Self-Supervised Sparse Sensor Fusion for Long Range Perception

Edoardo Palladin*[1] Samuel Brucker*[1]
Filippo Ghilotti[1] Praveen Narayanan[1] Mario Bijelic[1,2] Felix Heide[1,2]
[1]Torc Robotics [2]Princeton University *Equal contribution
https://light.princeton.edu/LRS4Fusion

Abstract

Outside of urban hubs, autonomous cars and trucks have to master driving on intercity highways. Safe, long-distance highway travel at speeds exceeding 100 km/h demands perception distances of at least 250 m, which is about five times the 50–100m typically addressed in city driving, to allow sufficient planning and braking margins. Increasing the perception ranges also allows to extend autonomy from light two-ton passenger vehicles to large-scale forty-ton trucks, which need a longer planning horizon due to their high inertia. However, most existing perception approaches focus on shorter ranges and rely on Bird's Eye View (BEV) representations, which incur quadratic increases in memory and compute costs as distance grows. To overcome this limitation, we built on top of a sparse representation and introduced an efficient 3D encoding of multi-modal and temporal features, along with a novel self-supervised pre-training scheme that enables large-scale learning from unlabeled camera-LiDAR data. Our approach extends perception distances to 250 meters and achieves an 26.6% improvement in mAP in object detection and a decrease of 30.5% in Chamfer Distance in LiDAR forecasting compared to existing methods, reaching distances up to 250 meters.

1. Introduction

Autonomous vehicles rely on precise perception of their surroundings for scene understanding, prediction, and planning. Today's most successful methods that allow for 360° perception rely on BEV features to perform 3D object detection [39, 69], semantic occupancy prediction [52], tracking [76] and planning [16, 62]. The methods operate on BEV features that encode critical information about the 3D environment, derived from single or multiple sensors that perceive the surroundings.

However, most existing BEV-based methods focus on short-term planning with ranges below 50m [30, 36], making them well-suited for robo-taxi applications in urban environments but insufficient for long-term planning needs, particularly necessary for highway driving and in the context of robo-trucking, where long braking distances demands the perception of objects and planning decisions at long-ranges, to ensure safe, strategic planning beyond 70 m, which existing approaches do not address [16, 22, 62] (Fig. 1).

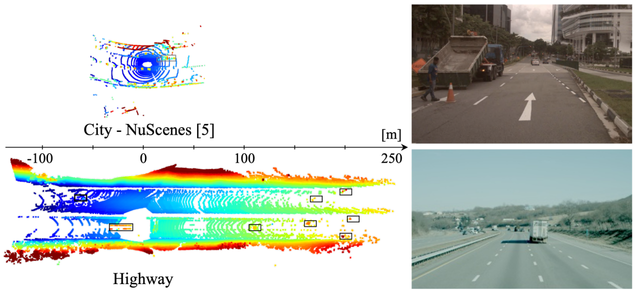

Figure 1. Autonomous vehicles, especially trucks with long braking distances, require long planning horizons for efficient and safe driving in highway scenarios (bottom). This requires extending the typical perception range from 50–100m (top [5]) to beyond 250m where dense representations struggle. We introduce a sparse voxel fusion approach for efficient and accurate 3D scene understanding enabling processing of long-range LiDAR-camera data that leverages spatio-temporal context up to 250m. The proposed model outputs *depth, occupancy, velocity, future LiDAR forecast, accurate object detection*, see Fig. 6 and Fig. 7.

Extending BEV features to cover longer ranges presents significant challenges, as computational complexity and memory footprint grow rapidly. A dense BEV feature map that holds all the information alone grows quadratically in memory with detection ranges. Relying on computations to projecting camera information into a unified representation, such as Lift-Splat-Shoot [42], also increases computational complexity quadratically with range. Therefore, in this work we propose a method that enables BEV features far beyond surround LiDAR and camera data by using a sparse voxel implementation, addressing the limitations of current approaches in long-range perception.

To prevent the training data corpus from growing equally, as objects become increasingly sparse at extended distances, we propose a self-supervision approach. As shown in Fig. 2, the frequency of object instances decreases significantly with distance, so simply increasing the spatial coverage forces a steep increase in required labeled data, which is both costly and time-consuming. Recent approaches have embraced self-supervised pre-training strategies [1, 70], no longer requiring large labeled ground truth datasets. By encoding both current and past sensor data, these methods leverage temporal information to predict future states of the environment. This forecasting is supervised by reconstructing sensor inputs and letting the model learn robust encodings from the natural evolution of the scene over time.

However, such existing self-supervised pre-training approaches are limited to a single modality (e.g., camera-only [70] or LiDAR-only [1]), which restricts their ability to generalize in multi-sensor systems, most commonly deployed in autonomous vehicles. Fusing multiple modalities, such as surround LiDAR and cameras, requires learning to identify complementary information and sparse 3D data from LiDAR with dense, high-resolution imagery from cameras, which we solve with a sparse local attention scheme. The proposed approach enables occupancy prediction, depth prediction, lidar forecasting and object detection.

