Eurasia Journal of Science and Technology
Print ISSN: 2663-1024
Online ISSN: 2663-1016
DOI: https://doi.org/10.61784/ejst3146

TOWARDS FOUNDATION MODELS FOR LIDAR SEMANTIC SEGMENTATION IN AUTONOMOUS DRIVING

YiFan Zhao*, ZiWei Huang
School of Information Science and Engineering, Hunan Institute of Engineering, Xiangtan 411100, Hunan, China.
**Corresponding Author: YiFan Zhao*

Abstract: LiDAR semantic segmentation (LSS), which assigns semantic labels to each point in a 3D scan, is a core perception task in autonomous driving. Over the past decade, fully supervised methods have achieved remarkable progress, with benchmark performance on SemanticKITTI improving from 14.6 mIoU in 2017 to over 75 mIoU in recent state-of-the-art models. Despite these advances, conventional supervised paradigms remain constrained by three fundamental limitations: dependence on large-scale dense annotations, restricted closed-set semantic understanding, and limited robustness under domain shifts and adverse environments.Recent advances in foundation models—including vision-language models, self-supervised pretraining frameworks, and segmentation foundation models—have opened a new direction for LiDAR perception by enabling transferable, label-efficient, and open-vocabulary 3D understanding. Motivated by this paradigm shift, this survey provides a systematic review of LiDAR semantic segmentation from supervised learning to foundation-model-driven approaches. We organize existing methods into five representative paradigms: cross-modal 2D-to-3D feature distillation, Segment Anything Model (SAM)-guided segmentation, open-vocabulary vision-language learning, LiDAR-specific self-supervised pretraining, and generalized 3D foundation models.Beyond taxonomy and benchmark comparison on SemanticKITTI and nuScenes, we further examine practical deployment factors—including inference latency, edge-device efficiency, robustness in adverse weather, and multimodal sensor fusion—that remain insufficiently captured by standard evaluation protocols. Finally, we identify six open research challenges and argue that the field is undergoing a fundamental transition: from adapting 2D foundation priors to developing native 3D LiDAR foundation models for autonomous driving.
Keywords: LiDAR semantic segmentation; Foundation models; Autonomous driving; Point cloud; Self-supervised learning

1 INTRODUCTION

1.1 Background

Autonomous driving (AD) has advanced from highway-only ADAS to vehicles operating at SAE Level 4 inside geofenced urban areas, but robust perception under unconstrained conditions remains the principal obstacle to wider deployment. A perception module must convert raw sensor measurements into a semantic description at frame rate, with negligible margin for systematic failure on safety-critical classes such as pedestrians, cyclists, and small obstacles. Among modern AD sensors—cameras, radars, and Light Detection and Ranging (LiDAR)—LiDAR holds a privileged role: it returns dense 3D point clouds that retain depth lost during projective imaging [1], and its active illumination remains operational at night and in low-contrast scenes.

LiDAR semantic segmentation (LSS) is the task of assigning a semantic class label to every point in such a sweep. A modern automotive LiDAR returns roughly 100,000 to 300,000 points per scan, organised in 32, 64, or 128 horizontal beams. The per-point labels feed directly into downstream modules: drivable-area extraction, dynamic-object isolation, occupancy-grid construction, and lane-level localisation. Errors that look small in aggregate metrics carry asymmetric consequences. A single pedestrian point silently mislabelled as background, for example, can propagate into a planning failure.

The community converged on a small set of large-scale benchmarks. SemanticKITTI extended the original KITTI driving sequences with point-wise annotations across 19 classes [2]. It comprises more than 43,000 scans recorded with a 64-beam Velodyne HDL-64E. The nuScenes dataset [3], originally constructed for 3D detection, was augmented in 2021 with the lidarseg task and 16-class point-level labels [4]. The Waymo Open Dataset added LiDAR semantic segmentation labels in 2022 [5]. ONCE expanded the data scale to one million scenes [6]. These four datasets anchor virtually every empirical comparison reported in this survey.

The methodological landscape evolved through three architectural waves. Point-based methods—PointNet [7], PointNet++ [8], KPConv [9], and RandLA-Net [10]—showed that permutation-invariant networks can operate directly on point sets at full LiDAR resolution. Voxel-based methods followed: MinkowskiNet used sparse 4D convolutions [11], Cylinder3D adopted a cylindrical partition aligned with rotating-LiDAR sampling [12], and SPVNAS interleaved point and voxel branches via neural architecture search [13]. Most recently, transformer designs—Stratified Transformer [14], SphereFormer [15], and Point Transformer V3 (PTv3) [16]—pushed SemanticKITTI validation mIoU to 75.5%, with PTv3 also winning the 2024 Waymo 3D Semantic Segmentation Challenge.

These methods share a common assumption: dense, fully-supervised, point-level annotations are available at training time. That assumption is increasingly difficult to maintain in practice.

1.2 Motivation

Three structural limitations of fully-supervised LSS have, over the last three years, motivated a shift toward foundation-model-based pipelines.

Annotation cost. Producing dense LiDAR segmentation labels is laborious. A recent label-efficiency study reports that fully annotating one LiDAR frame consumes roughly 88 minutes of human effort [17]. Even sparse "0.3%" weak-label settings still require 8 to 27 minutes per frame. At fleet scale, this translates into annotation budgets that exceed the cost of the sensors themselves. The cost is incurred again whenever a new class taxonomy or a new geographic domain is introduced.

Closed-vocabulary brittleness. Conventional LSS networks output probabilities over a class set fixed at training time. Anything outside that set — a fallen tree, a construction barrel, an electric scooter introduced after deployment — is either clipped to an unknown logit or, worse, silently aliased to an incorrect class. Open-vocabulary work in 2D vision suggests that closed-set assumptions are difficult to defend in safety-critical systems. Such systems must contend with the long tail of real-world scenes.

Domain shift. Models trained on clear-weather, mid-density LiDAR data degrade sharply when deployed elsewhere. The Cylinder3D baseline reaches 67.8 mIoU on SemanticKITTI but loses more than ten percentage points on SemanticSTF [18], a benchmark recorded under real rain, fog, and snow. Sensor changes (32-line versus 64-line versus 128-line LiDAR), geographic shifts, and synthetic-to-real gaps reproduce comparable drops [19].

These limitations cannot be resolved by collecting more labels alone. The 2D-vision community responded with foundation models—massively pretrained networks whose features transfer with minimal task-specific supervision: CLIP for open-vocabulary classification [20], ViT as a scalable architecture [21], MAE for reconstruction-based self-supervision [22], SAM and SAM2 for promptable segmentation [23-24], and DINOv2 for purely self-supervised dense features [25]. Section 2.4 details each.

Two new capabilities transferred to LSS: general-purpose visual features that move across tasks with little fine-tuning, and language- or prompt-conditioned segmentation over arbitrary classes. SLidR showed that distilling 2D features into a 3D LiDAR backbone lifts linear-probe nuScenes mIoU from 21.9 (PointContrast) to 38.8 by contrasting at superpixel granularity [26-27]; SEAL then raised it to 45.0 by switching to SAM/DINO-derived superpixels [28]. SAL cast the task as zero-shot text-prompted segmentation [29], recovering 42–54% of fully-supervised mIoU without any 3D labels, and SAM4D [30], the most recent entrant at ICCV 2025, extends SAM2 into a multimodal camera–LiDAR temporal foundation model reporting 55.7% mIoU on Waymo-4DSeg.

The trajectory is clear. Foundation models are no longer a peripheral technique applied to 3D after the fact. They are reshaping how LSS is formulated, trained, evaluated, and deployed. A coherent treatment of this shift is overdue.

1.3 Scope, Contributions, and Differentiation from Prior Surveys

Several recent surveys touch on parts of the territory this paper covers. Foundational Models for 3D Point Clouds is the closest related survey [31], but its scope—classification, detection, segmentation, generation, and reconstruction across indoor and outdoor 3D—forces only a high-level summary per task and de-emphasises automotive LiDAR. A 2025 perception survey spans the entire AD stack from detection to trajectory prediction [32], and a 2025 point-cloud-segmentation survey treats foundation-model approaches as one section among many.

The gap is therefore not the absence of survey activity. It is the absence of a survey that goes deep on the specific intersection of (i) foundation models, (ii) LiDAR data, (iii) semantic segmentation as the task, and (iv) autonomous driving as the application domain.

Scope of this survey. We restrict ourselves to point-cloud semantic segmentation produced by automotive LiDAR. We require that at least one foundation model — pretrained on substantial unlabelled or weakly-labelled data and intended for downstream transfer — participates in the training, supervision, or inference of the segmentation pipeline. We exclude indoor 3D segmentation (S3DIS, ScanNet) except where indoor-trained components are demonstrably reused for outdoor LiDAR. We also exclude pure 3D object detection and pure occupancy prediction, except where the same foundation-model machinery transfers directly.

The paper offers four contributions:

1. A five-paradigm taxonomy of foundation-model approaches to LSS. The five paradigms are: (a) 2D-to-3D feature distillation; (b) Segment-Anything-based approaches, including multimodal SAM extensions that ingest camera and LiDAR streams jointly; (c) open-vocabulary and vision–language methods, encompassing the foundation-model-driven pseudo-labelling pipelines built on top of them; (d) LiDAR-specific self-supervised pretraining; and (e) the emerging class of general 3D foundation models — Point-BERT, ULIP-2, Sonata, Concerto. We also examine how their representations transfer to outdoor LSS.
2. A unified performance comparison on SemanticKITTI and nuScenes-lidarseg. We place foundation-model methods alongside the strongest fully-supervised baselines, including PTv3 and SphereFormer. Where direct full-fine-tuning numbers are unavailable, we report linear probing and few-shot fine-tuning curves, which are the standard evaluation protocol for this family.

3. A practical-deployment perspective. Real-time inference, edge-device constraints, robustness under adverse weather, and synthetic-to-real domain adaptation are treated as first-class concerns rather than as appendices. These dimensions decide whether a method moves from an arXiv preprint into a production AD stack.
4. Coverage through the first quarter of 2026. The survey incorporates work appearing through SAM4D (ICCV 2025), Sonata (CVPR 2025 Highlight), Concerto (NeurIPS 2025), and LOSC (arXiv 2507.07605). To our knowledge, none of these is yet covered by any peer-reviewed survey.
Figure 1 provides a visual summary of the five paradigms, the upstream foundation-model families they draw on, and their representative methods.

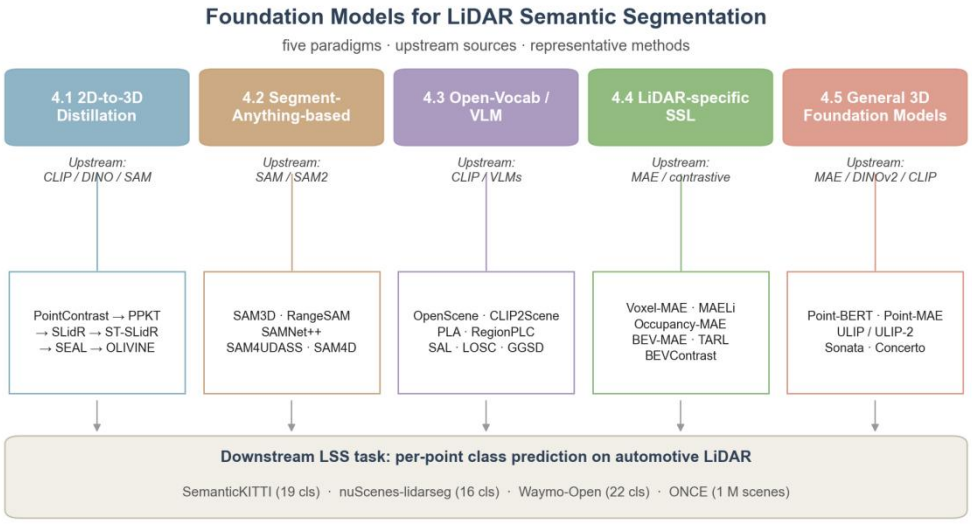

Figure 1 A Five-Paradigm Taxonomy of Foundation-Model-Based LSS Approaches
Note: Each colour corresponds to one paradigm reviewed in §4. Italic text under each header names the upstream 2D foundation-model family typically used; the lower box lists representative methods in chronological order. The bottom band shows the four standard automotive-LiDAR benchmarks on which all paradigms are evaluated.

