6	51.6	69.5	47.2	83.2	74.5	47.8
0.1%	RandLA-Net + ERDA	62.0	75.7	72.4	67.9	56.9	79.0	31.8	73.0	58.1	47.3	94.1	47.1	15.2	46.3	69.2	51.8	72.8	56.5	83.2	79.2	62.0
1%	RandLA-Net + ERDA	63.0	63.2	73.1	66.5	60.5	80.4	40.9	72.9	58.5	42.4	94.3	50.0	35.0	53.0	57.0	60.4	75.6	61.9	78.8	73.8	62.6

TABLE 4.18. The full results of ERDA with different baselines on ScanNet [43] test set, obtained from its online benchmark site by the time of submission.

settings	methods	mIoU	OA	Ground	Vegetation	Buildings	Walls	Bridge	Parking	Rail	Roads	Street Furniture	Cars	Footpath	Bikes	Water
Fully	RandLA-Net + ERDA	64.7	93.1	86.1	98.1	95.2	64.7	66.9	59.6	49.2	62.5	46.5	85.8	45.1	0.0	81.5
0.1%	RandLA-Net + ERDA	56.4	91.1	82.0	97.4	93.2	56.4	57.1	53.1	5.2	60.0	33.6	81.2	39.9	0.0	74.2

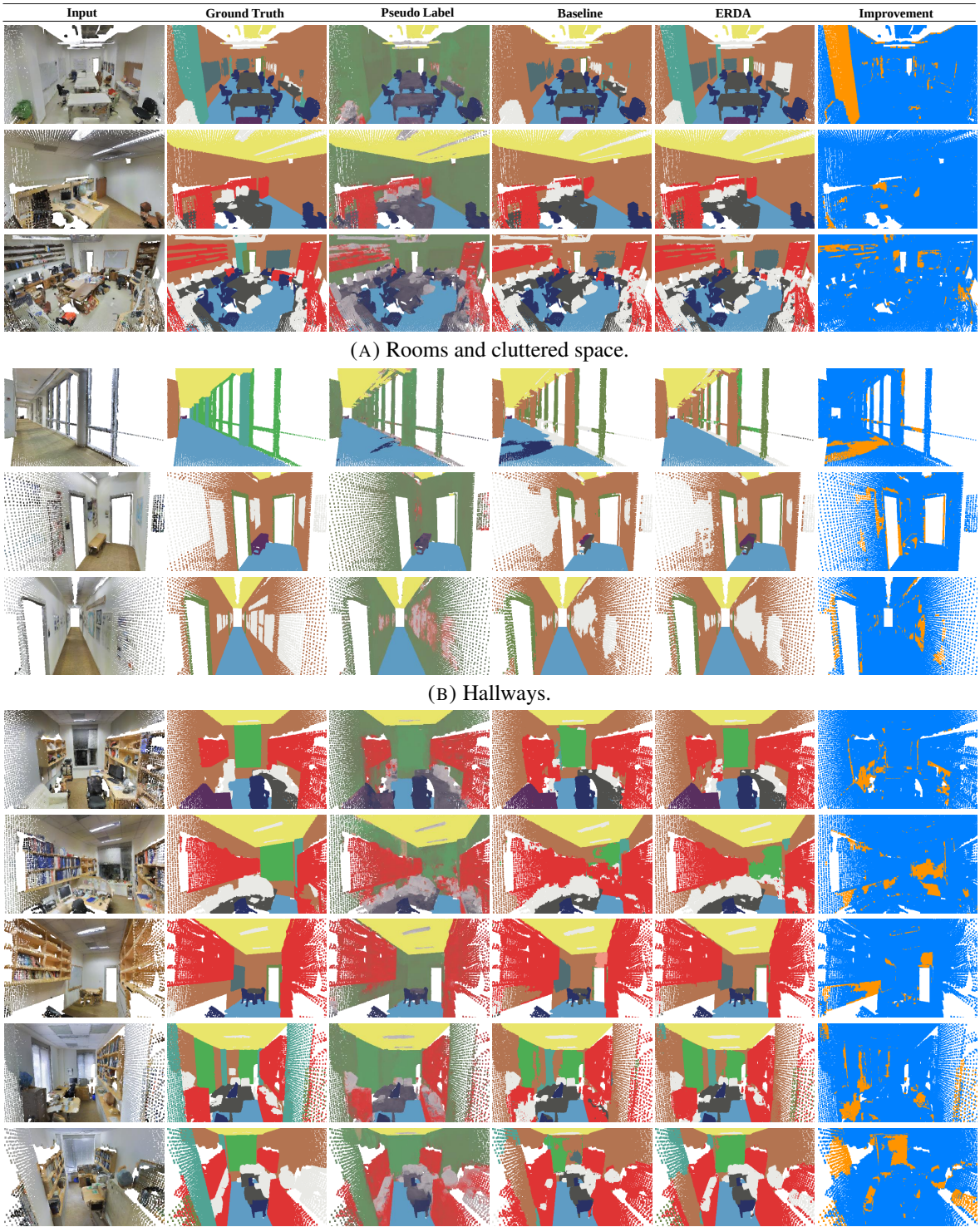
TABLE 4.19. The full results of ERDA with different baselines on SensatUrban [71] test set, obtained from its online benchmark site by the time of submission.

settings	mIoU	aero.	bicycle	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	monitor	backgrnd
1-pixel	70.2	91.7	81.1	39.6	78.7	62.3	66.8	88.0	79.9	85.9	28.8	84.7	50.2	83.1	81.4	74.9	79.4	51.1	80.5	41.9	80.3	63.5
1%	74.1	92.8	85.8	38.4	88.3	66.9	75.8	92.6	85.2	89.4	36.2	85.6	51.8	85.2	84.5	79.2	82.0	53.8	87.2	42.6	81.5	70.9
5%	75.1	93.4	88.0	37.2	86.3	67.1	78.5	92.7	87.6	91.4	33.2	88.3	48.8	87.8	86.0	79.9	82.8	56.1	87.1	47.5	86.9	69.7
25%	76.4	93.9	90.1	40.2	89.3	69.8	79.3	93.5	87.2	91.7	36.4	88.1	49.8	87.5	86.9	81.4	83.9	54.8	86.0	54.4	85.3	74.8
92	74.9	93.1	86.9	59.6	87.5	67.7	72.0	92.1	86.9	91.5	22.4	89.5	58.8	86.4	87.4	78.9	79.3	58.1	88.9	39.9	83.2	62.8
183	76.6	94.0	88.4	67.3	88.7	64.7	76.3	90.7	86.0	90.8	28.4	91.8	63.7	86.6	88.2	83.2	82.3	51.9	89.3	47.8	85.3	63.2
366	78.2	94.5	91.4	70.6	89.4	72.7	77.5	93.9	86.3	93.5	28.9	91.8	63.3	88.1	88.8	81.7	84.6	52.2	90.1	43.9	87.7	71.1
732	78.7	94.8	90.5	69.2	91.6	74.6	79.3	93.9	86.6	94.4	29.1	91.3	54.1	89.7	90.8	84.5	85.9	56.6	87.9	51.8	86.8	68.9
1464	80.4	95.3	90.8	68.6	90.9	74.7	80.8	93.9	88.7	92.9	33.1	93.6	63.7	89.8	90.6	85.3	87.4	63.1	87.7	52.6	88.7	75.6

TABLE 4.20. The full results of ERDA under different settings on Pascal [235] validation set. The methods are FixMatch + ERDA.

settings	mIoU	OA	road	sidewlk	parking	track	building	wall	fence	guard	bridge	tunnel	pole	polegroup	light	sign	veg.	terrain	sky	person	rider	car	truck	bus	caravan	trailer	train	motor	bike
Unsup	20.5	82.3	88.4	23.5	0.1	0.1	63.7	26.8	0.1	0.0	0.0	-	8.7	-	-	15.1	85.9	17.1	90.4	28.5	0.0	69.1	0.1	0.0	0.0	0.0	0.0	0.0	14.9

TABLE 4.21. The full results of ERDA under unsupervised setting on Cityscapes [222] validation set (27 classes). The baselines are DINO + ERDA. "-" indicates that the class does not present.



(A) Rooms and cluttered space.

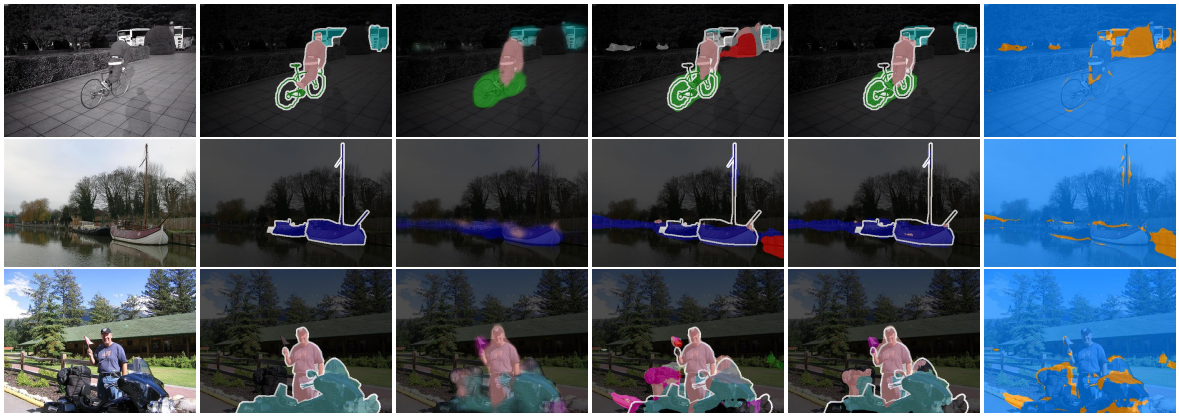
(B) Hallways.

(C) Offices.

FIGURE 4.12. We compare the results of baseline (RandLA-Net [109]) with the proposed ERDA. We additionally visualize the dense pseudo-labels (3rd column), by blending the color of different classes according to their estimated class likelihoods. It shows a clear indication of co-occurrence as semantic cues. With such dense and informative pseudo-labels for training, our ERDA can produce a cleaner and better segmentation with more details, as in the highlighted improvement (last column). The visualization is done on S3DIS Area 5.

(A) Indoor rooms.

(B) Animals.



(C) Outdoor.

FIGURE 4.13. We compare the results of baseline (FixMatch [227]) with the proposed ERDA. We also provide the dense pseudo-labels by blending the color, which show informative estimation on likely classes as well as its uncertainty to guide the model learning. We show that model trained with our ERDA can produce more accurate and detailed segmentations for both cluttered indoor scenes and outdoor scenes with occlusions, as in the highlighted improvement (last column). The visualization is done on Pascal validation.

CHAPTER 5

Structuring Adaptation via Geometric Context

In this chapter, we study geometry-grounded adaptation of large-scale pre-trained point cloud transformers under spatial and geometric context shift.

The emergence of large-scale pre-trained point cloud models has significantly advanced 3D scene understanding, but adapting these models to specific downstream tasks typically demands full fine-tuning, incurring high computational and storage costs. Parameter-efficient fine-tuning (PEFT) techniques, successful in natural language processing and 2D vision tasks, would underperform when naively applied to 3D point cloud models due to significant geometric and spatial distribution shifts. Existing PEFT methods commonly treat points as orderless tokens, neglecting important local spatial structures and global geometric contexts in 3D modeling. To bridge this gap, we introduce the Geometric Encoding Mixer (GEM), a novel geometry-aware PEFT module specifically designed for 3D point cloud transformers. GEM explicitly integrates fine-grained local positional encodings with a lightweight latent attention mechanism to capture comprehensive global context, thereby effectively addressing the spatial and geometric distribution mismatch. Extensive experiments demonstrate that GEM achieves performance comparable to or sometimes even exceeding full fine-tuning, while only updating 1.6% of the model's parameters, fewer than other PEFT methods. With significantly reduced training time and memory requirements, our approach thus sets a new benchmark for efficient, scalable, and geometry-aware fine-tuning of large-scale 3D point cloud models. Code is available at https://github.com/LiyaoTang/GEM.

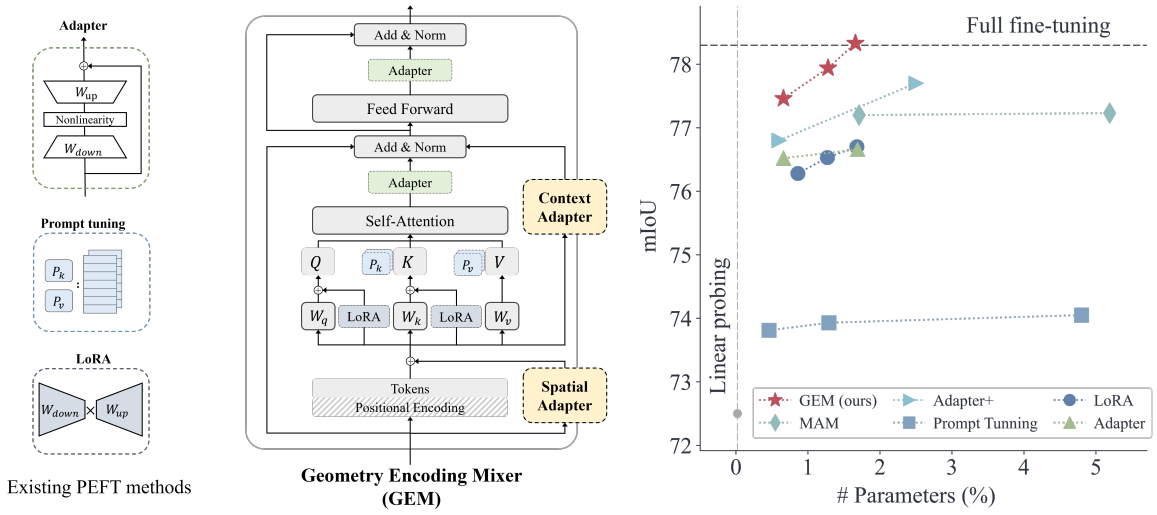


FIGURE 5.1. **(a)** Existing PEFT methods, such as adapters, prompt tuning, and LoRA, focus on adaptations inside attention and feed-forward layers. **(b)** In contrast, Geometry Encoding Mixer (GEM) explicitly encodes the geometric cues and mixes them into the pre-trained model, by the Spatial Adapter refining the pre-trained positional encoding and the Context Adapter complementing the local attention. **(c)** By capturing 3D spatial details and scene-wide geometry context, GEM reaches full fine-tuning performance while tuning $< 2\%$ parameters, outperforming existing PEFT methods.

5.1 Introduction

Scene semantic segmentation is a fundamental task for 3D world understanding that underpins many real-world applications, from autonomous driving and unmanned aerial vehicles to augmented reality [4, 9, 27]. Recent breakthroughs in large-scale models for language and 2D vision [37, 301, 311–314] have spurred an analogous trend toward training powerful point cloud backbones capable of capturing rich semantics from 3D scenes [17, 25, 28].

Despite significant capabilities, adapting large pre-trained backbone models to specific downstream target domains is non-trivial. The typical strategy heavily relies on full fine-tuning, a process that is both computationally expensive and storage-intensive, since each downstream task demands an independent copy of all model parameters. Furthermore, fine-tuning large models typically requires extensive datasets to prevent overfitting and catastrophic forgetting [315, 316], while large datasets are usually inaccessible when adapted to smaller,

specialized tasks. To address these issues, parameter-efficient fine-tuning (PEFT) methods offer a promising performance in reducing the cost of adaptation while reaching the performance of full fine-tuning with significantly fewer parameters.

While PEFT has been extensively validated in language processing and 2D vision tasks, its effectiveness on large-scale 3D point clouds remains largely unexplored and inadequately addressed. As demonstrated empirically in Fig. 5.1, existing PEFT methods only yield limited performance gains if directly applied to a 3D point cloud backbone. We tend to attribute this limitation to a key challenge that is often overlooked by current PEFT methods when using for 3D point cloud: unlike structured data such as text or images, point clouds are inherently unordered sets of coordinates in $\mathbb{R}^3$. They exhibit strong irregularity, sparsity, and structural variability, shaped by different sensing protocols and scene geometries. These factors lead to significant geometric and spatial distribution shifts between large-scale pre-training datasets and downstream domains, which existing PEFT methods fail to account for. In particular, representative PEFT methods, such as LoRA [49], adapters [48] and prompt tuning [317, 318], either adapt models at an isolated, per-point level or insert fixed external tokens at the global level, thus failing to adequately capture important geometric and spatial contexts inherent in 3D scenes. Moreover, to avoid the prohibitive computational cost of global attention over a large number of points, current 3D transformers predominantly employ local attention mechanisms without explicitly modeling global contexts, which further restricts the potential of current PEFT approaches in performance.

To address the above challenges, we re-examine PEFT for 3D scene understanding and hypothesize that effective PEFT on 3D scenes could take advantage of explicitly modeling both fine-grained local spatial patterns and global geometric contexts. We propose that neglecting either aspect would degrade fine-tuning performance: local-only adaptations may be prone to noise due to the lack of broader context consideration, whereas global-only methods would miss local details and lose precision. This motivates a novel PEFT adaptation framework tailored specifically for 3D scenes to bridge these local and global scales.

We introduce Geometry Encoding Mixer (GEM), a geometry-aware module for parameter-efficient fine-tuning of point cloud transformer on 3D scenes. It comprises two complementary

components: a spatial adapter at local neighborhood and a context adapter capturing the scene geometric context. In the spatial adapter, we employ a lightweight 3D convolutional bottleneck that operates on points neighborhood, enriching the pre-trained positional encoding by learning fine-grained local spatial details at target domains. In the context adapter, we introduce a set of learned latent tokens to serve as global context vectors. These latent tokens interact with the full point cloud through efficient attention, forming a bottleneck at the token dimension and bypassing the constraints of local attention, and thus aggregating scene-specific context from across the entire point cloud. By fusing these two paths, the Geometry Encoding Mixer effectively bridges local and global representations, providing the fine-tuned model with a richer understanding of the 3D scene than either component alone.

Empirically, we validate our approach on large-scale 3D scene datasets, including both indoor [14, 43, 44, 63] and outdoor [6] scenes. Our results indicate that models equipped with GEM consistently achieve performance matching or sometimes surpassing full fine-tuning methods, updating merely ~1.6% of model parameters, while being fewer than existing PEFT alternatives. These findings underscore the importance of explicitly modeling geometric and spatial contexts for efficient and effective adaptation in 3D scene understanding. To the best of our knowledge, our work represents the first exploration and validation of PEFT approaches tailored explicitly for large point cloud transformer under large-scale 3D scenes, hoping to establish a foundation for future research and practical deployments.

5.2 Related Work

Point cloud segmentation. Point clouds serve as an efficient representation for large-scale 3D scenes. Consequently, point cloud segmentation becomes a fundamental task that drives the design of 3D backbone architectures. Since the seminal work PointNet [18, 19], numerous 3D backbone architectures have been proposed for this task. These backbones either project the points onto a grid-like structure, *e.g.* 3D voxel grid, to exploit 3D convolutional networks [23, 90, 92, 187, 188, 319], or directly process the raw unordered point sets [20, 22, 24, 109, 129, 206]. Recently, inspired by the success of large transformer models in natural language

and 2D vision [33, 299, 301, 320], researchers have started training increasingly large and powerful point cloud backbones [25, 26, 102, 130–132, 237]. Although these models achieve strong performance, they typically need to be trained from scratch for each new dataset and often remain data-hungry [168, 244, 321, 322]. To reach optimal results, specialized training recipes and learning targets are also commonly required and actively explored [8, 130, 319].

Following the paradigm of large-scale pre-training that has proven successful in vision and language domains [11, 37, 52, 311, 312, 314], researchers have begun exploring similar strategies for 3D point clouds. In particular, self-supervised learning (SSL) on large 3D scene datasets has demonstrated promising results [145, 146, 151, 205]. However, most such efforts focus on modest convolutional backbones like SparseUNet [92] rather than transformer-based architectures. Only recently has an SSL approach been applied to a transformer-based 3D backbone: Sonata [17] introduced SSL for a large point transformer encoder [25], but even this method still requires full fine-tuning on each downstream dataset to achieve optimal performance.

In this work, we explore effective PEFT methods for large pre-trained point cloud backbones, with the goal of reducing the computational and storage overhead when adapting them to downstream datasets, while matching the performance of full fine-tuning.

Parameter-efficient fine-tuning (PEFT). The ever-growing size of transformer-based foundation models[11, 301, 323] makes full fine-tuning for downstream task prohibitively expensive in both memory and computation. PEFT methods address this challenge by adapting large models while updating only a small fraction of their parameters. Existing approaches fall into four broad categories.

Selective fine-tuning methods update a carefully chosen subset of the original weights. One simple variant is linear probing [37, 324] that trains only the classification head. Other strategies restrict training to specific parts of the network, *e.g.*, tuning only bias terms [325] and selecting a subset of parameters based on gradient magnitude criteria [326].

Adapter-based tuning inserts lightweight bottleneck modules into an otherwise frozen backbone. First explored for CNNs [327] and later extended to transformers, adapters may be

placed sequentially [48, 328] or in parallel [329–332] to the original modules. Beyond single task, adapters can be composed or fused, promoting knowledge sharing without catastrophic forgetting [316, 333].

Prompt-based tuning techniques, including prompt tuning [318] and prefix tuning [317], extend to the prompting paradigm in large language models [174]. Instead of hand-crafted prompts, these methods learn task-specific vectors that are prepended to the input sequence [318, 334] or injected as extra tokens in transformer layers [317, 335, 336], thereby steering the model without modifying its original weights [174, 318].

Low-rank adaptation (LoRA) constrains the weight updates to a low-dimensional subspace by learning a pair of low-rank matrices that are added to each pre-trained weight, typically in the attention layers [49]. This design approximates the effect of full fine-tuning while introducing only a small number of additional parameters. Successors further enhance the approximation ability, robustness, or rank allocation [337–341].

Recent works also propose hybrid schemes that highlight common design principles [175], for instance, integrating their respective strengths under a unified adapter framework [329].

Although PEFT has proved effective in language and 2D vision, a direct transfer to 3D scene understanding is often sub-optimal, largely due to the unordered nature of point clouds and the prominence of geometric cues. Our work thus investigates PEFT strategies tailored to large-scale 3D scenes.

PEFT for 3D point cloud. In the context of 3D point clouds, PEFT remains relatively underexplored. Existing methods thus far have focused mostly on object-level inputs with limited spatial scale. Some approaches introduce prompt tokens that adapt to 3D data, where the prefix tokens are dynamically generated from intermediate features [176, 342, 343] or from the spatial centers of local patches [344]. Several works also introduce auxiliary side networks, operating either in the spectral domain [345] or the spatial domain [346, 347], to provide geometric context. Other methods construct banks of prompt vectors using domain-specific dataset to inject 3D prior knowledge [177], or combine language-side prompt tuning with point-cloud adapters to handle open-vocabulary recognition [348].

Compared to object-level datasets [57, 58], 3D scene understanding involves much larger inputs containing on the order of millions of points [43, 44]. This scale amplifies computational challenges and underscores the need for PEFT techniques expressly designed for large 3D scene models, which is however a research direction that remains largely unexplored.

5.3 Methodology

We propose the *Geometry Encoding Mixer* (GEM) as a lightweight parameter-efficient module for fine-tuning point-cloud transformer. Fig. 5.2 summarises the overall architecture.

In particular, GEM consists of two complementary adaptations: a *Spatial Adapter* that refines the positional encoding of each point, injecting local geometry information overlooked by generic adapters; and a *Context Adapter* that distills a compact set of latent tokens that broadcast global scene-level cues. Both components follow the residual, bottleneck design of classic adapters, yet emphasize and operate on spatial rather than channel space.

5.3.1 Preliminaries

We first consider the direct application of existing PEFT methods on current point-cloud transformers.

Challenges under 3D. Following transformer [299, 301], point-cloud transformers also build upon self-attention layer (Attn) followed by feed-forward network (FFN). However, due to the large scale input points at *million* scale, global self-attention over all points would demand prohibitively large GPU memory and compute resource, if ever possible, due to the quadratic complexity of the attention operations: $\mathcal{O}(\text{Attn}) = n^2$, where n denotes the number of input points. In this regard, state-of-the-art point-cloud transformers propose various designs of local attentions [25, 26, 102, 131, 132] to cap the complexity, where each point can only attend to points within the same patch. With a patch size of p, the complexity of local attention is then $\mathcal{O}(\text{Attn}_{\text{loc}}) = np$. For example, it is set to $p = 16$ in PT [26] and $p = 1024$ in the larger PTv3 [25].

The irregular and sparse nature of 3D point cloud, together with these architecture designs, greatly impede the application of standard PEFT methods such as adapter, LoRA, and prompt tuning. As shown in Fig. 5.1, directly applying existing methods yields only marginal gains. We attribute this to their lack of spatial structure awareness of 3D geometry and the inability to capture scene-wide context under the local attention regimes of modern point cloud backbones.

Adapters. The adapter-based methods [48, 328] insert small modules after the attention or FFN layers:

$$f_{\text{adapter}}(\boldsymbol{x}) = \boldsymbol{x} + \sigma(\boldsymbol{x}\boldsymbol{W}_{\text{down}})\boldsymbol{W}_{\text{up}}, \tag{5.1}$$

where $\boldsymbol{W}_{\text{down}} \in \mathbb{R}^{d\times r}$ and $\boldsymbol{W}_{\text{up}} \in \mathbb{R}^{r\times d}$ are down and up projections that form a bottleneck structure with $r \ll d$ and activation σ in the mid, and are also surrounded by residual connection. As generic MLP bottlenecks, adapters adapt the model on a per-point basis, overlooking the important spatial cues in 3D point cloud.

LoRA. LoRA methods [49] typically adapt the attention projection layers and approximate the full fine-tuning with a bottleneck in weight space, transforming the attention into:

$$\text{Attn}_{\text{LoRA}} = \text{Attn}(\boldsymbol{Q} + \Delta\boldsymbol{Q}, \boldsymbol{K} + \Delta\boldsymbol{K}, \boldsymbol{V}), \tag{5.2}$$

where $\Delta\boldsymbol{Q} = \boldsymbol{x}\boldsymbol{W}_{\text{down}}\boldsymbol{W}_{\text{up}}$, with $\boldsymbol{W}_{\text{down}} \in \mathbb{R}^{d\times r}$ and $\boldsymbol{W}_{\text{up}} \in \mathbb{R}^{r\times d}$, and similarly for $\Delta\boldsymbol{K}$ with a different pair of weights. While LoRA can leverage the strong ability of the attention layers, we note that all attentions in point cloud transformer are constrained within local patches. LoRA methods are thus inherently limited by the prevalent design of local attention. With only limited rank $r << d$, it could fail to provide global context of the target dataset.

Pompt tuning. Prompt tuning methods [317, 318] explicitly introduce global tokens into the attention:

$$\text{Attn}_{\text{prompt}} = \text{Attn}(\boldsymbol{Q}, [\boldsymbol{P}_K; \boldsymbol{K}], [\boldsymbol{P}_V; \boldsymbol{V}]), \tag{5.3}$$

where $\boldsymbol{P}_K, \boldsymbol{P}_V \in \mathbb{R}^{m\times d}$ are two sets of tokens prepending to the original key-value pair. In spite of the constraint of local patches, prompt tokens can be duplicated into each patch.

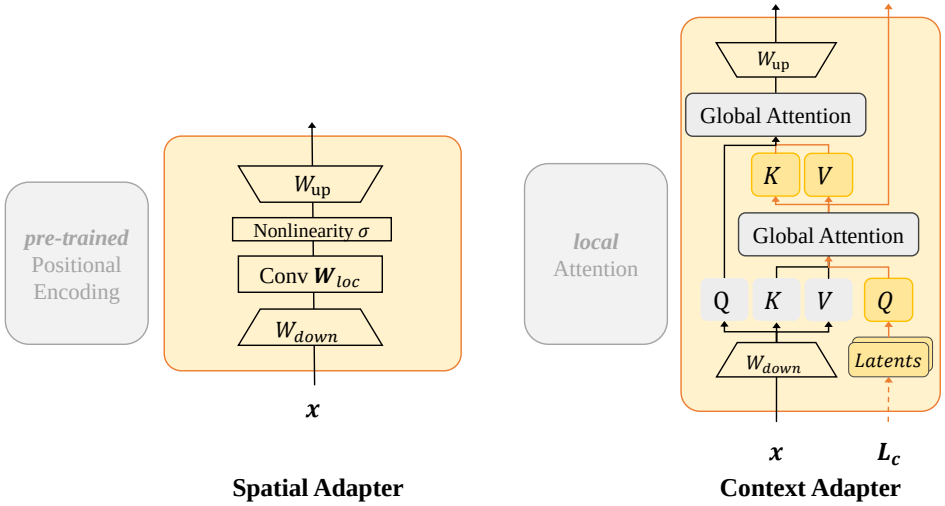


FIGURE 5.2. Geometry Encoding Mixer. We propose the spatial adapter to enhance the pre-trained positional encoding, and the context adapter to overcome the local attention mechanism, thus enhancing the efficient adaptation on large-scale 3D scenes with explicit geometry encoding.

However, with $m << n$, it can be hard for a few static external tokens to capture scene-specific context that can vary across point clouds within the same dataset. In addition, it also overlooks the spatial pattern of the point cloud, lacking the adaptation to local point features.

Others. There are more methods that explore other aspects. Selective tunings propose to tune a small subset of the frozen model parameters, such as BitFit [325] that tunes only bias terms. Hybrid methods are also explored, such as MAM [329] that couples FFN adapter with prompt tuning for attention layer. These methods still inherit the constraints from those representative PEFT methods, which are hindered by the local attention and fail to recognize the spatial structure.

In summary, existing methods overlook 3D geometry and remain confined to local patches, motivating a PEFT design that incorporates spatial awareness with scene-wide context, introduced next.

5.3.2 Our Methodology: Geometry Encoding Mixer (GEM)

To address these limitations, we present the Geometry Encoding Mixer (GEM). It refines positional encodings and injects scene context for efficient adaptation on large-scale 3D

scenes:

$$\boldsymbol{x} \leftarrow \boldsymbol{x} + \text{pos}(\boldsymbol{x}) + f_{\text{spatial}}(\boldsymbol{x}), \tag{5.4}$$

$$\boldsymbol{x} \leftarrow \boldsymbol{x} + \text{Attn}_{\text{loc}}(\boldsymbol{x}) + f_{\text{context}}(\boldsymbol{x}), \tag{5.5}$$

where $\boldsymbol{x}$ denotes input points and $\text{pos}(\cdot)$ is the pre-trained positional embedding. Fig. 5.2 illustrats the overall structure.

Spatial Adapter (SA). Point clouds are sparse, irregular samples in 3D. Adapting to their fine-grained geometry requires explicitly modeling local neighborhoods, which prior PEFT methods fail to address.

To this end, the spatial adapter refines per-point positional encoding through a lightweight 3D convolutional bottleneck. Concretely, for each point $\boldsymbol{x}$, we consider a 3D grid and gather points in the vicinity voxels as neighbors, denoted by $\mathcal{N}$:

$$f_{\text{spatial}}(\boldsymbol{x}) = \boldsymbol{x} + \sigma(\sum_{i \in \mathcal{N}} \boldsymbol{W}^i_{\text{loc}}(\boldsymbol{x}\boldsymbol{W}_{\text{down}}))\boldsymbol{W}_{\text{up}}, \tag{5.6}$$

where $\boldsymbol{W}_{\text{down}} \in \mathbb{R}^{d \times r}$ and $\boldsymbol{W}_{\text{up}} \in \mathbb{R}^{r \times d}$ are down and up projectsion with $r \ll d$. $\boldsymbol{W}^i_{\text{loc}} \in \mathbb{R}^{r \times r}$ composes the spatial kernel weights for i-th neighboring voxel and $\sigma(\cdot)$ is the nonlinearity, such as ReLU.