1.4 Paper Organisation

The remainder of this paper is organised as follows. Section 2 reviews datasets, evaluation protocols, and 2D foundation-model concepts. Section 3 recaps conventional supervised LSS methods, which serve as both historical predecessors and frequent backbones for the foundation-model approaches that follow. Section 4 presents the five-paradigm taxonomy with representative methods. Section 5 consolidates benchmark comparisons. Section 6 covers deployment: latency, memory, and integration with automotive software stacks. Section 7 examines six open challenges. Section 8 concludes.

2 BACKGROUND AND PRELIMINARIES

2.1 Point Cloud Representations

$\{(x_i,y_i,z_i,r_i)\}_{i=1}^{N}$ A single LiDAR sweep is, in raw form, an unordered set of points carrying spatial coordinates and reflectance, where ranges from roughly 100,000 for a 64-beam Velodyne HDL-64E to over 300,000 for a 128-beam sensor. The same physical sweep can be presented to a network in any of four standard forms.
Raw point sets preserve the geometry exactly and impose no quantisation. This is the form consumed directly by PointNet [7], PointNet++ [8], KPConv [9], RandLA-Net [10], and the Point Transformer family [14, 16]. The price is computational: indexing operations such as nearest-neighbour search and farthest-point sampling are inherently irregular and resist straightforward GPU vectorisation, although recent work on serialisation mitigates this [16].
Voxels discretise space into a regular 3D grid. Cylinder3D showed that a cylindrical partition aligned with the rotational sampling pattern of a spinning LiDAR is markedly better than naive Cartesian voxelisation [12]; SPVNAS couples voxels with a parallel point branch [13]; MinkowskiNet generalised voxel processing through 4D sparse convolutions [11]. Voxelisation enables fast, regular tensor operations but loses fine-grained geometry whenever the voxel size exceeds the local point spacing.
Range images project each point onto a 2D plane indexed by the laser's azimuth and elevation, producing a structured grid that 2D CNNs can ingest directly. RangeNet++ [33], SqueezeSeg and its successors, SalsaNext [34], and CENet all exploit this representation[35]. The advantage is throughput; the cost is information loss when multiple points fall onto the same pixel and a less direct correspondence to the physical scene.

Bird's-Eye View (BEV) projects points onto the ground plane. BEV is dominant in detection but less common in raw segmentation, although recent self-supervised work such as BEVContrast and BEV-MAE has revived interest in BEV-pretrained representations[36-37].

A pragmatic reading of recent leaderboards is that no single representation dominates: PTv3 operates on serialised raw points [16], Cylinder3D on cylindrical voxels, FlatFormer on flattened windows over a sparse voxel grid [38]. Hybrid networks combining two or more representations remain competitive. Foundation-model-based methods (Section 4) tend to be representation-agnostic, distilling features into whichever backbone happens to be deployed downstream.

2.2 Problem Formulation

$P=\{p_i=(x_i,y_i,z_i,r_i)\}_{i=1}^{N}$ $C=\{1,2,\ldots,C\}$ $f_\theta: P \rightarrow \{1,\ldots,C\}^N$ Given an input sweep and a fixed label space, a LiDAR semantic segmentation network learns a function that assigns each point an integer class label. Most methods produce point-wise softmax logits and select the argmax at inference; a smaller set of methods—particularly those derived from open-vocabulary 2D models (Section 4.3)—replace the categorical output with a similarity score against text-encoded class names, enabling label sets that are not fixed at training time.

Training is overwhelmingly fully supervised at the point level. A growing minority of methods replace dense supervision with weaker signals: scribble annotations [39], single-click labels [40], 2D bounding-box supervision propagated through SAM [41], or no manual labels whatsoever [29]. Section 4.3 and Section 7.4 treat these cases at length.

2.3 Datasets and Evaluation

Four datasets dominate empirical comparison.

SemanticKITTI augments the KITTI driving sequences with 19-class point-wise annotations across more than 43,000 sweeps recorded by a 64-beam Velodyne in suburban Karlsruhe [2]. It remains the most widely reported benchmark, although its single-sensor, single-city composition limits its generality.

nuScenes-lidarseg adds 16-class lidar-segmentation labels to the 1,000-scene nuScenes dataset, captured in Boston and Singapore with a 32-beam Velodyne HDL-32E [3-4]. Lower beam count and harsher sparsity at range make it a useful contrast to SemanticKITTI.

Waymo Open Dataset released LiDAR semantic segmentation labels in 2022 [5], covering 22 fine-grained classes across 1,150 sequences from US cities. The 2024 Waymo 3D Semantic Segmentation Challenge—won by an extension of PTv3—has become a useful annual indicator of the field's frontier.

ONCE supplies one million unlabelled scenes alongside a smaller annotated split, making it a natural pretraining corpus for self-supervised methods (Section 4.4) [6].

Two specialised datasets recur throughout the discussion of robustness. SemanticSTF provides 21-class point-wise annotations under real rain, fog, and snow, recorded in Saint-Étienne [18]; it is the de facto benchmark for adverse-weather generalisation. SynLiDAR is a synthetic 13-sequence corpus with 32 classes and roughly 20,000 scans, used to study sim-to-real transfer [19].

$\text{mIoU}=\frac{1}{C}\sum_{c=1}^{C}\frac{TP_c}{TP_c+FP_c+FN_c}$ The standard metric across all five datasets is mean Intersection-over-Union (mIoU), evaluated point-wise; per-class IoU is reported alongside. Submissions occasionally also report frequency-weighted mIoU and overall accuracy (the latter is dominated by ground and building points and is rarely informative on its own). Class taxonomies differ across datasets, so raw mIoU values are not directly comparable: 75.5 mIoU on SemanticKITTI's 19 classes does not translate to 75.5 on nuScenes' 16 classes.

For self-supervised and foundation-model-based methods, two additional protocols are standard. Linear probing freezes the pretrained backbone and trains only a linear classifier on the labelled split, isolating the quality of the learned representation. Few-shot fine-tuning—typically at 1%, 5%, 10%, and 25% of the labels—measures how efficiently the pretrained backbone converts limited supervision into segmentation accuracy. Most distillation papers (Section 4.1) report both.

2.4 Foundation Model Concepts

A foundation model, in the sense used here, is a network trained on a large unlabelled or weakly-labelled corpus with the explicit intent that its features will transfer to many downstream tasks with little or no further supervision [20-21]. Five 2D foundation models recur throughout Section 4, and a brief recap is useful.

The Vision Transformer (ViT) divides an image into a regular grid of 16×16 patches, embeds each patch linearly [21], and processes the resulting sequence with a standard transformer encoder. Stripped of convolutional inductive biases, ViT trained on hundreds of millions of images outperforms its CNN predecessors and—more relevantly here—provides the standard backbone that subsequent foundation models build on.

CLIP trains an image encoder and a text encoder jointly on roughly 400 million image–caption pairs with a contrastive objective that pulls each image close to its caption and away from others [20]. The result is a shared embedding space in which images and arbitrary text strings can be compared directly. This property drives every open-vocabulary 3D method in Section 4.3: project a 3D point's neighbourhood into CLIP image space, encode a candidate class name into

CLIP text space, take the cosine similarity, and the result is a class score that does not require the class to have been seen during training.

The Masked Autoencoder (MAE) pretrains a ViT by masking 75% of input patches at random and reconstructing the missing pixels [22]. The encoder sees only the visible patches—reducing compute by roughly 4×—and the decoder is discarded after pretraining. MAE established that reconstruction-based self-supervision could rival or exceed supervised pretraining at scale and inspired the LiDAR-specific masked autoencoders reviewed in Section 4.4: Voxel-MAE [42], MAELi [43], BEV-MAE [37], and Occupancy-MAE all transpose this recipe to outdoor 3D [44].

The Segment Anything Model (SAM) takes the foundation-model paradigm in a different direction [23]: rather than producing transferable features, SAM produces transferable masks. Trained on 11 million images and 1.1 billion auto-generated masks, it accepts a sparse prompt (point, box, or text via a separate encoder) and returns a segmentation mask for the indicated region; SAM2 extends this to video with mask propagation [24]. Section 4.2 reviews strategies for lifting SAM masks from images into 3D LiDAR space, and SAM4D takes this further by training a single foundation model that ingests synchronised camera and LiDAR streams natively [30].

DINOv2 is the strongest purely self-supervised image foundation model available at the time of writing [25]. Trained on 142 million curated images with no text supervision and no labelled data, its features match or exceed CLIP's on dense prediction tasks. DINOv2 is the upstream feature source for several recent 3D distillation works—including SEAL [28], FineGrained-I2L [45], and the cross-modal joint embedding objective in Concerto [46]—because its features encode object-part-level structure more precisely than CLIP, which is optimised for image-level alignment.

Terminology used hereafter: Vision Foundation Model (VFM) covers CLIP, DINOv2, MAE, SAM, and similar transferable visual encoders; Vision–Language Model (VLM) refers more narrowly to image–text aligners (CLIP, ALIGN, SigLIP, BLIP-2); Multimodal Large Language Model (MLLM) refers to LLM-augmented image-to-text systems (LLaVA, GPT-4V), which sit largely outside LSS and reappear only in §7.6.

3 CONVENTIONAL LIDAR SEMANTIC SEGMENTATION: A BRIEF RECAP

3.1 Point-based Methods

PointNet established that an unordered point set can be processed directly by a permutation-invariant network [7]: a shared MLP applied independently to each point, followed by a global max-pool to aggregate features. It made deep learning on point clouds tractable but lost local geometric structure. PointNet++ addressed this with a hierarchical sampling-and-grouping pipeline that aggregates features over progressively larger neighbourhoods, becoming the go-to baseline for the next several years [8].

KPConv generalised the convolution operator itself to point clouds [9]. A KPConv layer attaches a small set of kernel points to fixed positions in Euclidean space; their distance-weighted contributions to a query point form the convolutional response. The deformable variant lets the kernel-point positions adapt to local geometry. KPConv was the first point-based method to seriously challenge voxel-based competitors on outdoor benchmarks and remains a strong backbone choice when memory is tight.

RandLA-Net tackled scalability head-on. Earlier point-based networks could not ingest the 100,000-plus points in a single LiDAR sweep without aggressive subsampling [10]. RandLA-Net's contribution was the observation that random downsampling—theoretically unsound but empirically harmless when paired with a strong local feature aggregator—reduces inference time by two orders of magnitude with negligible accuracy loss. This made point-based methods practical for full-resolution outdoor scenes.

The transformer wave reached point clouds in 2021. Point Transformer V1 adapted self-attention to local point neighbourhoods [47], replacing KPConv's rigid kernel weights with input-dependent attention; Point Transformer V2 introduced grouped vector attention and partition-based pooling [48], narrowing the gap with voxel-based methods on outdoor data. Point Transformer V3 (PTv3) [16], the current strongest point-based design, set aside the question of how to compute attention efficiently and instead serialised points along Hilbert or Z-order space-filling curves for standard windowed attention—reaching 75.5 mIoU on SemanticKITTI validation and winning the 2024 Waymo 3D Semantic Segmentation Challenge in extended form. Stratified Transformer is a sister design that staggers attention windows across layers to enlarge the effective receptive field [14].

3.2 Voxel-based Methods

Voxel networks discretise space and apply 3D convolutions. The challenge is that the resulting tensors are overwhelmingly empty: 95–99% of voxels in an outdoor sweep contain no points. MinkowskiNet solved this with sparse convolutions [11], processing only occupied voxels and using a hash-table data structure to look up neighbours efficiently. It became the default backbone for many downstream pipelines, including the contrastive distillation methods discussed in Section 4.1.