With a common kernel dimension $k = 3$, the spatial adapter touches at most k^3 neighbours per point and adds $2rd + k^3r^2$ parameters, rendering a complexity of $\mathcal{O}(2ndr + nk^3r^2) = \mathcal{O}(nd)$ given $k, r \ll n, d$. It thus functions as an efficient convolution-based positional encoding [349, 350] in parallel to the pre-trained positional encoding, thereby capturing fine spatial layout to match the target distribution.

Context Adapter (CA). Due to the local attention, any local and channel-wise adaptation would be constrained from acquiring scene-wide context, hindering the adaptation performance on downstream tasks like semantic segmentation.

To inject global context without breaking the $\mathcal{O}(n^2)$ barrier, we introduce m latent tokens $\boldsymbol{L} \in \mathbb{R}^{m \times r}$, where $m \ll n$, that attend to all points once:

$$\boldsymbol{L}_c = \text{Attn}\big(\boldsymbol{L}\boldsymbol{W}^Q,\, \boldsymbol{K},\, \boldsymbol{V}\big), \tag{5.7}$$

$$f_{\text{context}}(\boldsymbol{x}) = \boldsymbol{x} \;+\; \text{Attn}\big(\boldsymbol{Q},\, \boldsymbol{L}_c\boldsymbol{W}^K,\, \boldsymbol{L}_c\boldsymbol{W}^V\big)\boldsymbol{W}_{\text{up}}, \tag{5.8}$$

where $\boldsymbol{L} \in \mathbb{R}^{m \times r}$ and $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V} = \boldsymbol{x}\boldsymbol{W}^Q_{\text{down}}, \boldsymbol{x}\boldsymbol{W}^K_{\text{down}}, \boldsymbol{x}\boldsymbol{W}^V_{\text{down}}$ are down-projected point features.

As both attention costs $O(nm)$ with $m \ll n$, global aggregation is affordable and at the same level of complexity as the conventional prompt tuning [317, 318]. Furthermore, we update $\boldsymbol{L} \leftarrow \boldsymbol{L} + \boldsymbol{L}_c$ at each adapter insertion to share across layers, yielding dynamic prompts that capture the context of current scene, unlike static prefixes in prompt tuning.

Discussion. In comparison to existing PEFT methods, GEM explicitly biases the adaptation to 3D geometry and context. The spatial adapter captures localized variations that generic adapters, such as LoRA and basic adapters, would miss, since those ignore spatial structure and use the positional encodings from pre-training. The context adapter circumvents the local attention design to provide scene-wide context for efficient adaptation with few parameters. By combining these two components, our approach enables scene-aware adaptation of 3D transformers, where each point is tuned with respect to both its neighbors and the entire point cloud. It thus results in significantly improved performance on 3D scene datasets with only a lightweight adaptation overhead[1].

5.4 Experiments

We present the results of our proposed GEM on semantic segmentation of large-scale 3D scene datasets and experiment with both self-supervised pre-trained backbone, Sonata [17],

[1]We study the empirical overheads in the supplementary Sec. 5.6.1.

and supervised pre-trained one, PTv3-PPT [28], for investigation[2]. We also provide ablation studies to reveal the detailed effects of different components better.

5.4.1 Experimental Setup

We primarily follow the comprehensive protocols proposed in Sonata [17], covering ScanNet [43], ScanNet200 [14], ScanNet++ [63] and S3DIS [44].

We adopt two representative pre-trained backbones, the Sonata model [17] with large-scale self-supervised training and the PTv3-PPT [28] with supervised pre-training on multiple large-scale curated datasets. We compare GEM with the existing popular PEFT methods, including bias-tuning from BitFit [325], Adapter [48], LoRA [49], and Prompt Tuning [317]. We consider linear probing as a baseline and the full fine-tuned model as our target strong reference. In addition, we note that the Sonata (full.) specifically introduces a task-specific decoder for semantic segmentation during the fine-tuning, which may not reflect the fair comparison. Therefore, we revive to the standard full fine-tuning without additional decoder network, which is denoted by Sonata (ft.).

For training, we follow the widely accepted fine-tuning setups to update only the inserted or selected weights with the pre-trained backbone weights remaining frozen. All PEFT baselines follow the implementations from released code, adopt the suggested common practice [351, 352], and are tuned to their best validation setting in Fig. 5.1(c). Specifically, we set the default rank to be $r = 32$ and the number of learnable tokens to be $m = 4$. More details are given in the supplementary.

5.4.2 Performance Comparison

Main results. Tab. 5.1 shows that GEM consistently surpasses all representative PEFT methods across datasets. It matches the performance of full fine-tuning on most datasets and even exceeds it on ScanNet++ [63]. ScanNet++ comprises large and diverse scenes captured

[2]For more generalization and comparisons in other settings, such as convolutional model and 3D shape analysis, please refer to the supplementary Sec. 5.6.2.

Semantic Seg.	Params		ScanNet Val [43]			ScanNet200 Val [14]			ScanNet++ Val [63]			S3DIS Area 5 [44]			S3DIS 6-fold [44]		
Methods	Learn.	Pct.	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc
Training from scratch																	
SparseUNet [92]	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
PTv3 [25]	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
Full fine-tuning																	
Sonata (full.) [17]	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
Sonata (ft.)	108.5M	100%	78.3	85.9	92.3	37.3	47.8	83.7	49.8	61.2	87.6	72.4	79.0	92.2	79.5	87.3	92.3
PEFT methods																	
Sonata (lin.)	0.02M	0.02%	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
+BitFit [325]	0.2M	0.2%	74.7	84.7	90.8	32.5	45.5	82.0	42.4	56.5	85.7	73.9	82.0	91.5	76.1	87.1	91.0
+Adapter [48]	2.8M	2.5%	77.0	85.4	91.8	33.6	45.7	82.5	42.6	57.5	85.6	73.8	82.9	91.5	76.4	87.5	91.4
+LoRA [49]	1.9M	1.7%	76.7	85.7	91.7	33.6	45.5	82.7	44.2	58.0	**86.5**	74.5	**83.2**	91.5	77.4	87.8	91.2
+Prompt Tunning [317]	5.5M	4.8%	74.3	84.1	90.5	31.4	44.4	81.6	41.2	56.2	84.8	73.4	82.5	91.0	73.7	86.5	90.5
+**GEM (ours)**	1.8M	1.6%	**78.3**	**86.6**	**92.3**	**35.6**	**46.9**	**83.3**	**46.6**	**60.3**	86.3	**75.1**	83.0	**92.1**	**77.9**	**88.2**	**92.1**

TABLE 5.1. Semantic segmentation.

at sub-millimeter resolution, diverging greatly from the spatial distribution of common pre-training datasets such as ScanNet [43]. Under this domain gap, the joint modeling of local spatial cues and global scene context introduced by GEM proves especially beneficial.

Interestingly, LoRA [49] and adapter [48] deliver similar scores, despite acting on different transformer components. This observation implies that, when updates remain local and geometric is not modeled explicitly, the precise choice of adaptation target can be secondary. In addition, prompt tuning [317] underperforms even the linear probing baseline in the S3DIS 6-fold evaluation, revealing the penalty of ignoring spatial structure during fine-tuning.

Data efficiency. We assess PEFT methods under the scenarios of limited data and annotations in Tab. 5.2. The results demonstrate the exceptional data efficiency of GEM, which outperforms both other PEFT methods and the full fine-tuning counterparts. Notably, under extreme data scarcity, such as 1% of the labeled scenes and limited annotations (20 points per scene), GEM surpasses both Sonata (full.) and Sonata (ft.), highlighting its superiority in low-data regimes.

Supervised pre-training. While self-supervised methods dominate the language domain, supervised pre-training remains competitive for vision [11]. To probe the limits of PEFT under large-scale 3D supervision, we adopt recent advances in 3D supervised pre-training [28], which achieve leading performance by training on a large collection of curated, labeled scene

datasets. As shown in Tab. 5.3, existing PEFT methods can even degrade performance, likely suffering from negative transfer [353, 354]. In contrast, GEM improves upon supervised backbone, further outperforming the larger PTv3 [25] with only 1.6% additional parameters.

PEFT with decoder. A dedicated segmentation decoder (13% of the total parameters) is known to lift fully fine-tuned baselines. We therefore ask whether PEFT can still add value in the presence of such a task-specific head. Tab. 5.4 answers in the affirmative: GEM outperforms all competing PEFT variants and even surpasses its own fully fine-tuned counterpart, setting a new state-of-the-art on ScanNet. The result underscores the merit of jointly modeling local geometry and scene-level context, even when the decoder already provides task-specific capacity.

Outdoor segmentations. During the original evaluation of Sonata [17], it employs separate pre-trained models for indoor and outdoor scenarios, due to the significant domain gaps between indoor and outdoor scenes. Here, we test a stronger setting: transferring an indoor pre-trained backbone to an outdoor autonomous-driving benchmark. We replace the input layer with a randomly initialized counterpart to match dimensionality and freeze all remaining weights.

Tab. 5.5 shows that GEM narrows the gap to a model trained from scratch on outdoor data, yet a noticeable performance gap persists. Closing this gap remains an interesting direction for future work.

5.4.3 Ablations and Analsys

We mainly consider the linear probing, Sonata (lin.), as baseline and ablate on ScanNet [43] to better investigate the key factors for effective PEFT in 3D scenes, shown in Tab. 5.6. For more studies on the effect of available parameters, please refer to the supplement.

Individual effectiveness of SA and CA. To validate our initial hypothesis, we study the individual effectiveness of the SA and CA in Tab. 5.6a. We find that both adapters can already be superior to the baseline and competitive to the existing PEFT methods. We notice that

Data Efficiency	Limited Scenes (Pct.)					Limited Annotation (Pts.)				
Methods	1%	5%	10%	20%	Full	20	50	100	200	Full
Training from scratch										
SparseUNet [92]	26.0	47.8	56.7	62.9	72.2	41.9	53.9	62.2	65.5	72.2
PTv3 [25]	25.8	48.9	61.0	67.0	77.2	60.1	67.9	71.4	72.7	77.2
Full fine-tuning										
Sonata (full) [17]	45.3	65.7	72.4	72.8	79.4	70.5	73.6	76.0	77.0	79.4
Sonata (ft.)	44.4	63.2	71.3	72.3	78.3	69.6	72.6	75.3	76.2	78.3
PEFT methods										
Sonata (lin.)	43.6	62.5	68.6	69.8	72.5	69.0	70.5	71.1	71.5	72.5
+BitFit [325]	46.5	64.9	69.9	71.7	74.7	71.0	72.3	72.9	73.6	74.7
+Adapter [48]	46.4	64.2	70.1	72.2	77.0	71.8	73.4	74.7	75.0	77.0
+LoRA [49]	46.6	63.0	70.1	72.6	76.7	72.1	73.6	75.2	75.5	76.7
+Prompt Tunning [317]	45.5	62.6	68.9	71.1	74.3	70.2	71.5	72.5	72.8	74.3
+**GEM (ours)**	**47.5**	**65.6**	**71.0**	**73.3**	78.3	**72.3**	**74.7**	**76.2**	**76.6**	78.3

TABLE 5.2. Data efficiency.

Supervised Pre-train.	Params		ScanNet Val [43]		
Methods	Learn.	Pct.	mIoU	mAcc	allAcc
Training from scratch					
SparseUNet [92]	39.2M	100%	72.3	80.2	90.0
PTv3 [25]	124.8M	100%	77.6	85.0	92.0
Full fine-tuning					
PTv3-PPT (ft.) [28]	97.4M	100%	78.6	86.0	92.5
PEFT methods					
PTv3-PPT (lin.)	0.1M	0.1%	78.6	85.9	92.5
+BitFit [325]	0.2M	0.2%	78.2	85.6	92.3
+Adapter [48]	1.8M	1.8%	78.5	85.9	92.4
+LoRA [49]	1.4M	1.7%	78.4	86.0	92.4
+Prompt Tunning [317]	4.8M	4.8%	78.3	85.9	92.4
+**GEM (ours)**	1.8M	1.6%	**79.1**	**86.6**	**92.6**

TABLE 5.3. Supervised pre-training.

with Decoder	Params		ScanNet Val [43]		
Methods	Learn.	Pct.	mIoU	mAcc	allAcc
Training from scratch					
PTv3 [25]	124.8M	100%	77.6	85.0	92.0
Full fine-tuning					
Sonata (full.) [17]	124.8M	100%	79.4	86.1	92.5
Sonata (full.)*	124.8M	100%	78.5	86.3	92.4
PEFT methods with decoder					
Sonata (dec.)	16.3M	13.1%	79.1	86.6	92.7
Sonata (dec.)*	16.3M	13.1%	77.2	85.9	91.9
+BitFit [325]	16.4M	13.2%	78.1	86.1	92.3
+Adapter [48]	19.1M	15.0%	78.2	86.5	92.3
+LoRA [49]	18.2M	14.3%	78.9	87.0	92.5
+Prompt Tunning [317]	22.6M	17.2%	77.4	86.1	92.1
+**GEM (ours)**	18.1M	14.3%	**79.5**	**87.3**	**92.7**

TABLE 5.4. PEFT with segmentation decoder. Methods with * reports our re-produced results.

Outdoor Seg.	Params		Sem.KITTI Val [6]		
Methods	Learn.	Pct.	mIoU	mAcc	allAcc
Training from scratch					
PTv3 [25]	124.8M	100%	69.1	76.1	92.6
Full fine-tuning					
Sonata (full.) [17]†	124.8M	100%	72.6	77.9	93.4
Sonata (ft.)	108.5M	100%	68.8	75.2	92.6
PEFT methods					
Sonata (lin.)†	0.02M	0.02%	62.0	72.5	91.0
Sonata (lin.)	0.02M	0.02%	52.0	63.3	87.1
+BitFit [325]	0.2M	0.2%	59.9	70.6	91.0
+Adapter [48]	2.8M	2.5%	62.5	72.2	92.0
+LoRA [49]	1.9M	1.7%	63.8	74.0	92.0
+Prompt Tunning [317]	5.5M	4.8%	59.1	71.0	90.8
+**GEM (ours)**	1.8M	1.6%	**67.7**	**75.5**	**93.1**

TABLE 5.5. Outdoor semantic segmentation. Methods with † use outdoor datasets for pre-training.

the SA takes the majority part of the introduced parameters, which is partially due to the inherently expensive 3D convolution with k^3 kernels. Such challenge is also 3D specific and urges us to develop the CA, rather than solely relying on the positional encoding to capture 3D geometry. By combining the two form of spatial adapters, we obtain GEM that reaches the best performance by comprehensively exploring the 3D scene while remaining parameter-efficient.

Understanding the latent tokens. In Tab. 5.6b, we are motivated to further investigate how efficient and effective the proposed CA could be. We thus study how few tokens it could afford before failing to capture the global context. Surprisingly, we find that CA can use as few as one single token to effectively express the global scene context, obtaining clear improvement over SA-only approach. We consider this to be the effectiveness of weight decomposition, where we are akin to approximating the n^2 global self-attention weights with two $n \times 1$ matrices. In comparison to LoRA that decomposes in the weight space, we decompose the spatial attention weight matrices and form a spatial bottleneck at token dimension. In addition, while we are aware that related topics have been similarly explored for efficient attentions [355, 356], we are, to the best of our knowledge, the first to motivate their effective application as PEFT methods for 3D scenes understanding.

Sharing the latent tokens. The last experiment, Tab. 5.6c probes whether the latent tokens learned by CA should remain layer-local or be shared across the hierarchy, assessing its ability to capture dynamic and scene-specific context. When each layer keeps its own bank of tokens (*N/A*), GEM already surpasses SA-only, confirming that CA supplies complementary global cues. Re-using the same tokens within each encoder stage (*per-stage*) yields a further yet modest gain, suggesting that a coherent context inside each resolution level helps the network reconcile local geometry with mid-range structure.

The strongest result emerges when a single set of latent tokens is shared by *all* transformer blocks. We attribute this improvement to its consistency with the fact that 3D point clouds represent the same underlying 3D geometry in spite of their irregularity and sparsity. From this perspective, such globally shared latent tokens confine the model to adapt to the actual underlying scene, rather than overfitting on irrelevant patterns due to the noisy sampled point cloud.