Cylinder3D made a representation-level observation [12]: a rotating LiDAR samples space along the azimuth dimension, so a cylindrical voxel partition aligns voxels with the natural sampling geometry rather than fighting it. The asymmetric kernel design that accompanies the cylindrical grid—elongated along the radial axis where points are sparser—produced a substantial boost, and Cylinder3D held the SemanticKITTI leaderboard top spot for a stretch in 2021.

SPVNAS paired sparse voxels with a parallel point branch [13], reasoning that fine-grained geometry is best captured by points but global context by voxels; a neural-architecture-search loop then tuned the per-layer balance between the two. (AF)²-S3Net introduced an attentive feature-fusion module with adaptive feature selection across scales [49], reaching 70.8 mIoU on SemanticKITTI.

The transformer wave reached voxels too. SphereFormer partitions spherical coordinates into long [15], narrow radial windows that propagate features from dense near-range points to sparse far-range points in a single attention hop—a geometrically-motivated answer to the long-tail-distance problem. OctFormer organises points along an octree as natural transformer tokens [50]; Swin3D ports Swin's windowed-attention pattern to 3D [51]. By late 2024, voxel- and point-based transformers are essentially tied at the top of every benchmark.

3.3 Range-view and Projection-based Methods

The range image—a 2D grid indexed by laser azimuth and elevation—is appealing because it lets standard 2D CNNs consume LiDAR data directly. RangeNet++ added a fast post-processing step using a learned conditional random field to clean up boundary points lost during projection [33]; SalsaNext and CENet continued this line, optimising for real-time inference on automotive hardware [34-35].

Range-view methods are typically fastest at inference and match voxel methods on common classes, but they degrade on small or thin objects (pedestrians, poles, traffic signs) where the projection collapses crucial 3D structure. They have also proven surprisingly fragile under adverse weather (Section 7.1), where rain droplets and snow particles produce spurious points that range-view CNNs latch onto. Recent work has pushed back: FlatFormer adapts windowed attention to flattened pillar tokens and recovers some of the small-object accuracy lost in earlier range-view designs [38].

3.4 Multi-modal Hybrids

Pure-LiDAR methods discard a complementary signal that the vehicle is already carrying: the camera. Multi-modal hybrids bring the camera back into the segmentation pipeline. MSeg3D proposed a unified multi-modal 3D segmentation framework that fuses image features into the voxel stream at multiple network depths [52]; on the Waymo and nuScenes leaderboards it reached 81.1 mIoU at the time of publication. UniSeg released alongside the OpenPCSeg codebase generalised this further by jointly handling semantic [53], instance, and panoptic segmentation in one network and one training run.

The sensor-fusion approach pays a real but bounded cost: synchronisation, calibration, and the loss of the elegant single-input simplicity that made point-based networks attractive in the first place. It also presupposes that the camera is informative—an assumption that can fail at night, in tunnels, or in heavy precipitation. Foundation-model-based methods (Section 4.2 and 4.3) extend the multi-modal idea but, crucially, push most of the heavy lifting into a pretrained image backbone whose cost is amortised across many downstream tasks.

3.5 Why Foundation Models?

The methods sketched above represent enormous progress: SemanticKITTI mIoU climbed from PointNet's 14.6 in 2017 to PTv3's 75.5 in 2024, an absolute gain of 60 points in seven years. Figure 2 visualises this progression and marks the point at which foundation-model methods entered the leaderboard. Yet three structural features of this trajectory motivate the shift the rest of the paper traces.

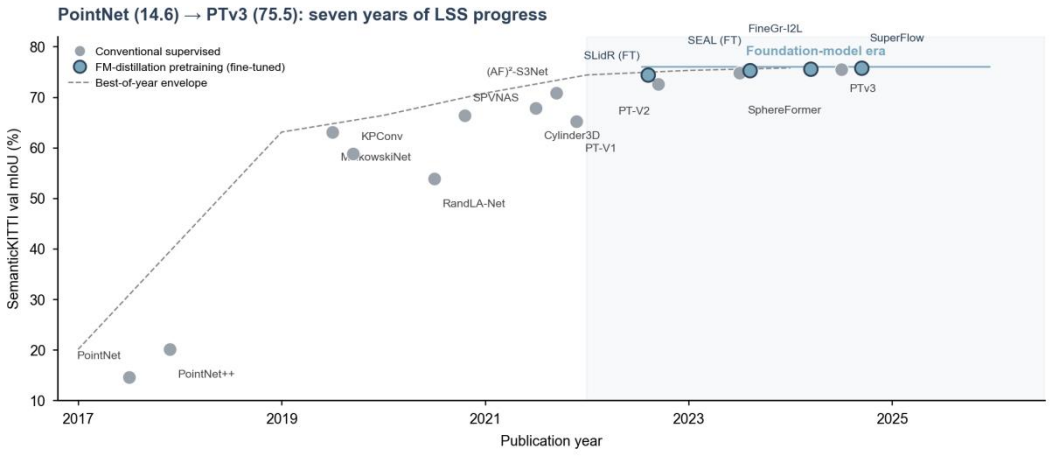

Figure 2 Methodological progress on SemanticKITTI validation, 2017–2026

Note: Grey markers denote conventional fully-supervised methods; blue markers with dark borders denote foundation-model-pretrained backbones evaluated under full fine-tuning. The dashed grey curve traces the best-of-year envelope. The shaded region marks the foundation-model era, in which FM-pretrained backbones consistently sit at or near the leaderboard frontier. Numbers are best reported on the val split in each cited paper.

First, these methods are trained from scratch on each dataset's labelled split and derive no benefit from the unlabelled raw sweeps that operating fleets produce in vast quantity. Second, they assume a closed label set fixed at training time, with no graceful way to extend it post-deployment. Third, they degrade sharply under sensor, geographic, and weather distribution shift, and the cost of acquiring labels for every new condition is the same as the cost of acquiring the original labels. Each of these limitations has been addressed in 2D vision by foundation models—large-scale unsupervised pretraining (MAE, DINOv2), language-conditioned classification (CLIP), and prompt-conditioned segmentation (SAM). The remainder of this survey traces how the 3D LSS community is importing these solutions.

4 FOUNDATION MODEL-BASED APPROACHES: TAXONOMY

4.1 2D-to-3D Knowledge Distillation

4.1.1 Intuition

The starting observation is concrete and asymmetric. Hundreds of millions of labelled and unlabelled images already exist on the web; pretrained 2D backbones such as MoCo, DINO, and CLIP have absorbed this data into rich feature representations. By contrast, labelled outdoor 3D point clouds barely exist at scale, and even the largest unlabelled corpora (one to a few million sweeps) are tiny compared with their 2D counterparts. The asymmetry suggests an obvious move: train a 3D network to mimic a frozen 2D network's features at the points where the two modalities can be paired by sensor calibration.

Concretely, every modern AD platform ships a calibrated camera and LiDAR. For each LiDAR point that falls within a camera's frustum, projection yields a corresponding image pixel, and through it a feature vector from any chosen 2D backbone. A contrastive objective then trains the 3D network to produce features close to the 2D feature at the same physical location and far from features at unrelated locations. No 3D labels are needed at any point in this pipeline.

4.1.2 The PointContrast → SLidR → SEAL lineage

PointContrast was the first work to make this idea concrete in 3D [27], although it operated within the 3D modality rather than across modalities: it required pairs of point clouds of the same scene captured from different viewpoints and contrasted matching versus non-matching points. The followup DepthContrast generalised this to single point clouds by exploiting random spatial transforms as the augmentation source [54]. Neither method exploited the much richer signal available in calibrated RGB.

PPKT (Pixel-Point Knowledge Transfer) made the cross-modal jump [55]. It trained a 3D backbone via an InfoNCE loss between point features and the corresponding pixel features extracted from a frozen ImageNet-pretrained 2D encoder. PPKT was simple, but it suffered two practical defects. First, single-pixel-to-single-point matching is sensitive to LiDAR–camera calibration error: a centimetre-level mismatch can pair a point on a pedestrian with a pixel on the building behind. Second, the loss weights every point equally, regardless of whether it lies on a meaningful object or on an undifferentiated patch of road.

SLidR addressed both defects with a single design choice: contrast at the level of superpixels and superpoints rather than individual pixels and points [26]. A SLIC superpixel pools tens to hundreds of visually coherent pixels into one feature; the corresponding LiDAR points form a superpoint. The contrastive task now pulls together pooled features of paired superpixel–superpoint pairs and pushes apart features of unrelated regions. Each region carries equal weight regardless of its point count, and minor calibration errors are absorbed within the region. The empirical impact was substantial: linear-probe nuScenes mIoU jumped from PointContrast's 21.9 to 38.8.

ST-SLidR noticed that SLidR's contrastive loss treats every non-paired superpixel as a negative [56], even when two superpixels belong to the same semantic class. The "self-conflict" problem is most acute for common classes such as road and vegetation, where many superpixels per scene share a label. ST-SLidR proposed a semantically-tolerant contrastive loss that down-weights negatives whose 2D features are similar to the anchor, plus a class-balancing term to prevent the loss from being dominated by abundant classes. Linear-probe mIoU rose to 40.5.

SEAL replaced the low-level SLIC superpixels with semantically rich superpixels generated by a vision foundation model—specifically SAM and DINO—and added a temporal-consistency regulariser that requires the 3D network's features at the same point across consecutive frames to remain stable [28]. SEAL was the first method to demonstrate that the choice of upstream 2D supervision dominates the choice of contrastive recipe: switching from a vanilla ResNet-50 to a SAM-derived superpixel source pushed linear-probe nuScenes mIoU from SLidR's 38.8 to SEAL's 45.0.

4.1.3 Recent refinements

Two 2024 NeurIPS papers refine the recipe. FineGrained-I2L returns to fine-grained pixel-to-point distillation using VFM features stable enough to make per-pixel matching tractable [45]. OLIVINE argues that purely contrastive distillation is insufficient [57]—contrastive losses encode only what distinguishes regions and not what makes them similar—so it adds an explicit reconstruction objective alongside the contrastive one. SuperFlow (ECCV 2024) integrates temporal flow information into the distillation objective on the grounds that motion is a uniquely 3D-native signal that 2D features cannot supply [58].

4.1.4 What remains unsolved

Three issues recur. First, the 2D-to-3D distillation paradigm structurally assumes camera–LiDAR overlap; points outside the camera frustum (the rear half of a 360° sweep on a forward-camera vehicle) receive no supervision. Second, distillation transfers 2D semantic priors but cannot transfer 3D-specific priors—surface normals, occlusion patterns,

free-space structure—that no 2D model has ever seen; §4.4's masked-autoencoder methods address this gap. Third, the resulting 3D backbone is still a closed-set classifier; §4.3's open-vocabulary methods address that.

4.2 Segment-Anything-based Approaches

4.2.1 Why SAM is a different tool

Where CLIP and DINOv2 produce features that downstream networks must learn to use, the Segment Anything Model produces masks directly. Given a point or box prompt on an image, SAM returns a binary mask covering the indicated region with state-of-the-art quality, and its successor SAM2 extends this to video with mask propagation [24]. For 3D segmentation, this is a qualitatively different primitive: rather than supplying features for a 3D network to learn from, SAM can supply labels—either as supervision during training or as direct outputs at inference. Figure 3 places five representative SAM-based methods side by side; the figure makes clear that SAM masks reach 3D space along structurally distinct routes, not as variations of one recipe.