Revealing the geometry cues in latent tokens. Under the above assumption that latent tokens regularize the model to capture the underlying 3D scene, we visualize the attention weights and compare them to those produced by other PEFT methods that also inject global context, such as prompt tuning.

	SA	CA	Params	mIoU
Sonata (lin.)			0.02%	72.5
	✓		1.0%	77.2
+ GEM		✓	0.6%	77.3
	✓	✓	1.6%	78.3

(A) **SA** and **CA** compose of effective PEFT for 3D scenes, individually and jointly.

m	mIoU	mAcc	allAcc
1	78.1	86.4	92.1
4	78.3	86.6	92.3
8	78.2	86.8	92.3

(B) **GEM** is robust to available latent tokens.

strategy	mIoU	mAcc	allAcc
N/A.	77.8	86.1	92.1
per-stage.	78.0	86.1	92.0
global	78.3	86.6	92.3

(C) **GEM** can provide better capabilities when sharing latent tokens.

TABLE 5.6. Ablations on GEM. If not specified, the backbone is the self-supervised pre-trained Sonata [17] with no additional task-specific decoder, evaluated on ScanNet [43]. Default settings are marked in gray.

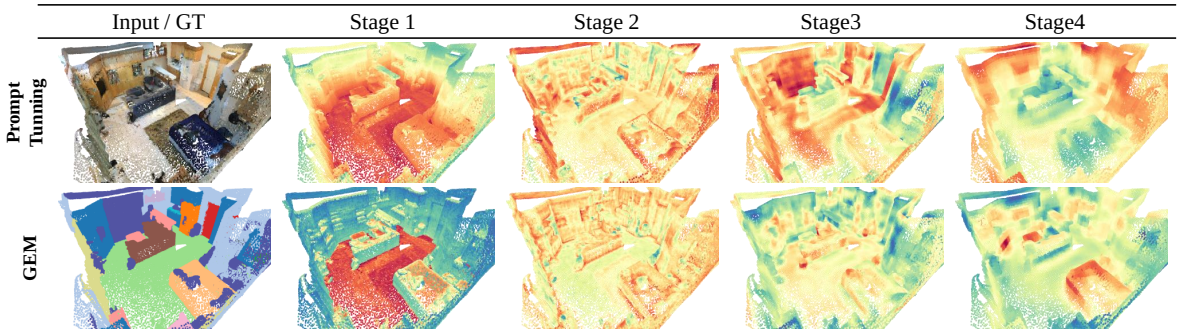


FIGURE 5.3. We visualize the attention weights of our latent tokens, comparing to the attentional weights with the prompt tuning, showing the enhanced geometry cues produced by GEM. More visualizations in Sec. 5.6.3.

As shown in Fig. 5.3, starting from the first stage, our latent tokens learn to produce crisper attention weights that follow geometric cues, such as the simple geometric primitives of floor surface. In deeper stages, the tokens progressively attend to more salient geometric objects, including clutter on the table and sofa. In contrast, the static tokens introduced by prompt tuning produce yield blurrier attention patterns that often span across object and region boundaries.

These observations suggest that the proposed latent tokens effectively capture and broadcast scene-wide geometric context. Combined with the local spatial structure extracted by 3D convolutions, this leads to superior performance, establishing our method as an effective PEFT approach for 3D scene understanding.

5.5 Summary

This study presents GEM, a geometry-aware fine-tuning module tailored for large-scale 3D scene segmentation. By combining local spatial refinement and global context modeling, GEM addresses key challenges inherent in adapting pre-trained 3D models to new domains. Experimental results across multiple benchmarks confirm its ability to achieve competitive performance with minimal parameter updates. These findings highlight the value of incorporating geometric priors into efficient model adaptation and offer a promising direction for scalable deployment in 3D scene understanding tasks.

Limitation and future work. While GEM achieves promising performance with little overhead in complexity, we notice it relies on the assumption that the model can be fully parallelized. For example, it would incur noticeable overhead if using large batch sizes, as the device is forced to process the input sequentially, in spite of the parallel design in GEM. In addition, we are able to successfully train our PEFT methods with a fraction of epochs than the full fine-tuning, *e.g.* 100 rather than 800 epochs, we realize that the gradients need to be back-propagated into early layers and could hinder the training efficiency, especially with dense point clouds like S3DIS [44]. We thus suggest that it is required a systematic examination on the additional overhead that PEFT methods would experience in 3D scene understanding.

5.6 Appendix

In this supplementary, we provide more materials as follows,
Sec. 5.6.1 details the implementation, training, and inference;
Sec. 5.6.2 presents additional experiments and analyses with tight parameter budgets
Sec. 5.6.3 offers more qualitative visualizations.

Methods	Params.	Memory	Latency
Base model			
Sonata (full.) [17]	124.8M	8.2G	61.8ms
Sonata (ft.)	108.5M	6.4G	50.2ms
PEFT methods			
Sonata (lin.)	0.02M	**6.4G**	**50.2ms**
+ Adapter [48]	2.8M	6.7G	50.6ms
+ LoRA [49]	1.9M	7.9G	52.8ms
+ Prompt Tunning [317]	5.5M	7.3G	51.5ms
+ **GEM (ours)**	1.8M	7.1G	57.8ms

TABLE 5.7. Inference efficiency of PEFT methods. We benchmark with the batch size fixed to 1.

Shape-Part Seg.	Params	ShapeNetPart [57]	
Methods	Learn.	Cls. mIoU	Inst. mIoU
Full fine-tuning			
ReCon (ft.) [150]	27.06	84.52	86.1
PEFT methods			
ReCon (lin.) [150]	5.23M	83.06	85.2
+ PointLoRA [347]	5.63M	83.98	85.4
+ PointGST [345]	5.59M	83.98	85.8
+ **GEM (ours)**	5.58M	**84.02**	**85.8**

TABLE 5.8. Generalizing to 3D shapes.

5.6.1 Details of Implementation, Training, and Inference

Our implementation is based on the open-source codebase Pointcept (here) and follows the official implementations for Sonata [17], PPT [28], as well as PTv3 [25].

Leveraging the parameter efficiency of our method, we train on a single 4090 GPU for much fewer epochs to obtain the reported performance. For example, we train on ScanNet for 100 epochs, in contrast to the 800 epochs in the released Sonata configuration. Our code is available at https://github.com/LiyaoTang/GEM.

Tab. 5.7 summarizes empirical latency and memory usage. We intentionally refrain from deployment-specific optimizations such as LoRA weight merging. Although the global attention and local convolution modules introduce extra cost, the runtime and memory footprint stay within the same order of magnitude as other PEFT baselines.

5.6.2 More Experiments and Analysis

In spite of the promising results shown in Sec. 5.3, we suggest that it can better demonstrate the full potential of PEFT methods under broader and more challenging settings.

Tight Budgets	Params		ScanNet200 Val [14]		
Methods	Learn.	Pct.	mIoU	mAcc	allAcc
Training from scratch					
SparseUnet [92]	39.2M	100%	25.0	32.9	80.4
Full fine-tuning					
SparseUnet (ft.) [151]	0.02M	0.05%	32.0	41.6	82.3
PEFT methods with rank=1					
SparseUnet (lin.)	0.02M	0.05%	1.5	2.5	53.6
+BitFit [325]	0.03M	0.07%	4.8	6.9	62.7
+Adapter [48]	0.6M	1.5%	9.5	13.4	69.5
+LoRA [49]	0.8M	2.0%	13.2	17.6	74.7
+**GEM (ours)**	0.6M	1.5%	**15.2**	**22.9**	**75.6**

TABLE 5.9. Generalizing to convolutional networks.

Tight Budgets	Params		ScanNet Val [43]		
Methods	Learn.	Pct.	mIoU	mAcc	allAcc
Training from scratch					
PTv3 [25]	124.8M	100%	77.6	85.0	92.0
Full fine-tuning					
Sonata (full.) [17]	124.8M	100%	79.4	86.1	92.5
Sonata (ft.)	108.5M	100%	78.3	85.9	92.3
PEFT methods with rank=1					
Sonata (lin.)	0.02M	0.02%	72.5	83.1	89.7
+Adapter [48]	0.05M	0.04%	74.0	84.1	90.5
+LoRA [49]	0.05M	0.05%	42.5	55.4	75.1
+Prompt Tunning [317]	0.05M	0.05%	72.9	83.5	90.0
+**GEM (ours)**	0.07M	0.06%	**75.2**	**84.8**	**91.1**
PEFT methods with 0.1% params.					
Sonata (lin.)					
+Adapter [48]	0.1M	0.1%	75.1	84.7	91.1
+LoRA [49]	0.1M	0.1%	75.0	84.4	91.1
+Prompt Tunning [317]	0.1M	0.1%	73.5	83.9	90.3
+**GEM (ours)**	0.1M	0.1%	**76.5**	**85.5**	**91.6**
PEFT methods with 1% params.					
Sonata (lin.)					
+Adapter [48]	1.1M	1.0%	76.6	85.3	91.7
+LoRA [49]	1.1M	1.0%	76.7	85.6	91.7
+Prompt Tunning [317]	1.1M	1.0%	73.8	84.2	90.4
+**GEM (ours)**	1.1M	1.0%	**78.2**	**86.3**	**92.2**

TABLE 5.10. Performance with tight budgets.

PEFT for shape analysis. While GEM is motivated by PEFT for large-scale 3D scenes, most existing 3D PEFT methods target object-level understanding, as discussed in Sec. 5.2. To assess cross-domain generalization and compare against these existing 3D-specific PEFT methods, we further evaluate GEM on 3D shape datasets, ShapeNetPart [57]. As shown in Tab. 5.8, GEM remains competitive and achieves the best performance. We also observe that common backbones for shape analysis, such as ReCon [150], attach large MLP heads that account for approximately 20% of all parameters, rendering fine-tuning inefficient. To broaden the applicability of backbone models for 3D shapes, an interesting direction for future work is to develop architectures with more parameter-efficient heads that are better suited to fine-tuning.

PEFT for 3D convolutional networks. While our primary experiments apply GEM to state-of-the-art backbones, mainly transformer models, we also verify its generality on

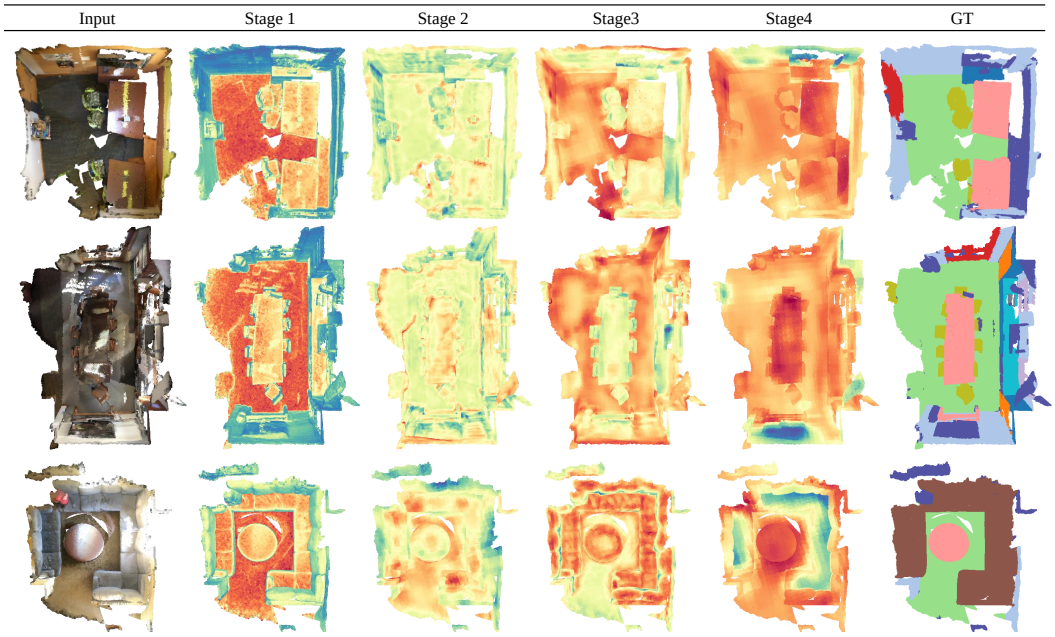


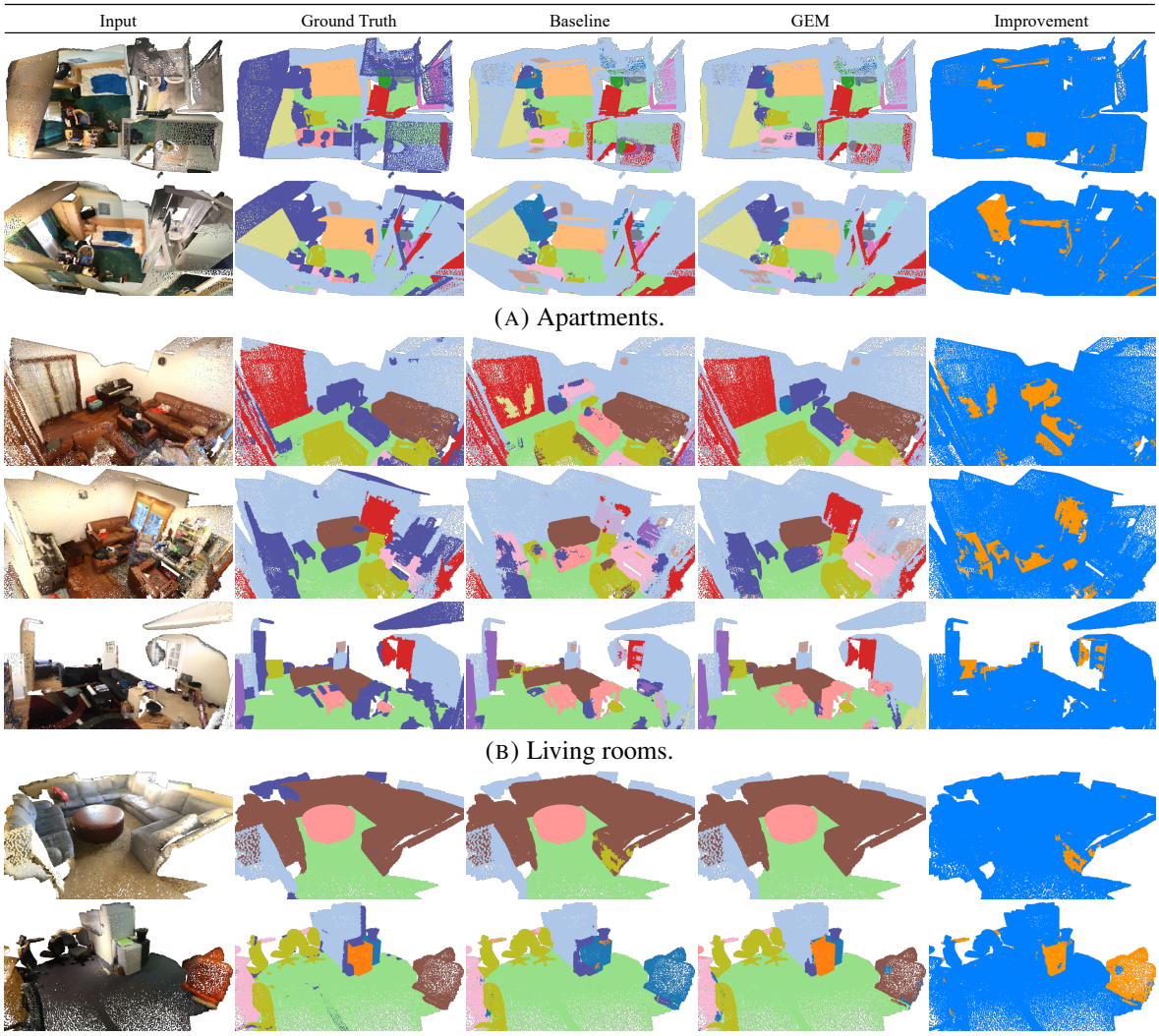
FIGURE 5.4. Attention maps of our latent tokens in context adapter.

convolutional architectures. In particular, we integrate GEM into SparseUNet [92], a 3D convolutional model, following the MSC [151] protocol to pre-train on ScanNet [43] and fine-tune on ScanNet200 [14]. As reported in Tab. 5.9, GEM improves performance by +4.6 mIoU over linear probing and outperforms other PEFT baselines, demonstrating robustness beyond transformer models. We notice the gains are narrower than those on transformer backbones, largely due to the limited capacity of convolutions, as also indicated by a near-collapse linear probing performance in this setting.