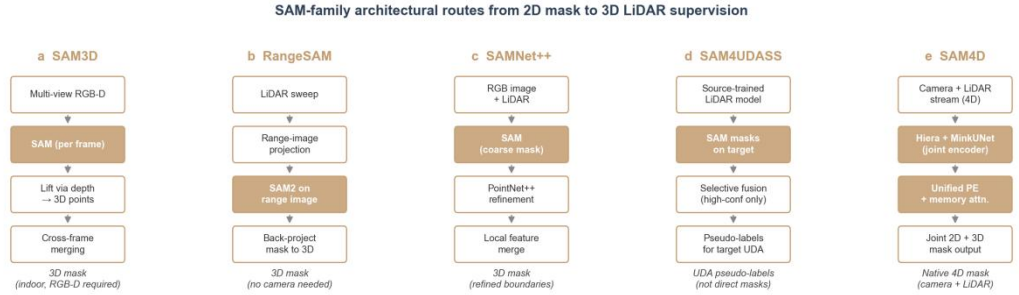

Figure 3 SAM-Family Architectural Routes from 2D Mask to 3D LiDAR Supervision

Note: a, SAM3D applies SAM per RGB-D frame and lifts masks to 3D via the depth channel; suitable for indoor scenes but not for outdoor LiDAR. b, RangeSAM projects the LiDAR sweep into a range image, runs SAM2 on it, and back-projects the resulting masks; no camera is required. c, SAMNet++ uses SAM for a coarse mask and refines it with a downstream PointNet++ branch that captures local 3D geometry. d, SAM4UDASS treats SAM masks as a pseudo-label source for unsupervised domain adaptation, fusing high-confidence regions only. e, SAM4D trains a single multimodal foundation model on synchronised camera and LiDAR streams; masks emerge jointly in both modalities. Coloured blocks mark stages that involve a foundation model directly; white blocks denote conventional 3D processing.

4.2.2 Mask lifting and projection

SAM3D was an early demonstration of the lift-and-project recipe in indoor scenes. Multi-view RGB-D images of a room are passed through SAM frame by frame [59]; the resulting 2D masks are projected onto the 3D point cloud via the depth map; cross-frame inconsistencies are resolved by a graph-based merging procedure. SAM3D worked in indoor settings but did not directly transfer outdoors: outdoor LiDAR has no aligned dense depth from RGB, and the 360° field of view exceeds any single camera's frustum.

RangeSAM sidesteps this by projecting the LiDAR sweep into a 2D range image, applying SAM2 to that image [60], and back-projecting the resulting 2D masks onto the original 3D points. The key observation is that the range image preserves enough 2D structure for SAM's image encoder to produce plausible masks, which the back-projection then converts into 3D supervision. RangeSAM is the first method to apply a 2D vision foundation model to a range-view representation of LiDAR with non-trivial accuracy.

4.2.3 Multimodal SAM foundation models

Lifting 2D SAM masks into 3D is convenient but inherits 2D limitations: anything outside the camera's view receives no supervision, and 2D segmentation errors propagate. The natural next step is a foundation model that ingests both modalities natively.

SAM4D [30], appearing at ICCV 2025, is the most ambitious instantiation. Built on SAM2's hierarchical image encoder (Hiera) and MinkUNet for sparse LiDAR voxelisation, SAM4D introduces a unified multimodal positional encoding that aligns camera and LiDAR features in a shared 3D space, plus a motion-aware cross-modal memory attention module that exploits ego-motion compensation for temporal consistency. To train it, the authors auto-generated the Waymo-4DSeg dataset—15 million image masks, 30 million LiDAR masks, and 300,000 cross-modal masklets—via VFM segmentation, 4D LiDAR reconstruction, and ray casting. SAM4D reports 69.8 mIoU on the image side and 55.7 mIoU on the LiDAR side of Waymo-4DSeg, substantially outperforming single-modality baselines. The deployment cost—running a foundation model on every frame—remains a serious open question for production AD systems and we revisit it in Section 6.

A complementary, less ambitious thread uses SAM as an annotation tool rather than an inference-time component. SAM4UDASS integrates SAM-generated masks into an unsupervised domain adaptation pipeline [61], selectively trusting high-confidence SAM regions to refine pseudo-labels for small and rare classes that the LiDAR-only model misses. The integration is light enough to pair with any existing UDA recipe.

4.2.4 What remains unsolved

The SAM-based paradigm is class-agnostic: SAM produces masks, not labels. Turning a SAM mask into a semantic segmentation requires a separate classification step, typically performed by querying CLIP with the mask region as

image input or by a downstream learnt classifier. The next subsection (§4.3) reviews the methods that close this gap by combining SAM-style masks with CLIP-style language alignment. Beyond this conceptual issue, deployment latency is a hard practical limit: SAM2 ViT-H runs at roughly 15–30 ms per image on an RTX 4090, and SAM4D's full multimodal pipeline is currently far from real-time on automotive hardware.

4.3 Open-Vocabulary and Vision–Language Approaches

4.3.1 Intuition

The methods in §4.1 produce a stronger 3D backbone but the backbone still emits logits over a closed label set. Open-vocabulary methods replace the categorical classifier with a similarity computation in a shared image–text embedding space. At inference, the user supplies class names as text strings; the network encodes each name with the CLIP text encoder; per-point scores are cosine similarities between point features and text features. New classes can be added without retraining, classes can be described compositionally, and the same network can be redeployed across scenes that demand different label vocabularies.

The price is that every method in this category inherits whatever closed-set blind spots its upstream 2D model carries. CLIP, in particular, was trained on web-scraped captions; its representation of fine-grained driving categories—lane markings, drivable area, the difference between a curb and a barrier—is uneven, and several methods have reported that the resulting 3D segmentations require careful prompt engineering to be usable.

4.3.2 CLIP-based feature distillation

OpenScene was the first method to demonstrate dense open-vocabulary 3D segmentation [62]. For each 3D point, its corresponding pixels across all cameras are looked up; their CLIP image features are aggregated into a single per-point CLIP feature. A 3D network is then trained to reproduce these features directly, so at inference the 3D backbone alone can classify any text query without re-projecting through the camera. OpenScene was developed for indoor scenes (ScanNet, Matterport3D) but the recipe transferred cleanly to nuScenes once camera–LiDAR calibration was substituted for the indoor depth sensor.

CLIP2Scene adapted the same idea to outdoor LiDAR with two refinements [63]. First, instead of distilling raw CLIP features, it used CLIP's text encoder to classify each pixel against a candidate driving-class vocabulary, then back-projected the resulting labels onto LiDAR points. This converts the cross-modal supervision from a feature-matching task into a denoised classification task. Second, it introduced a temporal-consistency regulariser that ties together features at the same physical point across consecutive sweeps, addressing the noise that arises when CLIP confidently assigns inconsistent labels across frames.

4.3.3 Point–language contrastive learning

A second thread treats text not as a classification target but as a contrastive signal. PLA (Point–Language Alignment) generates dense captions for each multi-view image of a scene using a captioning model [64], projects the captions onto 3D points via image–LiDAR correspondences, and trains a 3D network with a contrastive loss between point features and the corresponding captioned text. The method assumes that the captioning model knows what is in each region; its accuracy is therefore upper-bounded by 2D captioning quality.

RegionPLC refines this by working at a regional rather than scene level [65]. The image is partitioned into regions (using SAM or similar); each region receives its own caption; only the points falling in a region are aligned with that region's text feature. The locality dramatically reduces the noise in PLA's globally pooled scheme. Lowis3D extends the family to instance-level open-world segmentation [66], addressing the long-standing problem that semantic and instance distinctions are entangled at the point level.

4.3.4 Text-prompted LiDAR segmentation

The methods above produce 3D networks that accept text queries but do not require the user to supply prompts; they output a full segmentation given a fixed vocabulary. A separate strand asks: can a LiDAR network behave like SAM, accepting an arbitrary text prompt at inference and returning the corresponding mask?

SAL (Segment Anything in LiDAR) is the most fully developed answer [29]. It is trained on a pseudo-labelling pipeline: SAM masks on calibrated camera images are lifted into 3D via LiDAR–camera projection, the resulting 3D masks are paired with CLIP tokens computed on the masked image regions, and a 3D network is trained to reproduce these (mask, token) pairs given a text prompt. The result is a zero-shot, text-promptable 3D segmenter that recovers 42% of fully-supervised mIoU on SemanticKITTI and 54% on nuScenes-lidarseg, with no manual 3D annotation. A practical advantage over the lift-and-classify baseline is that SAL produces masks for the full 360° sweep, not only points within the camera frustum.

LOSC (LiDAR Open-vocabulary Segmentation Consolidator) takes a different design choice [67]. Rather than training a text-promptable 3D model, it uses an off-the-shelf 2D VLM as a black box: given a fixed list of target classes specified as text prompts, the VLM produces 2D segmentations of the calibrated camera images; these are consolidated through augmentation- and time-based aggregations, then used to train a fast 3D segmenter specialised to those classes. The 3D segmenter is closed-set at inference, but its specification is fully open-set: switching to a new class list requires only re-running the consolidation pipeline, no relabelling, no additional training data.

GGSD (Geometry-Guided Self-Distillation) and UniM-OV3D continue this line. GGSD adds explicit 3D-geometric regularisation to compensate for cases where 2D VLM segmentations propagate inconsistently to 3D [68-69]; UniM-OV3D pursues fine-grained feature representation, particularly for small objects that low-resolution VLM features miss.

VLM-3D integrates a vision–language backbone end-to-end through low-rank adaptation [70], demonstrating that LoRA-tuned VLMs can serve as 3D detection and segmentation backbones at modest deployment cost.

4.3.5 Foundation-model-driven pseudo-labelling

A natural by-product of the methods above is a cheap source of training labels. FM-WSLSS uses SAM-generated 2D segmentations to bootstrap weakly-supervised 3D LiDAR training [71], achieving 90%+ of fully-supervised mIoU with one click per scan as the only manual annotation. OccNeRF uses 2D VLM segmentations together with depth from neural radiance fields to supervise 3D occupancy in LiDAR-free settings, suggesting that the foundation-model substrate may eventually replace LiDAR for occupancy prediction itself in some deployment scenarios [72].

4.3.6 What remains unsolved

The chief unsolved issue is the gap between open-vocabulary in principle and open-vocabulary in practice. Reported mIoU on novel classes drops sharply for categories that 2D VLMs poorly represent (drivable area, fine-grained traffic infrastructure); prompt engineering still drives several percentage points of variation; and no method is yet competitive with closed-set fully-supervised baselines on a sufficiently large labelled corpus. Open-vocabulary remains a credible long-term direction more than a deployable solution today.

4.4 LiDAR-specific Self-supervised Pretraining

4.4.1 Intuition

The methods of §4.1 lift 2D priors into 3D, but this is structurally limited: 3D properties—surface normals, free-space versus occupied-space, occlusion patterns, the radial sparsity gradient of a rotating LiDAR—are invisible to any 2D foundation model. Self-supervised pretraining directly on outdoor LiDAR sweeps targets this gap. Two paradigms dominate: masked autoencoding (reconstruct what was hidden) and contrastive learning (pull together correlated pairs, push apart unrelated ones).

4.4.2 Masked autoencoders for outdoor LiDAR

Voxel-MAE was the first to adapt the MAE recipe to large-scale outdoor LiDAR [22, 42]. The key insight is that automotive point clouds have highly variable density—a point cloud patch near the sensor may contain hundreds of points per cubic metre, while a patch at 50 m may contain only a handful. Voxel-MAE replaces MAE's fixed-size patches with non-overlapping voxels containing a dynamic number of points, and masks 70% of non-empty voxels. The encoder sees only the visible voxels; the decoder reconstructs the masked ones. After pretraining, the decoder is discarded and the encoder transfers to detection and segmentation.

MAELi adds a crucial occlusion-aware design [43]. A naive masked-reconstruction loss penalises the network for failing to reconstruct points that the LiDAR could not have measured in the first place (occluded regions, free space). MAELi distinguishes empty space from occluded space in its reconstruction loss: the encoder is trained to predict not whether a voxel is occupied, but whether a voxel should be occupied given the line-of-sight geometry. This produces a representation that captures 3D structure rather than the LiDAR scanning pattern itself.