PEFT under tight budgets. To stress parameter efficiency, we constrain all methods to the same limited learnable-parameter budgets, including enforcing rank $r = 1$, allowing 0.1% parameters, and 1% parameters. As reported in Tab. 5.10, GEM delivers consistently higher accuracy, whereas several baselines fail when the budget becomes extremely stringent.

5.6.3 More Visualizations

We provide additional qualitative results regarding the attention maps generated by our latent tokens in Fig. 5.4. Although shared across stages, the latent tokens appear to attend to different



(A) Apartments.

(B) Living rooms.

(C) Offices and lounges.

FIGURE 5.5. We compare the results of baseline (lin.) with the proposed GEM.

parts of the scene, providing different scene context to the backbone models at different stages. Consistently, the visual comparison with the baseline in Fig. 5.5 shows clear improvements. Across different scenes, GEM yields cleaner and more coherent segmentations, with fewer confusions in cluttered regions and sharper boundaries at object interfaces.

CHAPTER 6

Conclusion and Outlook

6.1 Conclusion

This thesis advances a geometry-grounded view of 3D world understanding via point cloud semantic segmentation, a principled route from geometric measurements to high-level semantic understanding. Rather than treating 3D geometry as merely an input, we show that it can serve as structure that shapes the *output* we predict, the *supervision* we trust, and the *context* in which we adapt. From this perspective, scene segmentation is more than point-wise labeling: it is learning to recover coherent spatial and semantic structure from sparse, noisy, and irregular samples.

We developed this perspective along three complementary axes.

In Chapter 3, we study the most fragile yet consequential regions of the prediction space, *i.e.*, semantic boundaries, where small local mistakes can distort global partitions. By introducing metrics for boundary evaluation and the Contrastive Boundary Learning (CBL) framework, we demonstrate that improving boundary discriminability is a direct lever for more faithful scene segmentation. The broader implication is that output structure can and should be modeled explicitly, rather than left to emerge implicitly from backbone capacity.

In Chapter 4, we explore the structure of learning signals by modeling noise in supervision, especially in label-efficient regimes where supervision is synthesized from model predictions and is unavoidably noisy. We show that common pseudo-labeling relies on stable training dynamics but can degenerate under noisy data and geometric perturbations. By regularizing pseudo-label distributions through entropy and alignment, we derive the Entropy-Regularized

Distribution Alignment (ERDA) scheme. ERDA reframes noise as an intrinsic component of supervision that can be modeled and controlled. Building on the noise-aware signals from ERDA learning, we further propose query-based pseudo-labeling to enable more stable learning across modalities, including 3D point clouds and 2D images, and across labeling regimes, even under significant geometric disturbance.

In Chapter 5, we address the structure of context for efficient deployment. Pre-trained point cloud transformers offer strong representations, yet real-world scenarios exhibit spatial sampling patterns and geometric statistics that vary across sensors and scenes. We demonstrate that naive parameter-efficient fine-tuning methods can underperform, because they typically treat points as orderless tokens. By explicitly injecting local spatial cues and global geometric context, the proposed Geometric Encoding Mixer (GEM) provides an efficient, geometry-aware mechanism for adapting large-scale 3D backbones under spatial and geometric shift.

Taken together, these contributions support a unified argument: in 3D scene segmentation, *structure is not an auxiliary constraint but the substance of learning*. By grounding output coherence, supervision informativeness, and adaptation efficacy in explicit geometry, this thesis offers a consolidated pathway toward scalable 3D world understanding that remains accurate under ambiguity, efficient under resource constraints, and robust under distribution shift.

6.2 Limitations

Despite the scope and coherence of the proposed perspective, several limitations remain. They arise from the deliberate choice of semantic segmentation on static point clouds as the principled route to geometry-grounded world understanding, and from the assumptions that make this route tractable.

Scope of task and data assumptions. This thesis centers on semantic segmentation of static point clouds. While segmentation is fundamental, it is not exhaustive. Its formulation does not explicitly model temporal consistency, motion, or interactive perception, and it largely

assumes that metric geometry is already available in a usable form, typically after sensing and preprocessing, rather than jointly inferred with semantics. As a result, real-world settings that require spatio-temporal association and tracking, such as 4D segmentation and tracking of point cloud sequences [92, 357] or closed-loop perception-action, remain beyond the empirical scope of this thesis.

Assumptions in output structure. CBL emphasizes crisp boundaries and boundary-aware evaluation, reflecting both the importance and the fragility of boundary regions. However, this focus implicitly assumes that semantic discontinuities are well-aligned with local cues and that sharper partitions are universally preferable. In practice, boundaries can be intrinsically ambiguous, as in cluttered regions, gradual transitions, or mixed materials, making the definition of boundaries often a modeling choice rather than an observable fact. Moreover, current boundary derivation typically relies on dense annotations, which are costly and limit scalability; this becomes more pronounced in open-vocabulary settings where class definitions may be specified by language rather than closed-set labels [156, 358]. Finally, stronger boundary objectives may trade off against calibrated uncertainty where multiple interpretations of semantic boundaries are plausible.

Assumptions in supervision modeling. ERDA treats supervision as a distribution and controls it through entropy regularization and alignment. This design relies on the model's predictive distribution being sufficiently informative, so that shaping confidence improves learning rather than amplifying bias. Under severe domain shift, long-tail classes, or systematic label noise, even with noise-aware learning signals, entropy-based objectives can over-commit to erroneous modes and prolong recovery. In addition, ERDA largely assumes a fixed label space, and therefore does not directly resolve open-set or open-vocabulary learning, where the class set is incomplete or evolving [156, 358].

Constraints in adaptation setting. GEM targets efficient adaptation of large pre-trained point cloud backbones under spatial and geometric shift, largely within parameter-efficient fine-tuning (PEFT) regimes. While it improves transferability and efficiency, it does not fully address continual or test-time adaptation where shifts accumulate online and where stability-plasticity trade-offs become central [359, 360]. Its effectiveness also depends on

the quality of pre-training, and on access to target data for deployment. More generally, the effectiveness of geometry-aware PEFT beyond conventional supervised fine-tuning, such as instruction tuning for high-level spatial reasoning, remains underexplored [165].

Evaluation coverage and generalizability. Empirical validation is necessarily bounded by available datasets, annotation protocols, and benchmark assumptions. Although the proposed methods are designed to be broadly applicable, generalization to unconstrained real-world deployments may be affected by sensor-dependent noise and bias, reconstruction artifacts, and label mismatch across datasets. Extending evaluation to more diverse environments, sensing conditions, and downstream requirements remains essential for comprehensive geometry-grounded understanding.

Collectively, these limitations are not merely caveats; they delineate the next level of challenges on the way toward geometry-grounded world understanding. Several of them directly motivate the directions outlined in the future outlook below.

6.3 Future Outlook

The geometry-grounded lens developed in this thesis suggests promising extensions that both broaden the scope and relax the assumptions highlighted above.

Beyond per-point semantics: structure at higher levels. Semantic segmentation is a foundational problem, but real-world understanding often demands richer outputs, more flexible supervision, and more adaptive deployment.

On the output side, the same principle of explicit structure extends naturally from per-point semantics to instance and panoptic segmentation, where boundaries become instance-level separation and can be coupled with grouping, association, and mask partitioning [118, 361]. Beyond instances, structured scene understanding can move toward compositional abstractions such as object-part hierarchies and 3D scene graphs, enabling scene-level reasoning grounded in 3D geometric constraints and spatial relations. On the supervision side, treating learning signals as distributions invites tighter integration with calibration, uncertainty estimation, and

interactive labeling strategies, particularly for challenging cases such as boundary-adjacent, cluttered, and long-tail regions. On the adaptation side, geometry-aware PEFT can be extended to continual and test-time settings, where shift emerges gradually during online deployment and where efficiency must be coupled with stability and lifelong learning.

More broadly, geometry-grounded perception connects to embodied decision-making. In robotics and spatial agents, segmentation is rarely an endpoint, but provides the foundation for planning, interaction, and learning from experience. The three axes in this thesis suggest a concrete route forward: structured outputs that preserve actionable spatial partitions, supervision that reflects uncertainty under partial observability, and adaptation mechanisms that track changing spatial statistics as the agent moves through new environments.

Foundation models and multimodal grounding. Large-scale pre-training is rapidly reshaping 3D perception. A key opportunity is to align geometry-grounded structure with the emerging wave of multimodal and foundation models. The central challenge is to retain geometric faithfulness while leveraging semantic priors from broader data sources, such that semantics remains anchored to spatial geometry rather than dataset-specific correlations.

Recent progress increasingly blurs the boundary between 2D perception and 3D representation learning. Feed-forward reconstruction models [1, 158, 159] infer rich 3D attributes from multi-view inputs, while 3D-augmented VLM frameworks [165, 362] inject explicit 3D objectives into instruction tuning. In parallel, joint 2D-3D self-supervised learning [29] suggests that a shared representation can emerge with both geometric consistency and semantic transferability. These trends motivate a compelling hypothesis that point-based representations may serve as a latent representation or a form of spatial memory that is persistent, editable, and language-aligned, bridging raw 2D observations to world model reasoning. Realizing this vision will require mechanisms that preserve geometric faithfulness while supporting open-vocabulary semantics and interactive querying.

Bibliography

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

[2] Yang Liu, Weixing Chen, Yongjie Bai, Xiaodan Liang, Guanbin Li, Wen Gao, and Liang Lin. Aligning cyber space with physical world: A comprehensive survey on embodied ai. *IEEE/ASME Transactions on Mechatronics*, 2025.

[3] Hugues Hoppe, Tony DeRose, Tom Duchamp, John McDonald, and Werner Stuetzle. Surface reconstruction from unorganized points. In *Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '92, page 71–78, New York, NY, USA, 1992. Association for Computing Machinery.

[4] Yulan Guo, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, and Mohammed Bennamoun. Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[5] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, SGP '06, page 61–70, Goslar, DEU, 2006. Eurographics Association.

[6] 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 *Proc. of the IEEE/CVF International Conf. on Computer Vision (ICCV)*, 2019.

[7] 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 *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11618–11628, Los Alamitos, CA, USA, June 2020. IEEE Computer Society.

[8] Liyao Tang, Yibing Zhan, Zhe Chen, Baosheng Yu, and Dacheng Tao. Contrastive boundary learning for point cloud segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8479–8489, 2022.

[9] Y. Xie, J. Tian, and X. X. Zhu. Linking points with labels in 3d: A review of point cloud semantic segmentation. *IEEE Geoscience and Remote Sensing Magazine*, 8(4):38–59, 2020.

[10] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2014.

[11] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, Oct. 2023.

[12] Shuquan Ye, Dongdong Chen, Songfang Han, and Jing Liao. Learning with noisy labels for robust point cloud segmentation. *International Conference on Computer Vision*, 2021.

[13] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 5580–5590, Red Hook, NY, USA, 2017. Curran Associates Inc.

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

[15] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In L. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 17. MIT Press, 2004.

[16] Aoran Xiao, Xiaoqin Zhang, Ling Shao, and Shijian Lu. A survey of label-efficient deep learning for 3d point clouds. *IEEE transactions on pattern analysis and machine intelligence*, PP, 2023.

[17] Xiaoyang Wu, Daniel DeTone, Duncan Frost, Tianwei Shen, Chris Xie, Nan Yang, Jakob Engel, Richard Newcombe, Hengshuang Zhao, and Julian Straub. Sonata:

Self-supervised learning of reliable point representations. In *CVPR*, 2025.

[18] Charles Ruizhongtai Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *CoRR*, abs/1612.00593, 2016.

[19] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *CoRR*, abs/1706.02413, 2017.

[20] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph CNN for learning on point clouds. *CoRR*, abs/1801.07829, 2018.

[21] Guohao Li, Matthias Müller, Ali K. Thabet, and Bernard Ghanem. Can gcns go as deep as cnns? *CoRR*, abs/1904.03751, 2019.

[22] Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J. Guibas. Kpconv: Flexible and deformable convolution for point clouds. *CoRR*, abs/1904.08889, 2019.

[23] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018.

[24] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. Pointcnn: Convolution on x-transformed points. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[25] 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.

[26] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. Point transformer, 2021.

[27] Dening Lu, Qian Xie, Mingqiang Wei, Kyle Gao, Linlin Xu, and Jonathan Li. Transformers in 3d point clouds: A survey, 2022.

[28] 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.

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

[30] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[31] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *European Conference on Computer Vision*, 2018.

[32] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *NeurIPS*, 2021.

[33] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022.

[34] Chaitanya Ryali, Yuan-Ting Hu, Daniel Bolya, Chen Wei, Haoqi Fan, Po-Yao Huang, Vaibhav Aggarwal, Arkabandhu Chowdhury, Omid Poursaeed, Judy Hoffman, Jitendra Malik, Yanghao Li, and Christoph Feichtenhofer. Hiera: A hierarchical vision transformer without the bells-and-whistles. *ICML*, 2023.

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

[36] Mathilde Caron, Hugo Touvron, Ishan Misra, Herve Jegou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2021.

[37] 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, 2023.

[38] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

[39] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, CVPR '97, page 731, USA, 1997. IEEE Computer Society.

[40] Y.Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 1, pages 105–112 vol.1, 2001.

[41] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials, 2011.

[42] Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(3):447–461, Mar. 2016.

[43] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017.

[44] Iro Armeni, Sasha Sax, Amir R Zamir, and Silvio Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017.

[45] Dong-Hyun Lee. Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks. 2013.

[46] J. Quinonero-Candela, M. Sugiyama, A. Schwaighofer, and N.D. Lawrence. *Dataset Shift in Machine Learning*. Neural Information Processing series. MIT Press, 2008.

[47] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17(1):2096–2030, Jan. 2016.

[48] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, 2019.

[49] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[50] Towaki Takikawa, David Acuna, Varun Jampani, and Sanja Fidler. Gated-scnn: Gated shape cnns for semantic segmentation, 2019.

[51] Eric Arazo, Diego Ortego, Paul Albert, Noel E. O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning, 2020.

[52] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15979–15988, 2022.

[53] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, page 1097–1105, Red Hook, NY, USA, 2012. Curran Associates Inc.

[54] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, Los Alamitos, CA, USA, jun 2016. IEEE Computer Society.

[55] Charles R. Qi, Hao Su, Matthias NieBner, Angela Dai, Mengyuan Yan, and Leonidas J. Guibas. Volumetric and multi-view cnns for object classification on 3d data. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2016.

[56] Zhirong Wu, S. Song, A. Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1912–1920, June 2015.

[57] Li Yi, Vladimir G. Kim, Duygu Ceylan, I-Chao Shen, Mengyan Yan, Hao Su, Cewu Lu, Qixing Huang, Alla Sheffer, and Leonidas Guibas. A scalable active framework for region annotation in 3d shape collections. *ACM Trans. Graph.*, 35(6), Nov. 2016.

[58] Kaichun Mo, Shilin Zhu, Angel X. Chang, Li Yi, Subarna Tripathi, Leonidas J. Guibas, and Hao Su. PartNet: A large-scale benchmark for fine-grained and hierarchical part-level 3D object understanding. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[59] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Duc Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *International Conference on Computer Vision (ICCV)*, 2019.

[60] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. *arXiv preprint arXiv:2212.08051*, 2022.

[61] S. Song, S. P. Lichtenberg, and J. Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 567–576, June 2015.

[62] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *International Conference on 3D Vision (3DV)*, 2017.

[63] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12–22, 2023.

[64] Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. Structured3d: A large photo-realistic dataset for structured 3d modeling. In *Proceedings of The European Conference on Computer Vision (ECCV)*, 2020.

[65] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. In *SIGGRAPH*, 2018.

[66] 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 *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.

[67] 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 preprint arXiv:2406.10224*, 2024.

[68] Timo Hackel, N. Savinov, L. Ladicky, Jan D. Wegner, K. Schindler, and M. Pollefeys. SEMANTIC3D.NET: A new large-scale point cloud classification benchmark. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume IV-1-W1, pages 91–98, 2017.