Occupancy-MAE introduces a distance-aware masking strategy [44]: voxels closer to the sensor are masked with higher probability, reflecting their higher reconstruction value. BEV-MAE performs the same masked-reconstruction objective but in BEV space rather than voxel space [37], exploiting the fact that BEV is the dominant representation for downstream detection heads. All four methods report that pretraining on a few hundred thousand unlabelled sweeps lifts downstream segmentation mIoU by 2–5 points, with the gain largest in the few-shot fine-tuning regime.

4.4.3 Contrastive self-supervision

Where masked autoencoders learn from reconstruction, contrastive methods learn from similarity. TARL (Temporal-Aware Representation Learning) defines positives at the level of segments—roughly [73], persistent objects—across consecutive sweeps, and pulls them together while pushing apart segments from unrelated scans. Segment-level granularity sits between point-level (PointContrast) and frame-level (instance discrimination) and proves a productive sweet spot, although the segment-extraction preprocessing is non-trivial.

BEVContrast simplifies this further by performing contrast at the level of BEV cells rather than 3D segments [36]. The author motivation is pragmatic: BEV cells are cheap to compute and align, and the resulting representation transfers as well as the more expensive segment-level alternative. This is the design that the SLidR authors themselves arrived at after several years of work on cross-modal distillation, and it suggests that BEV may be the right intermediate representation for most outdoor SSL recipes.

4.4.4 What remains unsolved

LiDAR-specific SSL methods consistently produce 2–5 point gains but rarely close the gap with the strongest fully-supervised methods or the strongest cross-modal distillation methods. The reasons are partly empirical (corpus sizes are still small relative to 2D pretraining) and partly architectural (no clear winner has emerged among masked, contrastive, and BEV-based recipes). Section 7.6 returns to this question in the context of LiDAR-native foundation models.

4.5 General 3D Foundation Models

4.5.1 The object–scene divide

The methods of §4.1–§4.4 are tailored to outdoor automotive LiDAR. A separate community has been building general-purpose 3D foundation models, mostly trained on indoor scenes and object-centric data. We review them here because two recent works—Sonata and Concerto—suggest that their representations may transfer to outdoor LSS in non-trivial

ways, and because the community will need to decide over the next two years whether outdoor and general 3D foundations remain separate.

4.5.2 Object-level 3D pretraining

Point-BERT adapted the BERT masked-modelling paradigm to point clouds [74]: a tokeniser quantises point patches into discrete tokens, a transformer is pretrained to predict masked tokens, and the resulting backbone transfers to downstream object classification and part segmentation. Point-MAE removed the tokeniser [75], reverting to the simpler MAE recipe of reconstructing masked patches in raw coordinates. Point-M2AE introduced a multi-scale variant with coarse-to-fine reconstruction [76]. All three methods are evaluated on object-centric benchmarks (ModelNet, ScanObjectNN, ShapeNet) and were not designed with outdoor LiDAR in mind.

ULIP and ULIP-2 step beyond pure 3D self-supervision by aligning point clouds with images and text in a shared representation space, mirroring CLIP's recipe but with three modalities [77-78]. The pretrained ULIP-2 backbone, used without fine-tuning, exceeds the strongest fully-trained object-classification baselines on several datasets, demonstrating that multi-modal alignment scales.

4.5.3 Scene-level 3D foundation models

Object-level models do not transfer well to scene-level tasks because scenes contain orders of magnitude more points and substantially different statistical structure. Sonata argued that earlier 3D self-supervised methods [79], when evaluated by linear probing rather than full fine-tuning, fail because of a "geometric shortcut": the model collapses onto easily-extractable spatial features and never learns higher-level structure. Sonata uses Point Transformer V3 as backbone and a self-distillation objective with explicit spatial-information obscuration to force the model past this shortcut. The result is a 3D encoder whose linear-probe performance is 3.3× the previous state of the art and rivals the fine-tuning performance of earlier methods.

Concerto [46], appearing at NeurIPS 2025, takes the next step by training jointly on 2D and 3D self-supervised objectives. The 2D branch is DINOv2; the 3D branch is Sonata's recipe; the cross-modal objective is a JEPA-style joint embedding prediction that aligns point features with corresponding image patch features. Concerto's linear-probe performance on indoor ScanNet semantic segmentation reaches 77.3 mIoU—exceeding the simple concatenation of Sonata and DINOv2 features by 1.4 points and demonstrating that the synergy from joint training exceeds what either modality contributes alone.

4.5.4 Transferring to outdoor LSS

The open question is whether Sonata and Concerto, trained predominantly on indoor data, transfer to outdoor automotive LiDAR. Preliminary results in the Concerto paper suggest the transfer is non-trivial: indoor-trained Concerto features outperform indoor-trained DINOv2-only features on a held-out outdoor evaluation, but they still lag specialist outdoor pretraining methods such as SEAL. The community has yet to commit serious compute to training a Concerto-scale model on outdoor sweeps; doing so is among the most concrete near-term opportunities in this space (Section 7.6).

4.5.5 What remains unsolved

The boundary between "general" and "outdoor-specific" 3D foundations is currently drawn pragmatically rather than principally. Indoor data is plentiful and well-curated; outdoor LiDAR data is plentiful but largely unstructured; the right way to combine them—joint training, sequential pretraining, modality-specific fine-tuning—is open. Whether 2024-vintage scene-level FMs eventually subsume the more specialised distillation and SSL methods of §4.1 and §4.4, or whether two separate ecosystems persist, will likely determine the shape of the field in 2026 and beyond.

5 PERFORMANCE COMPARISON AND DISCUSSION

The methods reviewed in Section 4 cannot be ranked by a single number. Some are pretraining recipes whose value is realised only after fine-tuning; some are zero-shot inference systems whose value is realised when no labels exist; some are deployment-oriented and whose value is realised at low latency rather than high accuracy. This section therefore presents three complementary views: standard benchmark mIoU under full supervision (§5.1), pretraining transfer under linear probing and few-shot fine-tuning (§5.2), and the cost—measured in compute and annotation—of the gains in §5.1 and §5.2 (§5.3). Section 5.4 closes with a discussion of what these numbers do and do not tell us.

5.1 Fully-Supervised Benchmark Comparison

Table 1 collects validation-split mIoU on SemanticKITTI and nuScenes-lidarseg for the strongest methods covered in Sections 3 and 4. Several patterns are immediately visible.

Conventional methods still hold the top rows. PTv3 sits at 75.5 on SemanticKITTI val and 80.4 on nuScenes val[16] , marginally ahead of SphereFormer and Cylinder3D [12, 15]. Among foundation-model-based methods, no published number exceeds the best conventional baseline on a fully-supervised, in-domain comparison. This is unsurprising: full supervision on a large labelled corpus is the regime in which the FM advantage is least relevant. The right comparison for FM-based methods is in the regimes where labels are scarce, the test domain shifts, or the class set extends—and those comparisons appear in §5.2 and §5.3.

The gap between paradigms is narrower than headlines suggest. SLidR [26], when fully fine-tuned on 100% of the labels, reaches 74.8 on nuScenes val, against 80.4 for PTv3 fully supervised: a 5.6-point gap. SEAL narrows this further to 75.6 [28]. Concerto [46], although evaluated only on indoor benchmarks at the time of writing, demonstrates that the

right SSL recipe can match supervised fine-tuning rather than merely approach it. The "FM methods are weaker than supervised" narrative is correct in the strict ordering sense but materially overstated in magnitude.

Hybrid methods—those that combine a foundation-model representation with a strong supervised backbone—often dominate. UniSeg reports 81.0 on nuScenes val by fusing image and LiDAR streams with a learnt backbone [53], exceeding any single-modality method. SAM4D is conceptually similar in motivation, although direct nuScenes comparisons are not yet published [30].

Table 1 Validation-Set mIoU on SemanticKITTI (19 Classes) and nuScenes-Lidarseg (16 Classes)

Method	Type	SemKITTI val mIoU	nuSc-seg val mIoU	Year
PointNet++ [8]	Point-based, conv.	20.1	—	2017
KPConv [9]	Point-based, conv.	58.8	—	2019
Cylinder3D [12]	Voxel-based, conv.	67.8	76.1	2021
SPVNAS [13]	Hybrid, conv.	66.4	—	2020
(AF)²-S3Net [49]	Voxel + attention	70.8	—	2021
SphereFormer [15]	Voxel transformer	73.5	78.4	2023
PTv3 [16]	Point transformer	75.5	80.4	2024
MSeg3D [52]	Multi-modal, conv.	—	81.1	2023
UniSeg [53]	Multi-modal, conv.	—	81.0	2023
SLidR [26] (fine-tuned)	FM, distillation	—	74.8	2022
SEAL [28] (fine-tuned)	FM, VFM-distillation	—	75.6	2023
SAL [29] (zero-shot)	FM, text-promptable	(≈42% of supervised SemKITTI)	(≈54% of supervised nuSc)	2024

Note: Numbers are the best reported in the cited papers; "—" indicates the dataset was not evaluated. Conv. = conventional supervised; FM = foundation-model-based.

5.2 Pretraining Transfer: Linear Probing and Few-Shot Fine-Tuning

Foundation-model methods earn their keep in low-label regimes. The standard evaluation freezes the pretrained backbone, trains only a linear classifier ("linear probing") or a full classifier on a small labelled subset ("few-shot fine-tuning"), and measures mIoU on the validation set.

Table 2 reports nuScenes linear probing and few-shot results for the major distillation-family methods. Figure 4 visualises the same data for visual comparison. Two trends are clear. First, linear-probe mIoU rose from 21.9 (PointContrast, 2020) to 38.8 (SLidR, 2022) to 45.0 (SEAL, 2023)—a doubling of representation quality in three years, achieved without any new labels. Second, the gain compounds as the fraction of available labels grows: at 100% fine-tuning, SEAL beats PointContrast by only 1 mIoU; at 1% fine-tuning, SEAL beats PointContrast by more than 13.

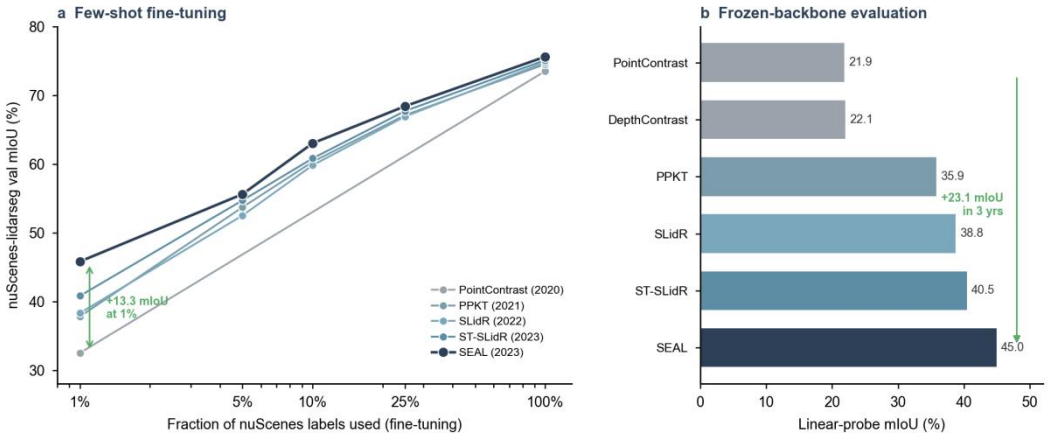

Figure 4 Label-Efficient Pretraining on nuScenes-Lidarseg

Note: a, Few-shot fine-tuning curves: each method's mIoU as a function of the labelled-data fraction used during fine-tuning (log scale). b, Linear-probe mIoU under a frozen backbone. The progression PointContrast → PPKT → SLidR → ST-SLidR → SEAL adds 23.1 mIoU over three years (2020–2023) without any change in labelling budget. Numbers as reported in under a unified evaluation protocol [28].

Table 2 nuScenes-Lidarseg Linear-Probing (LinProbe) and Few-Shot Fine-Tuning mIoU at 1%, 5%, 10%, 25%, and 100% of the Labelled Training Set

Method	LinProbe	1%	5%	10%	25%	100%
PointContrast [27]	21.9	32.5	—	—	—	73.5
DepthContrast [54]	22.1	31.7	—	—	—	73.6
PPKT [55]	35.9	37.8	53.7	60.3	67.1	74.5
SLidR [26]	38.8	38.3	52.5	59.8	66.9	74.8
ST-SLidR [56]	40.5	40.8	54.7	60.8	67.7	75.1
SEAL [28]	45.0	45.8	55.6	63.0	68.4	75.6

Note: Numbers reported by Liu et al. (SEAL paper [28]) under the unified evaluation protocol they introduced.

A complementary picture emerges from text-promptable methods. SAL reports zero-shot LiDAR panoptic segmentation that recovers 42% of the fully-supervised baseline on SemanticKITTI and 54% on nuScenes—without seeing a single 3D label [29]. LOSC reports similar magnitudes on different prompt sets [67]. These numbers should be read carefully: 42% sounds modest in absolute terms but represents most of the segmentation capability that previously required millions of human-annotated points. For applications in which adding a new class is more important than the last few mIoU points on a fixed taxonomy, the trade is often acceptable.

5.3 Annotation and Compute Costs

A complete picture must include the costs that produce the numbers above.

Annotation cost. Producing dense LiDAR segmentation labels has been variously reported as 30 to 90 minutes per frame at full coverage [17]. A 0.3% sparse-labelling protocol still requires 8 to 27 minutes per frame depending on the label policy [17]. For a 20,000-frame training set, full annotation thus consumes the equivalent of 5 to 15 person-years; FM-based methods that achieve 90%+ of fully-supervised mIoU with only one click per frame (FM-WSLSS [71], YoCo [40]) reduce this to fractions of a person-month.

Pretraining compute. SLidR reports a 32-GPU-day pretraining budget on nuScenes [26]; SEAL roughly doubles this [28]; Sonata and Concerto consume 200+ GPU-days at indoor scale and would consume more at outdoor scale [46, 79]. These costs are amortised over many downstream tasks—a single pretrained backbone can serve detection, segmentation, occupancy, and tracking heads—but the upfront commitment is non-trivial for academic labs.

Inference latency. This dimension matters for deployment and is treated in Section 6. A preview: FM-distilled backbones inherit the cost of whatever 3D backbone they distil into (Cylinder3D, MinkUNet, PTv3), with no inference overhead at all; SAM-based pipelines that invoke SAM at inference pay a 15–30 ms image-side cost per frame; SAM4D's full multimodal pipeline is currently far from real-time on automotive-grade compute.

Figure 5 collapses the patterns of §5.1 and §5.2 onto a single plot. The horizontal axis is annotation budget (log scale, from zero-shot at 0.1% to fully supervised at 100%); the vertical axis is nuScenes-lidarseg val mIoU. Conventional methods cluster on the right, near the fully-supervised plateau (~75 mIoU). FM-distillation methods extend leftward along a Pareto frontier inaccessible to conventional methods. Open-vocabulary zero-shot methods occupy the leftmost band. The plot makes the field's central claim visually concrete: progress in 2022–2025 has happened along the label-efficiency axis, not by lifting absolute mIoU.

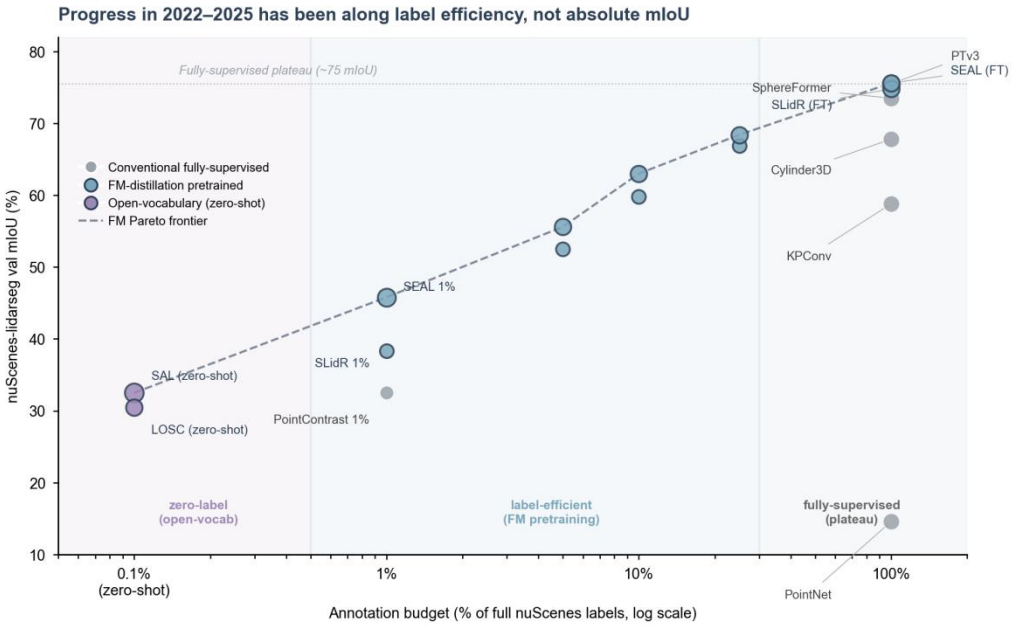

Figure 5 Label-efficiency Pareto Plot on nuScenes-Lidarseg

Noter: Each point represents a method evaluated under a particular annotation budget (1%, 5%, 10%, 25%, or 100% of the labelled training set, or zero-shot text prompting at 0.1%). The dashed line traces the Pareto frontier formed by foundation-model methods. Conventional methods (grey) cluster near the fully-supervised plateau but cannot reach the label-efficient regime; FM-distillation methods (blue) and open-vocabulary methods (purple) occupy the previously empty left half-plane.

5.4 What These Numbers Do and Do Not Tell Us

The numbers above are useful but should not be over-interpreted. Three caveats are particularly important.

Cross-dataset comparison is treacherous. SemanticKITTI's 19 classes and nuScenes' 16 classes are not the same partition of the same underlying world; a method that beats another on one dataset can lose on the other for reasons of

class taxonomy alone. The recent Multi-Space Alignments work reports striking ranking flips between the two datasets for several otherwise-similar methods [80].

Reported numbers are usually best-case. Authors tune hyperparameters on each new method and do not always re-tune the baselines; published improvements of 1–2 mIoU should be discounted accordingly. The community has begun to address this through unified evaluation protocols—the SEAL paper's protocol [28], used for our Table 2—but these are not yet standard.

Robustness numbers are absent from this section. Performance on SemanticSTF (rain/fog/snow) and on cross-dataset transfer (SemanticKITTI → nuScenes) is reported separately in Section 7 because methods that look strong on the standard benchmarks do not always survive distribution shift. The FM literature is, on balance, more robust than the conventional literature on this dimension—but the margin is smaller than its proponents claim and larger than its critics admit.

6 APPLICATIONS AND DEPLOYMENT CONSIDERATIONS

6.1 Latency and Throughput

A production AD perception stack runs at 10 Hz minimum, with budgets typically partitioned as roughly 30 ms for segmentation, 30 ms for detection, 20 ms for fusion and tracking, and the remainder for planning and control. A perception module that exceeds 100 ms total latency cannot keep up with a vehicle moving at urban speed. This places sharp constraints on what FM-based methods can be deployed.

FM-distilled backbones are deployable today. SLidR-pretrained and SEAL-pretrained MinkUNet or Cylinder3D backbones impose no inference overhead beyond the underlying backbone, which already runs at 10–30 Hz on automotive-grade GPUs. The FM compute is paid once at training and recovered at every inference.

SAM-based runtime methods are not yet deployable. Running SAM on every frame at 15–30 ms (RTX 4090; longer on automotive Drive Orin) consumes more than half the perception budget for a single auxiliary signal. SAM2's 4–6× speedup over SAM-H narrows but does not close the gap. SAM4D's full multimodal pipeline is currently several seconds per frame and is research-only at the time of writing.

Open-vocabulary methods sit in between. OpenScene-style methods that distil CLIP features into a 3D backbone pay no inference penalty but accept whatever 2D-derived blind spots CLIP carries. Methods that query the CLIP text encoder per inference for a small fixed set of classes (LOSC [67]) are deployable; methods that re-prompt the VLM on every frame are not.

A concrete deployment-frontier benchmark, Are We Ready for Real-Time LiDAR Semantic Segmentation in Autonomous Driving? [81], evaluated recent methods on automotive-grade hardware and found that only range-view and lightweight voxel architectures—not the top-of-leaderboard transformers—survive a 30 ms latency budget at full input resolution. Foundation-model pretraining can lift the accuracy of these lightweight backbones above what training-from-scratch achieves, and this combination, rather than the headline-grabbing transformer architectures, is what production teams actually deploy.

6.2 Edge and Embedded Hardware

Most production AD systems run on the NVIDIA Drive Orin (254 TOPS INT8) or its successor Drive Thor; some Tier-1 suppliers deploy on FPGA-based platforms (Xilinx Zynq UltraScale+) for sustained-throughput, low-power scenarios. Both platforms favour quantised, sparse, regular computation—exactly the operations that range-view and voxel CNNs perform efficiently and that point-based transformers do not.

Recent work on real-time LiDAR segmentation reports lightweight CNN architectures such as CENet running at >50 Hz on Drive Orin with 0.5–2 mIoU below the top transformer architectures [35, 81]. Pretrained with SLidR or SEAL, these same backbones close most of the gap. The deployable pipeline is not "the strongest method"; it is "the strongest method that fits the hardware budget, pretrained with the strongest available SSL recipe."

6.3 Sensor Fusion in the Wild

The multi-modal hybrids of §3.4 and the multimodal foundation models of §4.2.3 share a structural advantage—using both camera and LiDAR—and a structural risk: they fail when one modality fails. Production systems must degrade gracefully when the camera is blinded by direct sunlight, when the LiDAR is degraded by heavy precipitation, or when calibration drifts after a curb strike.

A pragmatic deployment pattern that has emerged in industry is the redundant cascade: run a strong LiDAR-only segmenter (for example a SEAL-pretrained Cylinder3D) as the primary, a lightweight camera-only segmenter as a backup, and a multi-modal fusion stage (UniSeg [53], MSeg3D [52]) to refine when both modalities are available. Foundation models contribute at every layer: better LiDAR-only pretraining, better camera segmentation, better cross-modal fusion. The cascade is not elegant—a cleaner design would be a single multi-modal foundation model that handles every case—but it is robust to the failure modes production teams actually observe.

6.4 Operational Constraints Beyond Latency

Latency is the most-discussed deployment metric, but production teams report that several others matter as much in practice.

Memory. A foundation-model pretrained network may have hundreds of millions of parameters. Drive Orin's 16 GB shared memory must accommodate the segmenter alongside detection, occupancy, planning, and OS overhead; backbones above ~50 M parameters are increasingly difficult to fit.

Determinism. Safety certification requires reproducible inference. SAM and similar prompt-conditioned models that include random sampling steps in their pipelines must be modified or replaced; deterministic distillation-pretrained backbones do not have this issue.

Continual updates. A deployed model must accept periodic updates—new classes, new geographies—without full retraining of the foundation model. The pretrained-then-fine-tuned pattern of §4.1 supports this naturally; the train-once-deploy-once pattern of SAM4D-style multimodal foundations does not.

These constraints, taken together, explain why the production deployment of FM-based LSS is at present concentrated in distillation-pretrained backbones (§4.1) and open-vocabulary feature distillation (§4.3.2) rather than the headline-grabbing SAM4D and Concerto demonstrations. The latter two are likely the future; the former two are the present.