[69] Weikai Tan, Nannan Qin, Lingfei Ma, Ying Li, Jing Du, Guorong Cai, Ke Yang, and Jonathan Li. Toronto-3d: A large-scale mobile lidar dataset for semantic segmentation of urban roadways. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, page 797–806. IEEE, June 2020.

[70] Xavier Roynard, Jean-Emmanuel Deschaud, and François Goulette. Paris-lille-3d: A large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification. *The International Journal of Robotics Research*, 37(6):545–557, 2018.

[71] Qingyong Hu, Bo Yang, Sheikh Khalid, Wen Xiao, Niki Trigoni, and Andrew Markham. Sensaturban: Learning semantics from urban-scale photogrammetric point clouds. *International Journal of Computer Vision*, 130(2):316–343, 2022.

[72] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361, 2012.

[73] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset, 2019.

[74] Jakob Geyer, Yohannes Kassahun, Mentar Mahmudi, Xavier Ricou, Rupesh Durgesh, Andrew S. Chung, Lorenz Hauswald, Viet Hoang Pham, Maximilian Mühlegg, Sebastian Dorn, Tiffany Fernandez, Martin Jänicke, Sudesh Mirashi, Chiragkumar Savani,

Martin Sturm, Oleksandr Vorobiov, Martin Oelker, Sebastian Garreis, and Peter Schuberth. A2D2: Audi Autonomous Driving Dataset. 2020.

[75] Ozan Unal, Dengxin Dai, and Luc Van Gool. Scribble-supervised lidar semantic segmentation. *ArXiv*, abs/2203.08537, 2022.

[76] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021)*, 2021.

[77] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec 2015.

[78] Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, and Siddhartha Chaudhuri. 3d shape segmentation with projective convolutional networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6630–6639, 2017.

[79] A. Boulch, B. Le Saux, and N. Audebert. Unstructured point cloud semantic labeling using deep segmentation networks. In *Proceedings of the Workshop on 3D Object Retrieval*, 3Dor '17, page 17–24, Goslar, DEU, 2017. Eurographics Association.

[80] Felix Järemo Lawin, Martin Danelljan, Patrik Tosteberg, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. Deep projective 3d semantic segmentation. *Lecture Notes in Computer Science*, page 95–107, 2017.

[81] Angela Dai and Matthias Nießner. 3dmv: Joint 3d-multi-view prediction for 3d semantic scene segmentation. *CoRR*, abs/1803.10409, 2018.

[82] Maximilian Jaritz, Jiayuan Gu, and Hao Su. Multi-view pointnet for 3d scene understanding. In *ICCV Workshop 2019*, 2019.

[83] Bo Li, Tianlei Zhang, and Tian Xia. Vehicle detection from 3d lidar using fully convolutional network. *Robotics: Science and Systems XII*.

[84] Gregory P. Meyer, Ankit Laddha, Eric Kee, Carlos Vallespi-Gonzalez, and Carl K. Wellington. Lasernet: An efficient probabilistic 3d object detector for autonomous driving. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*

(CVPR), pages 12669–12678, 2019.

[85] Bichen Wu, Alvin Wan, Xiangyu Yue, and Kurt Keutzer. Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3d lidar point cloud. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018.

[86] Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5mb model size, 2016.

[87] D. Maturana and S. Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 922–928, Sep. 2015.

[88] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018.

[89] Jing Huang and Suya You. Point cloud labeling using 3d convolutional neural network. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 2670–2675, 2016.

[90] Lyne Tchapmi, Christopher Choy, Iro Armeni, JunYoung Gwak, and Silvio Savarese. Segcloud: Semantic segmentation of 3d point clouds. *2017 International Conference on 3D Vision (3DV)*, Oct 2017.

[91] Dario Rethage, Johanna Wald, Jürgen Sturm, Nassir Navab, and Federico Tombari. Fully-convolutional point networks for large-scale point clouds. *Lecture Notes in Computer Science*, page 625–640, 2018.

[92] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3075–3084, 2019.

[93] Ben Graham. Sparse 3d convolutional neural networks. *ArXiv*, abs/1505.02890, 2015.

[94] Martin Engelcke, Dushyant Rao, Dominic Zeng Wang, Chi Hay Tong, and Ingmar Posner. Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1355–1361, 2017.

[95] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, October 2018.

[96] Benjamin Graham and Laurens van der Maaten. Submanifold sparse convolutional networks. *ArXiv*, abs/1706.01307, 2017.

[97] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018.

[98] Zongji Wang and Feng Lu. Voxsegnet: Volumetric cnns for semantic part segmentation of 3d shapes. *IEEE transactions on visualization and computer graphics*, 2018.

[99] Roman Klokov and Victor Lempitsky. Escape from cells: Deep kd-networks for the recognition of 3d point cloud models. *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[100] Gernot Riegler, Ali Osman Ulusoy, and Andreas Geiger. Octnet: Learning deep 3d representations at high resolutions. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017.

[101] Peng-Shuai Wang, Yang Liu, Yu-Xiao Guo, Chun-Yu Sun, and Xin Tong. O-cnn: octree-based convolutional neural networks for 3d shape analysis. *ACM Transactions on Graphics*, 36(4):1–11, Jul 2017.

[102] Peng-Shuai Wang. Octformer: Octree-based transformers for 3d point clouds. *ACM Transactions on Graphics*, 42(4):1–11, July 2023.

[103] Hang Su, Varun Jampani, Deqing Sun, Subhransu Maji, Evangelos Kalogerakis, Ming-Hsuan Yang, and Jan Kautz. Splatnet: Sparse lattice networks for point cloud processing. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018.

[104] Radu Alexandru Rosu, Peer Schutt, Jan Quenzel, and Sven Behnke. Latticenet: Fast point cloud segmentation using permutohedral lattices. *ArXiv*, abs/1912.05905, 2019.

[105] Xiuye Gu, Yijie Wang, Chongruo Wu, Yong Jae Lee, and Panqu Wang. Hplflownet: Hierarchical permutohedral lattice flownet for scene flow estimation on large-scale point clouds. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2019.

[106] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabás Póczos, Ruslan Salakhutdinov, and Alexander J. Smola. Deep sets. *CoRR*, abs/1703.06114, 2017.

[107] Mor Joseph-Rivlin, Alon Zvirin, and Ron Kimmel. Mo-net: Flavor the moments in learning to classify shapes. *CoRR*, abs/1812.07431, 2018.

[108] Xiao Sun, Zhouhui Lian, and Jianguo Xiao. Srinet. *Proceedings of the 27th ACM International Conference on Multimedia*, Oct 2019.

[109] 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. *CoRR*, abs/1911.11236, 2019.

[110] L. Jiang, H. Zhao, S. Liu, X. Shen, C. Fu, and J. Jia. Hierarchical point-edge interaction network for point cloud semantic segmentation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10432–10440, 2019.

[111] Oren Dovrat, Itai Lang, and Shai Avidan. Learning to sample. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2019.

[112] Chenyang Zhu, Kai Xu, Siddhartha Chaudhuri, Li Yi, Leonidas Guibas, and Hao Zhang. Adacoseg: Adaptive shape co-segmentation with group consistency loss, 2019.

[113] J. Liu, B. Ni, C. Li, J. Yang, and Q. Tian. Dynamic points agglomeration for hierarchical point sets learning. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7545–7554, Oct 2019.

[114] 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. *CoRR*, abs/1904.03375, 2019.

[115] Weiyue Wang, Ronald Yu, Qiangui Huang, and Ulrich Neumann. Sgpn: Similarity group proposal network for 3d point cloud instance segmentation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2569–2578, 2018.

[116] Xinlong Wang, Shu Liu, Xiaoyong Shen, Chunhua Shen, and Jiaya Jia. Associatively segmenting instances and semantics in point clouds. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4091–4100, 2019.

[117] Quang-Hieu Pham, Duc Thanh Nguyen, Binh-Son Hua, Gemma Roig, and Sai-Kit Yeung. Jsis3d: Joint semantic-instance segmentation of 3d point clouds with multi-task pointwise networks and multi-value conditional random fields. *2019 IEEE/CVF*

Conference on Computer Vision and Pattern Recognition (CVPR), pages 8819–8828, 2019.

[118] Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Pointgroup: Dual-set point grouping for 3d instance segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[119] Zhijian Liu, Haotian Tang, Yujun Lin, and Song Han. Point-voxel cnn for efficient 3d deep learning. In *NeurIPS*, 2019.

[120] Shaoshuai Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network. *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[121] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–779, 2018.

[122] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. *ArXiv*, abs/1912.13192, 2019.

[123] Yongcheng Liu, Bin Fan, Gaofeng Meng, Jiwen Lu, Shiming Xiang, and Chunhong Pan. Densepoint: Learning densely contextual representation for efficient point cloud processing. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2019.

[124] Lin-Zhuo Chen, Xuan yi Li, Deng-Ping Fan, Ming-Ming Cheng, Kai Wang, and Shao-Ping Lu. Lsanet: Feature learning on point sets by local spatial attention. *ArXiv*, abs/1905.05442, 2019.

[125] Alexandre Boulch. Convpoint: Continuous convolutions for point cloud processing. *Computers & Graphics*, 88:24 – 34, 2020.

[126] P. Hermosilla, T. Ritschel, P-P Vazquez, A. Vinacua, and T. Ropinski. Monte carlo convolution for learning on non-uniformly sampled point clouds. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2018)*, 37(6), 2018.

[127] Matan Atzmon, Haggai Maron, and Yaron Lipman. Point convolutional neural networks by extension operators. *ACM Transactions on Graphics*, 37(4):1–12, Jul 2018.

[128] Yongcheng Liu, Bin Fan, Shiming Xiang, and Chunhong Pan. Relation-shape convolutional neural network for point cloud analysis. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2019.

[129] Wenxuan Wu, Zhongang Qi, and Fuxin Li. Pointconv: Deep convolutional networks on 3d point clouds. *CoRR*, abs/1811.07246, 2018.

[130] 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 *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[131] Xin Lai, Jianhui Liu, Li Jiang, Liwei Wang, Hengshuang Zhao, Shu Liu, Xiaojuan Qi, and Jiaya Jia. Stratified transformer for 3d point cloud segmentation. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8490–8499, 2022.

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

[133] Maxim Tatarchenko, Jaesik Park, Vladlen Koltun, and Qian-Yi Zhou. Tangent convolutions for dense prediction in 3d. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018.

[134] Artem Komarichev, Zichun Zhong, and Jing Hua. A-cnn: Annularly convolutional neural networks on point clouds. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2019.

[135] Yiqun Lin, Zizheng Yan, Haibin Huang, Donglei Du, Ligang Liu, Shuguang Cui, and Xiaoguang Han. Fpconv: Learning local flattening for point convolution. *ArXiv*, abs/2002.10701, 2020.

[136] Mingyang Jiang, Yiran Wu, and Cewu Lu. Pointsift: A sift-like network module for 3d point cloud semantic segmentation. *ArXiv*, abs/1807.00652, 2018.

[137] Zhiyuan Zhang, Binh-Son Hua, and Sai-Kit Yeung. Shellnet: Efficient point cloud convolutional neural networks using concentric shells statistics. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2019.

[138] Pengyu Wang, Yuan Gan, Panpan Shui, Fenggen Yu, Yan Zhang, Song-Le Chen, and Zhengxing Sun. 3d shape segmentation via shape fully convolutional networks.

Comput. Graph., 70:128–139, 2018.

[139] Alexandre Boulch, Gilles Puy, and Renaud Marlet. Lightconvpoint: convolution for points. *ArXiv*, abs/2004.04462, 2020.

[140] Martin Simonovsky and Nikos Komodakis. Dynamic edge-conditioned filters in convolutional neural networks on graphs. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017.

[141] Kuangen Zhang, Ming Hao, Jing Wang, Clarence W. de Silva, and Chenglong Fu. Linked dynamic graph cnn: Learning on point cloud via linking hierarchical features. *ArXiv*, abs/1904.10014, 2019.

[142] Yifan Xu, Tianqi Fan, Mingye Xu, Long Zeng, and Yu Qiao. Spidercnn: Deep learning on point sets with parameterized convolutional filters. *CoRR*, abs/1803.11527, 2018.

[143] Saining Xie, Sainan Liu, Zeyu Chen, and Zhuowen Tu. Attentional shapecontextnet for point cloud recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[144] Hengshuang Zhao, Li Jiang, Chi-Wing Fu, and Jiaya Jia. Pointweb: Enhancing local neighborhood features for point cloud processing. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[145] Ji Hou, Benjamin Graham, Matthias Nießner, and Saining Xie. Exploring data-efficient 3d scene understanding with contrastive scene contexts. *CoRR*, abs/2012.09165, 2020.

[146] Saining Xie, Jiatao Gu, Demi Guo, Charles R. Qi, Leonidas J. Guibas, and Or Litany. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. *CoRR*, abs/2007.10985, 2020.

[147] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[148] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, pages 604–621. Springer, 2022.

[149] 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. *arXiv preprint arXiv:2205.14401*, 2022.

[150] Zekun Qi, Runpei Dong, Guofan Fan, Zheng Ge, Xiangyu Zhang, Kaisheng Ma, and Li Yi. Contrast with reconstruct: Contrastive 3d representation learning guided by generative pretraining. In *International Conference on Machine Learning (ICML)*, 2023.

[151] Xiaoyang Wu, Xin Wen, Xihui Liu, and Hengshuang Zhao. Masked scene contrast: A scalable framework for unsupervised 3d representation learning. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9415–9424. IEEE, June 2023.

[152] Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. Ulip: Learning a unified representation of language, images, and point clouds for 3d understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1179–1189, 2023.

[153] Le Xue, Ning Yu, Shu Zhang, Artemis Panagopoulou, Junnan Li, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, et al. Ulip-2: Towards scalable multimodal pre-training for 3d understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27091–27101, 2024.

[154] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. Pointclip: Point cloud understanding by clip. *arXiv preprint arXiv:2112.02413*, 2021.

[155] Xiangyang Zhu, Renrui Zhang, Bowei He, Ziyu Guo, Ziyao Zeng, Zipeng Qin, Shanghang Zhang, and Peng Gao. Pointclip v2: Prompting clip and gpt for powerful 3d open-world learning. *arXiv preprint arXiv:2211.11682*, 2022.

[156] Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[157] Ayça Takmaz, Elisabetta Fedele, Robert W. Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. OpenMask3D: Open-Vocabulary 3D Instance Segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[158] Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. *arXiv preprint arXiv:2507.13347*, 2025.

[159] Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, Jonathon Luiten, Manuel Lopez-Antequera, Samuel Rota Bulò, Christian Richardt, Deva Ramanan, Sebastian Scherer, and Peter Kontschieder. MapAnything: Universal feed-forward metric 3D reconstruction, 2025. arXiv preprint arXiv:2509.13414.

[160] Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In *ECCV*, 2024.

[161] Yuan Tang, Xu Han, Xianzhi Li, Qiao Yu, Yixue Hao, Long Hu, and Min Chen. Minigpt-3d: Efficiently aligning 3d point clouds with large language models using 2d priors. *arXiv preprint arXiv:2405.01413*, 2024.

[162] Zekun Qi, Runpei Dong, Shaochen Zhang, Haoran Geng, Chunrui Han, Zheng Ge, Li Yi, and Kaisheng Ma. Shapellm: Universal 3d object understanding for embodied interaction. *arXiv preprint arXiv:2402.17766*, 2024.

[163] Junliang Ye, Zhengyi Wang, Ruowen Zhao, Shenghao Xie, and Jun Zhu. Shapellm-omni: A native multimodal llm for 3d generation and understanding. *arXiv preprint arXiv:2506.01853*, 2025.

[164] Yongsen Mao, Junhao Zhong, Chuan Fang, Jia Zheng, Rui Tang, Hao Zhu, Ping Tan, and Zihan Zhou. Spatiallm: Training large language models for structured indoor modeling. In *Advances in Neural Information Processing Systems*, 2025.