7 OPEN CHALLENGES AND FUTURE DIRECTIONS

7.1 Robustness under Adverse Weather

The single largest gap between research benchmarks and real-world deployment lives here. Cylinder3D [12], which reaches 67.8 mIoU on SemanticKITTI's clear-weather data, drops by more than ten percentage points when evaluated on SemanticSTF [18], the de facto benchmark for adverse-condition LiDAR. Rain droplets, fog, and snow particles produce spurious returns that shift the spatial distribution of points, attenuate intensity values, and alter object boundaries in ways that training-set augmentation only partially captures.

Three lines of work are converging on this problem. Augmentation-based methods such as Selective Jittering and Learnable Point Drop simulate the geometric perturbations and energy-absorption point drops induced by adverse weather [82], recovering roughly five mIoU points on the SemanticKITTI → SemanticSTF transfer. Universal mixing methods such as UniMix construct a physically-realistic "bridge domain" and mix samples between source and target [83], achieving simultaneous gains on UDA and DG. Class-aware methods such as NTN focus optimisation on safety-critical classes [84], recognising that mis-classifying a pedestrian under fog is qualitatively different from mis-classifying a road surface. Recent range-view-specific work reports +15 to +20 mIoU gains through reflectance-distortion calibration [85].

Survey-level coverage of this dimension is now mature [86]. The unsolved problem is not how to handle weather if you have a foundation-model representation; it is whether the foundation-model paradigm itself confers any inherent robustness advantage. Preliminary evidence suggests it does—FM-distilled backbones degrade more gracefully than randomly-initialised ones—but the magnitude is small and the mechanism is not understood.

7.2 Domain Generalisation and Synthetic-to-Real Transfer

Sensor changes (32-line vs 64-line vs 128-line LiDAR), geographic shifts, and synthetic-to-real gaps all manifest as distribution shift at deployment time. SynLiDAR supplies a 32-class synthetic corpus to study the last case [19]. CoSMix introduced a compositional sample-mix strategy that pulled SynLiDAR → SemanticKITTI transfer into the 30–35 mIoU range [87]; GIPSO extended this to source-free online adaptation [88]; and LiDOG generalised across multiple domains simultaneously rather than one at a time [89].

Foundation-model methods enter this space from two directions. First, FM-pretrained backbones have systematically smaller source–target gaps than from-scratch baselines, because their features encode domain-invariant priors learnt from very large 2D corpora. Second, open-vocabulary methods (§4.3) sidestep parts of the domain-shift problem entirely: text prompts are domain-invariant in a way that label distributions are not. The frontier question is whether a strong FM-pretrained backbone plus domain-specific test-time adaptation can outperform domain-specific full-fine-tuning—and whether the deployment story (no per-domain retraining) is operationally viable.

7.3 Real-Time Inference at Scale

Section 6.1 covered the latency frontier as it stands today. The forward-looking question is whether the strongest research methods—SAM4D, Concerto, and their successors—can be made to run within an automotive perception budget. Three avenues are visible.

The first is distillation into a deployable backbone: train SAM4D-style multimodal foundation models offline, then distil their outputs into a lightweight student that runs in real time. This is the dominant pattern in 2D vision (CLIP → ALIGN → SigLIP → MobileCLIP) and there is no obvious obstacle to its repetition in 3D. The second is quantisation and sparsity: 4-bit and 8-bit quantised inference on automotive chips routinely achieves 4× speedups with <1 mIoU loss; the missing piece is industry-grade tooling for sparse 3D operations under quantisation. The third, more speculative, is hardware co-design: dedicated 3D-sparse accelerators that handle voxelised foundation models natively rather than re-using GPU primitives designed for dense 2D tensors.

A useful empirical anchor for this section is Are We Ready for Real-Time LiDAR Semantic Segmentation [81], which evaluated several recent methods on automotive hardware and found that lightweight range-view and voxel architectures, pretrained with FM-based SSL, are the only combinations that survive a 30 ms budget at full resolution. The conclusion holds for now; whether it holds in 2026, given the rate at which inference engines for transformers are improving, is genuinely open.

7.4 Annotation-Efficient Learning at Scale

Foundation-model methods reduce but do not eliminate the dependence on labels. Several recent works push toward the genuinely label-free regime. YoCo requires only one click per scan and recovers most of fully-supervised mIoU [40]; FM-WSLSS uses SAM-generated 2D segmentations to bootstrap weak supervision (and the scatter-image variant of the same line transfers annotation effort from 3D points to 2D) [71]; MWSIS propagates 2D bounding-box supervision into 3D instance segmentation [41]. The most ambitious entry, ALISE [90], achieves annotation-free LiDAR instance segmentation by combining text-prompted mask generation (à la SAL [29]) with a learnt 3D refinement step.

The frontier is at the intersection of two questions. First, can the no-label regime ultimately match the full-label regime within a few mIoU? Open-vocabulary methods today reach roughly 50–55% of fully-supervised performance; the remaining 45–50% is a real gap that no current method has decisively closed. Second, what is the right interaction model when some labels are available? The pretraining-then-fine-tuning recipe of §4.1 is a partial answer; the active-labelling literature suggests that even one to ten labels per class, chosen well, can cover much of the gap.

7.5 Continual and Class-Incremental Learning

A deployed AD model does not stop learning. New classes are introduced post-deployment; existing classes drift as fleet composition changes; geographies expand. Conventional retraining—gather labels, retrain end-to-end—does not scale to continuous fleet operation. Continual learning addresses this, and a small but growing literature applies it to LSS specifically [91]. The challenge is that the catastrophic-forgetting problem is acute in 3D: the same network must retain proficiency on classes seen in the original training distribution while learning new classes from a small new-data sample.

Foundation-model representations help in a structurally interesting way. Because the FM-pretrained backbone encodes general 3D priors rather than dataset-specific features, fine-tuning a lightweight head on new classes is much less likely to overwrite the priors. The community has not yet quantified this advantage rigorously for LSS specifically, but the analogous result in 2D vision (foundation-model backbones forget less catastrophically than from-scratch backbones) is well established.

7.6 Toward LiDAR-Native Foundation Models

The final and most consequential open question is whether the field will eventually move from "foundation models adapted to LiDAR" to "foundation models trained natively on LiDAR." The recipe is straightforward in principle: a Concerto-scale architecture [46], 100 to 1,000 GPU-days of pretraining compute, and an unlabelled corpus of several million outdoor sweeps drawn from a heterogeneous set of sensors and geographies. Sonata and Concerto demonstrate that the recipe works at scene scale on indoor data [79]; ULIP-2 demonstrates that multi-modal alignment scales for 3D objects. No single instantiation has yet committed the resources required for outdoor scene-level foundations [78].

If and when such a model appears, it will likely subsume several of the paradigms reviewed in this paper. The 2D-to-3D distillation methods of §4.1 are, structurally, a workaround for the absence of a 3D-native foundation; if the foundation exists, distillation becomes a tool for compressing it rather than for replacing absent priors. The SAM-based methods of §4.2 may persist for prompt-based interaction but lose their role as primary pretraining sources. The open-vocabulary methods of §4.3 will likely remain, since text-conditioned recognition is a property the foundation model itself should provide.

The two-year question that the field will resolve in 2026–2027 is whether such a model emerges from academic labs with compute support, from industrial AD perception teams whose unlabelled data dwarfs anything publicly available, or from neither—leaving the patchwork of specialised methods reviewed here as the long-term equilibrium. Our best guess is that the model emerges from industry, is published only partially, and that academic equivalents arrive 12–18 months later. We may be wrong; we will know soon.

8 CONCLUSION

Foundation models have reshaped how the LiDAR semantic segmentation community trains, evaluates, and deploys its networks—not by displacing the conventional supervised pipeline, but by enriching the substrate on which it runs. We have organised the resulting literature into five paradigms: 2D-to-3D distillation; Segment-Anything-based approaches; open-vocabulary and vision–language methods; LiDAR-specific self-supervised pretraining; and the emerging class of general 3D foundation models. The practical impact today is concentrated in distillation-pretrained backbones that fit production latency budgets; the headline-grabbing multimodal foundation models such as SAM4D and Concerto represent the future rather than the present.

Three claims summarise where the field stands. First, foundation models reduce but do not eliminate the dependence on labelled 3D data. The strongest distillation methods recover most of fully-supervised performance with one to two orders of magnitude fewer labels. Zero-label open-vocabulary methods recover roughly half. Second, the choice of upstream 2D supervision dominates the choice of downstream 3D recipe. SEAL's gain over SLidR came mostly from substituting SAM-derived superpixels for SLIC superpixels. The contrastive loss itself was not changed. Third, the open frontier is no longer about how to use 2D foundation models in 3D, but whether to train 3D foundations natively. The answer to this question is due in the next two years. It will determine whether this paper documents an interim recipe or a durable architectural principle.

The cost structure that motivates this work has not gone away: outdoor LiDAR data remains expensive to label at scale, foundation-model training remains expensive in compute, and production deployment remains constrained by latency, memory, and certification requirements that benchmarks systematically under-represent. The gap between what the leaderboards reward and what the deployed fleet requires remains the most informative diagnostic of progress. We expect the next round of foundation-model-based LSS research to be evaluated on that gap, not on incremental mIoU.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] Geiger J, Lenz P, Urtasun R. Are we ready for autonomous driving? The KITTI vision benchmark suite. CVPR, 2012: 3354–3361.
[2] Behley J, Garbade M, Milioto A, et al. SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences. ICCV, 2019.
[3] Caesar H, Bankiti V, Lang A H, et al. nuScenes: A multimodal dataset for autonomous driving. CVPR, 2020: 11618–11628.
[4] Fong W K, Mohan R, Hurtado J V, et al. Panoptic nuScenes: A large-scale benchmark for LiDAR panoptic segmentation and tracking. IEEE Robotics and Automation Letters, 2022, 7(2): 3795–3802.
[5] Sun P, Kretzschmar H, Dotiwalla X, et al. Scalability in perception for autonomous driving: Waymo Open Dataset. CVPR, 2020: 2446–2454.
[6] Mao J, Niu M, Jiang C, et al. One million scenes for autonomous driving: ONCE dataset. NeurIPS, 2021.
[7] Qi C R, Su H, Mo K, et al. PointNet: Deep learning on point sets for 3D classification and segmentation. CVPR, 2017: 652–660.
[8] Qi C R, Yi L, Su H, et al. PointNet++: Deep hierarchical feature learning on point sets in a metric space. NeurIPS, 2017.
[9] Thomas H, Qi C R, Deschaud JE, et al. KPConv: Flexible and deformable convolution for point clouds. ICCV, 2019: 6410–6419.
[10] Hu Q, Yang B, Xie L, et al. RandLA-Net: Efficient semantic segmentation of large-scale point clouds. CVPR, 2020: 11108–11117.
[11] Choy C, Gwak J, Savarese S. 4D spatio-temporal ConvNets: Minkowski convolutional neural networks. CVPR, 2019: 3075–3084.
[12] Zhu X, Zhou H, Wang T, et al. Cylindrical and asymmetrical 3D convolution networks for LiDAR segmentation. CVPR, 2021: 9939–9948.
[13] Tang H, Liu Z, Zhao S, Lin Y, Lin J, Wang H, Han S. Searching efficient 3D architectures with sparse point-voxel convolution. ECCV, 2020.
[14] Lai X, Liu J, Jiang L, et al. Stratified Transformer for 3D point cloud segmentation. CVPR, 2022.
[15] Lai X, Chen Y, Lu F, et al. Spherical Transformer for LiDAR-based 3D recognition. CVPR, 2023.
[16] Wu X, Jiang L, Wang PS, et al. Point Transformer V3: Simpler, faster, stronger. CVPR, 2024: 4840–4851.
[17] Zhang W, Song H, Zhang Z, et al. From sparse semantics to rich instances: Empowering label-efficient LiDAR panoptic segmentation via geometric priors. Neural Networks, 2026, 200: 108767.
[18] Xiao A, Huang J, Xuan W, et al. 3D semantic segmentation in the wild: Learning generalized models for adverse-condition point clouds. CVPR, 2023.
[19] Xiao A, Huang J, Guan D, et al. Transfer Learning from Synthetic to Real LiDAR Point Cloud for Semantic Segmentation. AAAI, 2022.
[20] Radford A, Kim J W, Hallacy C, et al. Learning transferable visual models from natural language supervision (CLIP). ICML, 2021.
[21] Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: Transformers for image recognition at scale. ICLR, 2021.
[22] He K, Chen X, Xie S, et al. Masked autoencoders are scalable vision learners. CVPR, 2022.
[23] Kirillov A, Mintun E, Ravi N, et al. Segment Anything. ICCV, 2023.
[24] Ravi N, Gabeur V, Hu YT, et al. SAM 2: Segment Anything in images and videos. arXiv preprint arXiv:2408.00714, 2024.