[165] Zhiwen Fan, Jian Zhang, Renjie Li, Junge Zhang, Runjin Chen, Hezhen Hu, Kevin Wang, Huaizhi Qu, Dilin Wang, Zhicheng Yan, Hongyu Xu, Justin Theiss, Tianlong Chen, Jiachen Li, Zhengzhong Tu, Zhangyang Wang, and Rakesh Ranjan. Vlm-3r: Vision-language models augmented with instruction-aligned 3d reconstruction, 2025.

[166] Yi Kun and Wu Jianxin. Probabilistic End-to-end Noise Correction for Learning with Noisy Labels. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[167] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–19, 2022.

[168] Xun Xu and Gim Hee Lee. Weakly supervised semantic point cloud segmentation: Towards 10x fewer labels. In *CVPR*, 2020.

[169] Hyeokjun Kweon and Kuk-Jin Yoon. Joint learning of 2d-3d weakly supervised semantic segmentation. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[170] Biao Gao, Yancheng Pan, Chengkun Li, Sibo Geng, and Huijing Zhao. Are we hungry for 3d lidar data for semantic segmentation? a survey of datasets and methods. *IEEE Transactions on Intelligent Transportation Systems*, 23(7):6063–6081, Jul 2022.

[171] Can Qin, Haoxuan You, Lichen Wang, C. C. Jay Kuo, and Yun Fu. Pointdan: A multi-scale 3d domain adaption network for point cloud representation, 2019.

[172] Maximilian Jaritz, Tuan-Hung Vu, Raoul de Charette, Emilie Wirbel, and Patrick Pérez. xMUDA: Cross-modal unsupervised domain adaptation for 3D semantic segmentation. In *CVPR*, 2020.

[173] Cristiano Saltori, Fabio Galasso, Giuseppe Fiameni, Nicu Sebe, Elisa Ricci, and Fabio Poiesi. Cosmix: Compositional semantic mix for domain adaptation in 3d lidar segmentation. In *European Conference on Computer Vision*, pages 586–602. Springer, 2022.

[174] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, Jan. 2023.

[175] Zhiqiang Hu, Yihuai Lan, Lei Wang, Wanyu Xu, Ee-Peng Lim, Roy Ka-Wei Lee, Lidong Bing, and Soujanya Poria. Llm-adapters: An adapter family for parameter-efficient fine-tuning of large language models. *arXiv preprint arXiv:2304.01933*, 2023.

[176] Yaohua Zha, Jinpeng Wang, Tao Dai, Bin Chen, Zhi Wang, and Shu-Tao Xia. Instance-aware dynamic prompt tuning for pre-trained point cloud models. In *Proceedings of*

the IEEE/CVF International Conference on Computer Vision (ICCV), 2023.

[177] Yiwen Tang, Ray Zhang, Zoey Guo, Xianzheng Ma, Bin Zhao, Zhigang Wang, Dong Wang, and Xuelong Li. Point-peft: Parameter-efficient fine-tuning for 3d pre-trained models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 5171–5179, 2024.

[178] Hong Joo Lee, Jung Uk Kim, Sangmin Lee, Hak Gu Kim, and Yong Man Ro. Structure boundary preserving segmentation for medical image with ambiguous boundary. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[179] Bowen Cheng, Ross Girshick, Piotr Dollár, Alexander C. Berg, and Alexander Kirillov. Boundary IoU: Improving object-centric image segmentation evaluation. In *CVPR*, 2021.

[180] Jianlong Yuan, Zelu Deng, Shu Wang, and Zhenbo Lui. Multi receptive field network for semantic segmentation. *CoRR*, abs/2011.08577, 2020.

[181] R. B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE International Conference on Robotics and Automation*, pages 3212–3217, 2009.

[182] Loïc Landrieu, Hugo Raguet, Bruno Vallet, Clément Mallet, and Martin Weinmann. A structured regularization framework for spatially smoothing semantic labelings of 3d point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 132:102–118, 2017.

[183] Chenfeng Xu, Bichen Wu, Zining Wang, Wei Zhan, Peter Vajda, Kurt Keutzer, and Masayoshi Tomizuka. Squeezesegv3: Spatially-adaptive convolution for efficient point-cloud segmentation. *ArXiv*, abs/2004.01803, 2020.

[184] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss. Rangenet++: Fast and accurate lidar semantic segmentation. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4213–4220, 2019.

[185] Abhijit Kundu, Xiaoqi Yin, Alireza Fathi, David A. Ross, Brian Brewington, Thomas A. Funkhouser, and Caroline Pantofaru. Virtual multi-view fusion for 3d semantic segmentation. In *ECCV*, 2020.

[186] Zhe Chen, Jing Zhang, and Dacheng Tao. Progressive lidar adaptation for road detection. *IEEE/CAA Journal of Automatica Sinica*, 6:693–702, 05 2019.

[187] Peng-Shuai Wang, Yang Liu, Yu-Xiao Guo, Chun-Yu Sun, and Xin Tong. O-CNN: Octree-based Convolutional Neural Networks for 3D Shape Analysis. *ACM Transactions on Graphics (SIGGRAPH)*, 36(4), 2017.

[188] Lei Han, Tian Zheng, Lan Xu, and Lu Fang. Occuseg: Occupancy-aware 3d instance segmentation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2937–2946, 2020.

[189] Meng-Hao Guo, Junxiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph R. Martin, and Shi-Min Hu. PCT: point cloud transformer. *CoRR*, abs/2012.09688, 2020.

[190] L. Landrieu and M. Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4558–4567, 2018.

[191] Xu Yan, Chaoda Zheng, Zhen Li, Sheng Wang, and Shuguang Cui. Pointasnl: Robust point clouds processing using nonlocal neural networks with adaptive sampling. *CoRR*, abs/2003.00492, 2020.

[192] Shi Qiu, Saeed Anwar, and Nick Barnes. Semantic segmentation for real point cloud scenes via bilateral augmentation and adaptive fusion. *CoRR*, abs/2103.07074, 2021.

[193] Nina Varney, Vijayan K. Asari, and Quinn Graehling. Pyramid point: A multi-level focusing network for revisiting feature layers, 2020.

[194] Tao Lu, Limin Wang, and Gangshan Wu. Cga-net: Category guided aggregation for point cloud semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11693–11702, June 2021.

[195] Zeyu Hu, Mingmin Zhen, Xuyang Bai, Hongbo Fu, and Chiew-Lan Tai. Jsenet: Joint semantic segmentation and edge detection network for 3d point clouds. *CoRR*, abs/2007.06888, 2020.

[196] Dmitrii Marin, Zijian He, Peter Vajda, Priyam Chatterjee, Sam Tsai, Fei Yang, and Yuri Boykov. Efficient segmentation: Learning downsampling near semantic boundaries. In *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[197] Jingyu Gong, Jiachen Xu, Xin Tan, Jie Zhou, Yanyun Qu, Yuan Xie, and Lizhuang Ma. Boundary-aware geometric encoding for semantic segmentation of point clouds, 2021.

[198] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2018.

[199] Nicholas Frosst, Nicolas Papernot, and Geoffrey E. Hinton. Analyzing and improving representations with the soft nearest neighbor loss. In *ICML*, 2019.

[200] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning, 2020.

[201] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019.

[202] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[203] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training, 2020.

[204] Wenguan Wang, Tianfei Zhou, Fisher Yu, Jifeng Dai, Ender Konukoglu, and Luc Van Gool. Exploring cross-image pixel contrast for semantic segmentation. *CoRR*, abs/2101.11939, 2021.

[205] Yunze Liu, Li Yi, Shanghang Zhang, Qingnan Fan, Thomas A. Funkhouser, and Hao Dong. P4contrast: Contrastive learning with pairs of point-pixel pairs for RGB-D scene understanding. *CoRR*, abs/2012.13089, 2020.

[206] Ze Liu, Han Hu, Yue Cao, Zheng Zhang, and Xin Tong. A closer look at local aggregation operators in point cloud analysis. *ECCV*, 2020.

[207] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. *Journal of Machine Learning Research - Proceedings Track*, 9:297–304, 01 2010.

[208] Jingyu Gong, Jiachen Xu, Xin Tan, Haichuan Song, Yanyun Qu, Yuan Xie, and Lizhuang Ma. Omni-supervised point cloud segmentation via gradual receptive field component reasoning. *CoRR*, abs/2105.10203, 2021.

[209] Qiangui Huang, Weiyue Wang, and Ulrich Neumann. Recurrent slice networks for 3d segmentation of point clouds. *2018 IEEE/CVF Conference on Computer Vision and*

Pattern Recognition, pages 2626–2635, 2018.

[210] Siqi Fan, Qiulei Dong, Fenghua Zhu, Yisheng Lv, Peijun Ye, and Fei-Yue Wang. Scf-net: Learning spatial contextual features for large-scale point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14504–14513, June 2021.

[211] G. Truong, S. Z. Gilani, S. M. S. Islam, and D. Suter. Fast point cloud registration using semantic segmentation. In *2019 Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8, 2019.

[212] Jonas Schult, Francis Engelmann, Theodora Kontogianni, and Bastian Leibe. Dualconvmesh-net: Joint geodesic and euclidean convolutions on 3d meshes. *CoRR*, abs/2004.01002, 2020.

[213] Zeyu Hu, Xuyang Bai, Jiaxiang Shang, Runze Zhang, Jiayu Dong, Xin Wang, Guangyuan Sun, Hongbo Fu, and Chiew-Lan Tai. Vmnet: Voxel-mesh network for geodesic-aware 3d semantic segmentation. *CoRR*, abs/2107.13824, 2021.

[214] Alexey Nekrasov, Jonas Schult, Or Litany, Bastian Leibe, and Francis Engelmann. Mix3d: Out-of-context data augmentation for 3d scenes, 2021.

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

[216] Zhidong Liang, Ming Yang, Liuyuan Deng, Chunxiang Wang, and Bing Wang. Hierarchical depthwise graph convolutional neural network for 3d semantic segmentation of point clouds. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8152–8158, 2019.

[217] Alexandre Boulch, Gilles Puy, and Renaud Marlet. Fkaconv: Feature-kernel alignment for point cloud convolution, 2020.

[218] A. Xiao, J. Huang, D. Guan, X. Zhang, S. Lu, and L. Shao. Unsupervised point cloud representation learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(09):11321–11339, sep 2023.

[219] Jiacheng Wei, Guosheng Lin, Kim-Hui Yap, Tzu-Yi Hung, and Lihua Xie. Multi-path region mining for weakly supervised 3d semantic segmentation on point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4384–4393, 2020.

[220] Cheng Jin, Zhengrui Guo, Yi Lin, Luyang Luo, and Hao Chen. Label-efficient deep learning in medical image analysis: Challenges and future directions, 2023.

[221] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. *Microsoft COCO: Common Objects in Context*, pages 740–755. Springer International Publishing, 2014.

[222] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[223] Tao Yu, Runseng Feng, Ruoyu Feng, Jinming Liu, Xin Jin, Wenjun Zeng, and Zhibo Chen. Inpaint anything: Segment anything meets image inpainting. *arXiv preprint arXiv:2304.06790*, 2023.

[224] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2023.

[225] Yachao Zhang, Zonghao Li, Yuan Xie, Yanyun Qu, Cuihua Li, and Tao Mei. Weakly supervised semantic segmentation for large-scale point cloud. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 3421–3429, 2021.

[226] Qingyong Hu, Bo Yang, Guangchi Fang, Yulan Guo, Ales Leonardis, Niki Trigoni, and Andrew Markham. Sqn: Weakly-supervised semantic segmentation of large-scale 3d point clouds. In *European Conference on Computer Vision*, 2022.

[227] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 596–608. Curran Associates, Inc., 2020.

[228] Mengtian Li, Yuan Xie, Yunhang Shen, Bo Ke, Ruizhi Qiao, Bo Ren, Shaohui Lin, and Lizhuang Ma. Hybridcr: Weakly-supervised 3d point cloud semantic segmentation via hybrid contrastive regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14930–14939, June 2022.

[229] Cheng-Kun Yang, Ji-Jia Wu, Kai-Syun Chen, Yung-Yu Chuang, and Yen-Yu Lin. An mil-derived transformer for weakly supervised point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11830–11839, June 2022.

[230] Guo-Jun Qi and Jiebo Luo. Small data challenges in big data era: A survey of recent progress on unsupervised and semi-supervised methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(4):2168–2187, Apr. 2022.

[231] Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. In *International Conference on Learning Representations*, 2021.

[232] Yanning Zhou, Hang Xu, Wei Zhang, Bin Gao, and Pheng-Ann Heng. C3-semiseg: Contrastive semi-supervised segmentation via cross-set learning and dynamic class-balancing. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7016–7025, 2021.

[233] Hao Huang, Jack Chu, and Fangyun Wei. Unsupervised prompt learning for vision-language models. *ArXiv*, abs/2204.03649, 2022.

[234] Lihe Yang, Lei Qi, Litong Feng, Wayne Zhang, and Yinghuan Shi. Revisiting weak-to-strong consistency in semi-supervised semantic segmentation. In *CVPR*, 2023.

[235] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, Jan. 2015.

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

[237] Haibo Qiu, Baosheng Yu, and Dacheng Tao. Collect-and-distribute transformer for 3d point cloud analysis. *arXiv preprint arXiv:2306.01257*, 2023.

[238] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2921–2929, 2016.

[239] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. *2016 IEEE Conference*

on Computer Vision and Pattern Recognition (CVPR), pages 3159–3167, 2016.

[240] Yuchao Wang, Haochen Wang, Yujun Shen, Jingjing Fei, Wei Li, Guoqiang Jin, Liwei Wu, Rui Zhao, and Xinyi Le. Semi-supervised semantic segmentation using unreliable pseudo-labels. *ArXiv*, abs/2203.03884, 2022.

[241] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4981–4990, 2018.

[242] Julian Chibane, Francis Engelmann, Tuan Anh Tran, and Gerard Pons-Moll. Box2mask: Weakly supervised 3d semantic instance segmentation using bounding boxes. *Computer Vision - ECCV 2022*, pages 681–699, 2022.

[243] Yongbin Liao, Hongyuan Zhu, Yanggang Zhang, Chuangguan Ye, Tao Chen, and Jiayuan Fan. Point cloud instance segmentation with semi-supervised bounding-box mining. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):10159–10170, Dec. 2022.

[244] Zhengzhe Liu, Xiaojuan Qi, and Chi-Wing Fu. One thing one click: A self-training approach for weakly supervised 3d semantic segmentation. *CoRR*, abs/2104.02246, 2021.

[245] Yachao Zhang, Yanyun Qu, Yuan Xie, Zonghao Li, Shanshan Zheng, and Cuihua Li. Perturbed self-distillation: Weakly supervised large-scale point cloud semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15520–15528, 2021.

[246] Hanyu Shi, Jiacheng Wei, Ruibo Li, Fayao Liu, and Guosheng Lin. Weakly supervised segmentation on outdoor 4d point clouds with temporal matching and spatial graph propagation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11830–11839, 2022.

[247] Zhonghua Wu, Yicheng Wu, Guosheng Lin, and Jianfei Cai. Reliability-adaptive consistency regularization for weakly-supervised point cloud segmentation, 2023.

[248] Zhonghua Wu, Yicheng Wu, Guosheng Lin, Jianfei Cai, and Chen Qian. Dual adaptive transformations for weakly supervised point cloud segmentation. *arXiv preprint arXiv:2207.09084*, 2022.

[249] Yuxiang Lan, Yachao Zhang, Yanyun Qu, Cong Wang, Chengyang Li, Jia Cai, Yuan Xie, and Zongze Wu. Weakly supervised 3d segmentation via receptive-driven pseudo label consistency and structural consistency. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(1):1222–1230, Jun. 2023.

[250] Li Jiang, Shaoshuai Shi, Zhuotao Tian, Xin Lai, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Guided point contrastive learning for semi-supervised point cloud semantic segmentation. In *ICCV*, pages 6403–6412, 10 2021.

[251] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[252] Xin Tan, Qihang Ma, Jingyu Gong, Jiachen Xu, Zhizhong Zhang, Haichuan Song, Yanyun Qu, Yuan Xie, and Lizhuang Ma. Positive-negative receptive field reasoning for omni-supervised 3d segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12):15328–15344, 2023.

[253] Mingmei Cheng, Le Hui, Jin Xie, and Jian Yang. Sspc-net: Semi-supervised semantic 3d point cloud segmentation network. 2021.

[254] Xiaojin Zhu and Andrew B. Goldberg. *Introduction to Semi-Supervised Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2009.

[255] Hieu Pham, Zihang Dai, Qizhe Xie, and Quoc V. Le. Meta pseudo labels. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2021.

[256] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. *arXiv preprint arXiv:1911.04252*, 2019.

[257] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D. Cubuk, and Quoc V. Le. Rethinking pre-training and self-training. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[258] David Berthelot, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *arXiv preprint arXiv:1911.09785*, 2019.

[259] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6256–6268. Curran Associates, Inc., 2020.

[260] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NIPS*, 2017.

[261] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 1171–1179, Red Hook, NY, USA, 2016. Curran Associates Inc.

[262] Junnan Li, Caiming Xiong, and Steven C.H. Hoi. Semi-supervised learning with contrastive graph regularization. In *ICCV*, 2021.

[263] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*, 2017.

[264] Chengyue Gong, Dilin Wang, and Qiang Liu. Alphamatch: Improving consistency for semi-supervised learning with alpha-divergence. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13678–13687, 2021.

[265] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *NeurIPS*, 2019.

[266] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. 2021.

[267] Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, , Zhen Wu, Jindong Wang, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, Bernt Schiele, and Xing Xie. Freematch: Self-adaptive thresholding for semi-supervised learning. 2023.

[268] Hao Chen, Ran Tao, Yue Fan, Yidong Wang, Jindong Wang, Bernt Schiele, Xing Xie, Bhiksha Raj, and Marios Savvides. Softmatch: Addressing the quantity-quality trade-off in semi-supervised learning. 2023.

[269] Lihe Yang, Wei Zhuo, Lei Qi, Yinghuan Shi, and Yang Gao. St++: Make self-trainingwork better for semi-supervised semantic segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2022.

[270] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2021.

[271] Yuyuan Liu, Yu Tian, Yuanhong Chen, Fengbei Liu, Vasileios Belagiannis, and Gustavo Carneiro. Perturbed and strict mean teachers for semi-supervised semantic segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2022.

[272] Hai-Ming Xu, Lingqiao Liu, Qiuchen Bian, and Zhen Yang. Semi-supervised semantic segmentation with prototype-based consistency regularization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[273] Shikun Liu, Shuaifeng Zhi, Edward Johns, and Andrew J Davison. Bootstrapping semantic segmentation with regional contrast. In *International Conference on Learning Representations*, 2022.

[274] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. CutMix: Regularization strategy to train strong classifiers with localizable features. In *International Conference on Computer Vision (ICCV)*, 2019.

[275] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. ClassMix: Segmentation-Based Data Augmentation for Semi-Supervised Learning . In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1368–1377, Los Alamitos, CA, USA, Jan. 2021. IEEE Computer Society.

[276] Terrance Devries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout. *CoRR*, abs/1708.04552, 2017.

[277] Xide Xia and Brian Kulis. W-net: A deep model for fully unsupervised image segmentation, 2017.

[278] Xu Ji, João F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9865–9874, 2019.

[279] Jang Hyun Cho, Utkarsh Mall, Kavita Bala, and Bharath Hariharan. Picie: Unsupervised semantic segmentation using invariance and equivariance in clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16794–16804, June 2021.

[280] Tsung-Wei Ke, Jyh-Jing Hwang, Yunhui Guo, Xudong Wang, and Stella X. Yu. Unsupervised hierarchical semantic segmentation with multiview cosegmentation and clustering transformers. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, page 2561–2571. IEEE, June 2022.

[281] Tim Lebailly, Thomas Stegmüller, Behzad Bozorgtabar, Jean-Philippe Thiran, and Tinne Tuytelaars. Cribo: Self-supervised learning via cross-image object-level bootstrapping, 2023.

[282] Robert Harb and Patrick Knöbelreiter. *InfoSeg: Unsupervised Semantic Image Segmentation with Mutual Information Maximization*, page 18–32. Springer International Publishing, 2021.

[283] Yin Zhaoyun, Wang Pichao, Wang Fan, Xu Xianzhe, Zhang Hanling, Li Hao, and Jin Rong. Transfgu: A top-down approach to fine-grained unsupervised semantic segmentation. In *European Conference on Computer Vision*, pages 73–89. Springer, 2022.

[284] Mark Hamilton, Zhoutong Zhang, Bharath Hariharan, Noah Snavely, and William T. Freeman. Unsupervised semantic segmentation by distilling feature correspondences. In *International Conference on Learning Representations*, 2022.

[285] Adrian Ziegler and Yuki M. Asano. Self-supervised learning of object parts for semantic segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, page 14482–14491. IEEE, June 2022.

[286] Zhedong Zheng and Yi Yang. Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation. *International Journal of Computer Vision*, 129(4):1106–1120, Jan 2021.

[287] Yuxi Wang, Junran Peng, and Zhaoxiang Zhang. Uncertainty-aware pseudo label refinery for domain adaptive semantic segmentation. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9072–9081, 2021.

[288] Zhengyang Feng, Qianyu Zhou, Qiqi Gu, Xin Tan, Guangliang Cheng, Xuequan Lu, Jianping Shi, and Lizhuang Ma. Dmt: Dynamic mutual training for semi-supervised learning. *Pattern Recognition*, 130:108777, Oct 2022.

[289] Yuyuan Liu, Yu Tian, Yuanhong Chen, Fengbei Liu, Vasileios Belagiannis, and Gustavo Carneiro. Perturbed and strict mean teachers for semi-supervised semantic segmentation. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2022.

[290] Guo-Hua Wang and Jianxin Wu. Repetitive reprediction deep decipher for semi-supervised learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):6170–6177, Apr. 2020.

[291] Pan Zhang, Bo Zhang, Ting Zhang, Dong Chen, and Fang Wen. Robust mutual learning for semi-supervised semantic segmentation, 2021.

[292] Ruifei He, Jihan Yang, and Xiaojuan Qi. Re-distributing biased pseudo labels for semi-supervised semantic segmentation: A baseline investigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6930–6940, 2021.

[293] Donghyeon Kwon and Suha Kwak. Semi-supervised semantic segmentation with error localization network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9957–9967, June 2022.

[294] Yan Zhang, Wenhan Zhao, Bo Sun, Ying Zhang, and Wen Wen. Point cloud upsampling algorithm: A systematic review. *Algorithms*, 15(4), 2022.

[295] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *ArXiv*, abs/2006.09882, 2020.

[296] Claude Elwood Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 1948.

[297] Xiaolong Zhang, Zuqiang Su, Xiaolin Hu, Yan Han, and Shuxian Wang. Semisupervised momentum prototype network for gearbox fault diagnosis under limited labeled samples. *IEEE Transactions on Industrial Informatics*, 18(9):6203–6213, 2022.

[298] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, 2017.

[299] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[300] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 18613–18624. Curran Associates, Inc., 2020.

[301] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021.

[302] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021.

[303] Mengcheng Lan, Xinjiang Wang, Yiping Ke, Jiaxing Xu, Litong Feng, and Wayne Zhang. Smooseg: Smoothness prior for unsupervised semantic segmentation. In *NeurIPS*, 2023.

[304] Lequan Yu, Shujun Wang, Xiaomeng Li, Chi-Wing Fu, and Pheng-Ann Heng. Uncertainty-aware self-ensembling model for semi-supervised 3d left atrium segmentation. In *MICCAI*, 2019.

[305] Xiangde Luo, Minhao Hu, Tao Song, Guotai Wang, and Shaoting Zhang. Semi-supervised medical image segmentation via cross teaching between CNN and transformer. In *Medical Imaging with Deep Learning*, 2022.

[306] Zhuo Huang, Zhiyou Zhao, Banghuai Li, and Jungong Han. Lcpformer: Towards effective 3d point cloud analysis via local context propagation in transformers. *ArXiv*, abs/2210.12755, 2022.

[307] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository, 2015.

[308] Olivier Bernard, Alain Lalande, Clement Zotti, Frederick Cervenansky, Xin Yang, Pheng-Ann Heng, Irem Cetin, Karim Lekadir, Oscar Camara, Miguel Angel Gonzalez Ballester, Gerard Sanroma, Sandy Napel, Steffen Petersen, Georgios Tziritas, Elias Grinias, Mahendra Khened, Varghese Alex Kollerathu, Ganapathy Krishnamurthi, Marc-Michel Rohé, Xavier Pennec, Maxime Sermesant, Fabian Isensee, Paul Jäger, Klaus H. Maier-Hein, Peter M. Full, Ivo Wolf, Sandy Engelhardt, Christian F. Baumgartner, Lisa M. Koch, Jelmer M. Wolterink, Ivana Išgum, Yeonggul Jang, Yoonmi Hong, Jay Patravali, Shubham Jain, Olivier Humbert, and Pierre-Marc Jodoin. Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: Is the problem solved? *IEEE Transactions on Medical Imaging*, 37(11):2514–2525, 2018.

[309] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020.

[310] Jean-Bastien Grill, Florian Strub, Florent Altch'e, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. *ArXiv*, abs/2006.07733, 2020.

[311] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[312] DeepSeek-AI. Deepseek-v3 technical report, 2025.

[313] Qwen. Qwen2.5 technical report, 2025.

[314] OpenAI. Hello gpt-4o, 2024.

[315] Ankur Bapna and Orhan Firat. Simple, scalable adaptation for neural machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, 2019.

[316] Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. Adapterfusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Association for Computational Linguistics, 2021.

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

[318] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2021.

[319] Angela Dai, Daniel Ritchie, Martin Bokeloh, Scott Reed, Jürgen Sturm, and Matthias Nießner. Scancomplete: Large-scale scene completion and semantic segmentation for 3d scans. *CoRR*, abs/1712.10215, 2017.

[320] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020.

[321] Liyao Tang, Zhe Chen, Shanshan Zhao, Chaoyue Wang, and Dacheng Tao. All points matter: Entropy-regularized distribution alignment for weakly-supervised 3d segmentation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[322] Liyao Tang, Zhe Chen, Shanshan Zhao, Chaoyue Wang, and Dacheng Tao. Towards modality-agnostic label-efficient segmentation with entropy-regularized distribution alignment, 2024.

[323] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023.

[324] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual

recognition. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 647–655, Bejing, China, 22–24 Jun 2014. PMLR.

[325] Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. BitFit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1–9, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[326] Zhi Zhang, Qizhe Zhang, Zijun Gao, Renrui Zhang, Ekaterina Shutova, Shiji Zhou, and Shanghang Zhang. Gradient-based parameter selection for efficient fine-tuning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 28566–28577. IEEE, June 2024.

[327] Sylvestre-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi. Learning multiple visual domains with residual adapters. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 506–516, Red Hook, NY, USA, 2017. Curran Associates Inc.

[328] Jan-Martin O. Steitz and Stefan Roth. Adapters strike back. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, page 23449–23459. IEEE, June 2024.

[329] Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. In *International Conference on Learning Representations*, 2022.

[330] Yaoming Zhu, Jiangtao Feng, Chengqi Zhao, Mingxuan Wang, and Lei Li. Counter-interference adapter for multilingual machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, page 2812–2823. Association for Computational Linguistics, 2021.

[331] Chin-Lun Fu, Zih-Ching Chen, Yun-Ru Lee, and Hung-yi Lee. AdapterBias: Parameter-efficient token-dependent representation shift for adapters in NLP tasks. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz, editors, *Findings*

of the Association for Computational Linguistics: NAACL 2022, pages 2608–2621, Seattle, United States, July 2022. Association for Computational Linguistics.

[332] Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. *arXiv preprint arXiv:2205.13535*, 2022.

[333] Yaqing Wang, Sahaj Agarwal, Subhabrata Mukherjee, Xiaodong Liu, Jing Gao, Ahmed Hassan Awadallah, and Jianfeng Gao. Adamix: Mixture-of-adaptations for parameter-efficient model tuning. *arXiv preprint arXiv:2205.12410*, 2022.

[334] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. Exploring visual prompts for adapting large-scale models, 2022.

[335] Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 61–68, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[336] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision (ECCV)*, 2022.

[337] Damjan Kalajdzievski. A rank stabilization scaling factor for fine-tuning with lora, 2023.

[338] Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: weight-decomposed low-rank adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

[339] Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adaptive budget allocation for parameter-efficient fine-tuning. In *The Eleventh International Conference on Learning Representations*, 2023.

[340] Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D Manning, and Christopher Potts. ReFT: Representation finetuning for language models. In *The Thirty-eighth Annual Conference on Neural Information Processing*

Systems, 2024.

[341] Haokun Liu, Derek Tam, Muqeeth Mohammed, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[342] Xin Zhou, Dingkang Liang, Wei Xu, Xingkui Zhu, Yihan Xu, Zhikang Zou, and Xiang Bai. Dynamic adapter meets prompt tuning: Parameter-efficient transfer learning for point cloud analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14707–14717, 2024.

[343] Zixiang Ai, Zichen Liu, Yuanhang Lei, Zhenyu Cui, Xu Zou, and Jiahuan Zhou. Gaprompt: Geometry-aware point cloud prompt for 3d vision model, 2025.

[344] Shaochen Zhang, Zekun Qi, Runpei Dong, Xiuxiu Bai, and Xing Wei. Positional prompt tuning for efficient 3d representation learning, 2024.

[345] Dingkang Liang, Tianrui Feng, Xin Zhou, Yumeng Zhang, Zhikang Zou, and Xiang Bai. Parameter-efficient fine-tuning in spectral domain for point cloud learning. *arXiv preprint arXiv:2410.08114*, 2024.

[346] Takahiko Furuya. Token adaptation via side graph convolution for efficient fine-tuning of 3d point cloud transformers, 2025.

[347] Song Wang, Xiaolu Liu, Lingdong Kong, Jianyun Xu, Chunyong Hu, Gongfan Fang, Wentong Li, Jianke Zhu, and Xinchao Wang. Pointlora: Low-rank adaptation with token selection for point cloud learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6605–6615, 2025.

[348] Hongyu Sun, Yongcai Wang, Wang Chen, Haoran Deng, and Deying Li. Parameter-efficient prompt learning for 3d point cloud understanding. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[349] Xiangxiang Chu, Zhi Tian, Bo Zhang, Xinlong Wang, and Chunhua Shen. Conditional positional encodings for vision transformers. In *The Eleventh International Conference on Learning Representations*, 2023.

[350] Md Amirul Islam*, Sen Jia*, and Neil D. B. Bruce. How much position information do convolutional neural networks encode? In *International Conference on Learning Representations*, 2020.

[351] Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. Peft: State-of-the-art parameter-efficient fine-tuning methods. https://github.com/huggingface/peft, 2022.

[352] Clifton Poth, Hannah Sterz, Indraneil Paul, Sukannya Purkayastha, Leon Engländer, Timo Imhof, Ivan Vulić, Sebastian Ruder, Iryna Gurevych, and Jonas Pfeiffer. Adapters: A unified library for parameter-efficient and modular transfer learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 149–160, Singapore, Dec. 2023. Association for Computational Linguistics.

[353] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 3320–3328, Cambridge, MA, USA, 2014. MIT Press.

[354] Zirui Wang, Zihang Dai, Barnabas Poczos, and Jaime Carbonell. Characterizing and avoiding negative transfer. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, page 11285–11294. IEEE, June 2019.

[355] Andrew Jaegle, Felix Gimeno, Andrew Brock, Andrew Zisserman, Oriol Vinyals, and Joao Carreira. Perceiver: General perception with iterative attention, 2021.

[356] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3744–3753. PMLR, 09–15 Jun 2019.

[357] Kadir Yilmaz, Jonas Schult, Alexey Nekrasov, and Bastian Leibe. Mask4former: Mask transformer for 4d panoptic segmentation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9418–9425. IEEE, 2024.

[358] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.

[359] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference*

on Learning Representations, 2021.

[360] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of Conference on Computer Vision and Pattern Recognition*, 2022.

[361] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3D: Mask Transformer for 3D Semantic Instance Segmentation. 2023.

[362] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *NeurIPS*, 2023.