[25] Oquab M, Darcet T, Moutakanni T, et al. DINOv2: Learning robust visual features without supervision. Transactions on Machine Learning Research, 2024.
[26] Sautier C, Puy G, Gidaris S, et al. Image-to-Lidar self-supervised distillation for autonomous driving data. CVPR, 2022: 9891–9901.
[27] Xie S, Gu J, Guo D, et al. PointContrast: Unsupervised pre-training for 3D point cloud understanding. ECCV, 2020.
[28] Liu Y, Kong L, Cen J, et al. Segment any point cloud sequences by distilling vision foundation models. NeurIPS, 2023.
[29] Ošep A, Meinhardt T, Ferroni F, et al. Better call SAL: Towards learning to Segment Anything in LiDAR. ECCV, 2024.
[30] Xu J, Wang S, Ni Z, et al. SAM4D: Segment Anything in camera and LiDAR streams. ICCV, 2025.
[31] Thengane V, Zhu X, Bouzerdoum S, et al. Foundational models for 3D point clouds: A survey and outlook. arXiv preprint arXiv:2501.18594, 2025.
[32] Sathyam R, Li Y. Foundation models for autonomous driving perception: A survey through core capabilities. arXiv preprint arXiv:2509.08302, 2025.
[33] Milioto A, Vizzo I, Behley J, et al. RangeNet++: Fast and accurate LiDAR semantic segmentation. IROS, 2019: 4213–4220.
[34] Cortinhal T, Tzelepis G, Aksoy E E. SalsaNext: Fast, uncertainty-aware semantic segmentation of LiDAR point clouds. ISVC, 2020.
[35] Cheng H, Han X, Xiao G. CENet: Toward concise and efficient LiDAR semantic segmentation for autonomous driving. ICME, 2022.
[36] Sautier C, Puy G, Boulch A, et al. BEVContrast: Self-supervision in BEV space for automotive LiDAR. 2026.
[37] Lin Z, Wang Y, Qi S, et al. BEV-MAE: Bird's eye view masked autoencoders for point cloud pre-training in autonomous driving scenarios. AAAI, 2024.
[38] Liu Z, Yang X, Tang H, et al. FlatFormer: Flattened window attention for efficient point cloud transformer. CVPR, 2023.
[39] Unal O, Dai D, Van Gool L. Scribble-supervised LiDAR semantic segmentation. CVPR, 2022.
[40] Liu J, Zhang T, Sun J, et al. You only click once: Single point weakly supervised 3D instance segmentation for autonomous driving. arXiv preprint arXiv:2502.19698, 2025.
[41] Lin L, Yang J, Zhao X, et al. MWSIS: Multimodal weakly supervised instance segmentation with 2D box annotations for autonomous driving. AAAI, 2024.
[42] Hess G, Jaxing J, Svensson E, et al. Masked autoencoder for self-supervised pre-training on LiDAR point clouds. WACV, 2023.
[43] Krispel G, Schinagl D, Fruhwirth-Reisinger C, et al. MAELi: Masked autoencoder for large-scale LiDAR point clouds. WACV, 2024.
[44] Min C, Xu X, Zhao D, et al. Occupancy-MAE: Self-supervised pre-training large-scale LiDAR point clouds with masked occupancy autoencoders. IEEE Transactions on Intelligent Vehicles, 2023.
[45] Wu Y, Zhang M, Cui J, et al. Fine-grained image-to-LiDAR contrastive distillation with visual foundation models. NeurIPS, 2024.
[46] Zhang Y, Wu X, Lao Y, et al. Concerto: Joint 2D-3D self-supervised learning emerges spatial representations. NeurIPS, 2025.
[47] Zhao H, Jiang L, Jia J, et al. Point Transformer. ICCV, 2021.
[48] Wu X, Lao Y, Jiang L, et al. Point Transformer V2: Grouped vector attention and partition-based pooling. NeurIPS, 2022.
[49] Cheng R, Razani R, Taghavi E, et al. (AF)2-S3Net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network. CVPR, 2021.
[50] Wang PS. OctFormer: Octree-based transformers for 3D point clouds. ACM Transactions on Graphics, 2023, 42(4).
[51] Yang Y, Yang YQ, Wang X, et al. Swin3D: A pretrained transformer backbone for 3D indoor scene understanding. NeurIPS, 2023.
[52] Li J, Dai H, Han H, et al. MSeg3D: Multi-modal 3D semantic segmentation for autonomous driving. CVPR, 2023.
[53] Liu Y, Chen R, Li X, et al. UniSeg: A unified multi-modal LiDAR segmentation network and the OpenPCSeg codebase. ICCV, 2023.
[54] Zhang Z, Girdhar R, Joulin A, et al. Self-supervised pretraining of 3D features on any point-cloud. ICCV, 2021.
[55] Liu YC, Huang YK, Chiang HY, et al. Learning from 2D: Contrastive pixel-to-point knowledge transfer for 3D pretraining. arXiv preprint arXiv:2104.04687, 2021.
[56] Mahmoud A, Hu J S K, Kuai T, et al. Self-supervised image-to-point distillation via semantically tolerant contrastive loss. CVPR, 2023.
[57] Zhang Y, Hou J. Is contrastive distillation enough for learning comprehensive 3D representations?. NeurIPS, 2024.
[58] Xu X, Kong L, Shuai H, et al. 4D contrastive superflows are dense 3D representation learners. ECCV, 2024.
[59] Yang Y, Wu X, He T, et al. SAM3D: Segment Anything in 3D scenes. arXiv preprint arXiv:2306.03908, 2023.
[60] Kühn P J, Nguyen D A, Kuijper A, et al. RangeSAM: On the potential of visual foundation models for range-view represented LiDAR segmentation. arXiv preprint arXiv:2509.15886, 2025

[61] Yan W, Qian Y, Wang C, et al. SAM4UDASS: When SAM meets unsupervised domain adaptive semantic segmentation in intelligent vehicles. IEEE Transactions on Intelligent Vehicles, 2024.
[62] Peng S, Genova K, Jiang C, et al. OpenScene: 3D scene understanding with open vocabularies. CVPR, 2023.
[63] Chen R, Liu Y, Kong L, et al. CLIP2Scene: Towards label-efficient 3D scene understanding by CLIP. CVPR, 2023.
[64] Ding R, Yang J, Xue C, et al. PLA: Language-driven open-vocabulary 3D scene understanding. CVPR, 2023.
[65] Yang J, Ding R, Deng W, et al. RegionPLC: Regional point-language contrastive learning for open-world 3D scene understanding. CVPR, 2024.
[66] Ding R, Yang J, Xue C, et al. Lowis3D: Language-driven open-world instance-level 3D scene understanding. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2024.
[67] Samet N, Puy G, Marlet R. LOSC: LiDAR Open-vocabulary Segmentation Consolidator. arXiv preprint arXiv:2507.07605, 2025.
[68] Wang P, Wang Y, Li S, et al. Open vocabulary 3D scene understanding via geometry guided self-distillation. ECCV, 2024.
[69] He Q, Peng J, Jiang Z, et al. UniM-OV3D: Uni-modality open-vocabulary 3D scene understanding with fine-grained feature representation. IJCAI, 2024.
[70] Chang F, Li S, Li Y, et al. VLM-3D: End-to-end vision-language models for open-world 3D perception. arXiv preprint arXiv:2508.09061, 2025.
[71] Chen Y, Xu Z, Huang X, et al Weakly Supervised LiDAR Semantic Segmentation via Scatter Image Annotation. IEEE Transactions on Multimedia, 2025, 27: 4121-4136. DOI: 10.1109/TMM.2025.3535350.
[72] Zhang C, Yan J, Wei Y, et al. OccNeRF: Advancing 3D occupancy prediction in LiDAR-free environments. arXiv preprint arXiv:2312.09243, 2023.
[73] Nunes L, Wiesmann L, Marcuzzi R, et al. Temporal Consistent 3D LiDAR Representation Learning for Semantic Perception in Autonomous Driving. CVPR, 2023: 21674–21683.
[74] Yu X, Tang L, Rao Y, et al. Point-BERT: Pre-training 3D point cloud transformers with masked point modeling. CVPR, 2022: 19313–19322.
[75] Pang Y, Wang W, Tay F E H, et al. Masked autoencoders for point cloud self-supervised learning. ECCV, 2022.
[76] Zhang R, Guo Z, Fang R, et al. Point-M2AE: Multi-scale masked autoencoders for hierarchical point cloud pre-training. NeurIPS, 2022.
[77] Xue L, Gao M, Xing C, et al. ULIP: Learning a unified representation of language, images, and point clouds for 3D understanding. CVPR, 2023.
[78] Xue L, Yu N, Zhang S, et al. ULIP-2: Towards scalable multimodal pre-training for 3D understanding. CVPR, 2024.
[79] Wu X, DeTone D, Frost D, et al. Sonata: Self-supervised learning of reliable point representations. CVPR, 2025.
[80] Liu Y, Kong L, Wu X, et al. Multi-space alignments towards universal LiDAR segmentation. CVPR, 2024.
[81] Haidar S, Chariot A, Darouich M, et al. Are we ready for real-time LiDAR semantic segmentation in autonomous driving?. arXiv preprint arXiv:2410.08365, 2024.
[82] Park J, Kim K, Shim H. Rethinking data augmentation for robust LiDAR semantic segmentation in adverse weather. ECCV, 2024.
[83] Zhao H, Zhang J, Chen Z, et al. UniMix: Towards domain-adaptive and generalizable LiDAR semantic segmentation in adverse weather. CVPR, 2024.
[84] Park J, Lee H, Kang I, et al. No thing, nothing: Highlighting safety-critical classes for robust LiDAR semantic segmentation in adverse weather. arXiv preprint arXiv:2503.15910, 2025.
[85] Yang L, Zhang L, Liu J, et al. Towards generalised range-view LiDAR segmentation in adverse weather. arXiv preprint arXiv:2506.08979, 2025.
[86] Dreißig M, Scheuble D, Piewak F, et al. Survey on LiDAR perception in adverse weather conditions. IEEE Intelligent Vehicles Symposium, 2023.
[87] Saltori C, Galasso F, Fiameni G, et al. CoSMix: Compositional semantic mix for domain adaptation in 3D LiDAR segmentation. ECCV, 2022.
[88] Saltori C, Krivosheev E, Lathuilière S, et al. GIPSO: Geometrically informed propagation for online adaptation in 3D LiDAR segmentation. ECCV, 2022.
[89] Saltori C, Ošep A, Ricci E, et al. Walking your LiDOG: A journey through multiple domains for LiDAR semantic segmentation. ICCV, 2023.
[90] Lyu Y, Jiang G, Liu H, et al. ALISE: Annotation-free LiDAR instance segmentation for autonomous driving. arXiv preprint arXiv:2510.05752, 2025.
[91] Camuffo E, Milani S. Continual learning for LiDAR semantic segmentation: Class-incremental and coarse-to-fine strategies on sparse data. CVPR Workshops, 2023.