To train the proposed method for these diverse tasks, we devise a new self-supervised pre-training method that enables long-range multimodal perception without relying on labeled data. Through direct supervision of future LiDAR point clouds and velocities, we demonstrate that the pre-training leads to high performance on tasks such as LiDAR forecasting and 3D object detection for ranges up to $250m$.

We make the following contributions:

- We introduce a long-range LiDAR-camera fusion approach built with a computationally efficient fully sparse voxel representation.
- We devise a self-supervised training approach that incorporates temporal information from past frame history and with self-supervised pretraining providing spatio-temporal context without labels.
- We validate that the method achieves state-of-the-art performance on the LiDAR forecasting task, improving Chamfer Distance by up to 30.5% for ranges up to 250m range and by up to 29% on the NuScenes Dataset for up to 51.2m, on the hardest $3sec$ future horizon prediction.
- We validate the method for Object Detection, improving by *26.6%* (*+11.06 mAP*) on long detection ranges of up to *250m* over existing methods.

2. Related Work

3D Scene Representations are the underpinning of efficient 3D environment perception. Existing work has modeled 3D space as voxels, where each voxel is characterized by an assigned vector [80]. While voxel-based representations excel at capturing detailed 3D structures for tasks such as 3D semantic occupancy prediction, including LiDAR segmentation [7, 57, 59], they lack computational efficiency due to the large number of voxels involved. However, since the vertical dimension typically carries less critical information than the horizontal dimensions, BEV-based approaches streamline the representation by encoding height data within each grid cell [26]. BEV-based approaches are popular in 3D object detection [30, 36, 39], but also semantic occupancy prediction [18], trajectory prediction [12, 50] and planning [16, 20, 21, 62]. As a compromise between voxel and BEV representations, the TPV (Tri-Perspective View) representation has been proposed [17] which relies on three perpendicular cross-planes to represent the 3D scene, initializes query sets on these planes to gather features from images, and exchanges features across views using attention. Other works explore sparse BEVs for 3D detection [34] and sparse voxels [29], drawing on efficient implementations from sparse pointcloud processing for tasks like occupancy prediction [48] and other downstream applications [56]. GaussianFormer [19] propose a latent 3D gaussian representation that is splatted to voxels for 3D Occupancy prediction.

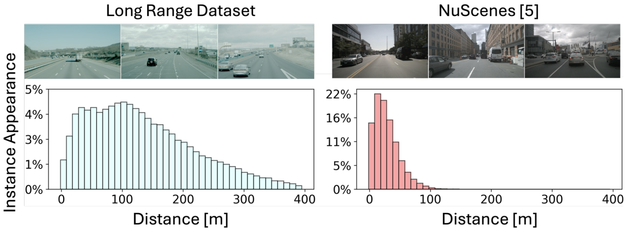

Figure 2. Experimental Long-Range Dataset. To assess the proposed method, we capture and annotate a long-range dataset, showing the instance distribution in the training split alongside NuScenes [5]. While only NuScenes is shown, other popular datasets like Argoverse2 [63] and ONCE [37] follow similar urban distributions, with a concentrated peak of instances in the very close range, unlike our long-range setting with boxes up to 400m.

Multi-Modal Perception aim to enrich LiDAR feature maps by integrating semantic information from camera images [53, 55, 72]. These methods were foundational in combining data from different sensor types. Subsequent research has explored cross-modal feature-level fusion, further refining this integration by directly merging features from both modalities [66, 75]. To address the challenge of accurately projecting RGB camera features into the LiDAR space [28] leverage deformable attention mechanisms [82] to create a unified 3D voxel representation, blending both modalities within a shared spatial framework. More recent approaches operating within the BEV space have enabled fusion of features from multiple sensors in a common reference frame (typically the LiDAR BEV perspective). The aggregated features are then processed by task-specific decoders for applications such as 3D object detection [6, 30, 31, 35, 71], lane estimation [27, 35, 41], object tracking [16], semantic segmentation [30, 35, 36], and planning [16]. This multi-task and multi-modal setup benefits from additional supervision and regularization, improving overall performance across various perception tasks. However, current BEV-based methods still face limitations in projecting detailed camera features into BEV coordinates, primarily due to their reliance on monocular depth estimation techniques [25] or Lift-Splat-Shoot (LSS) methods [36], which estimate depth for camera features and may introduce inaccuracies.

Depth Estimation is a core capability of camera-only geometric perception. Approaches [30, 36] utilize variants of Lift-Splat-Shoot (LSS) [42] to lift 2D camera features into a 3D space. Alternatively, some methods [39] focus on using predicted dense maps and subsequently projecting image features into the 3D LiDAR frame based on the estimated depth. Such depth maps could be predicted by applying widely used monocular depth estimation [2, 43, 68, 74], though with limited accuracy, especially at long ranges due to the inherent scale ambiguity. Stereo depth estimation [4, 33, 54, 65], in contrast, has shown to be more accurate, particularly over long distances, but requires overlapping camera setups for effectiveness. In configurations that include both camera and LiDAR sensors, depth completion methods [40, 47, 77, 83] can be employed to achieve the highest-precision depth predictions. These methods project sparse LiDAR points into image space and utilize image features to interpolate and complete the sparse depth information effectively. We build on this idea to predict dense depth for accurate image feature projection, but devise a lightweight architecture that extends ranges beyond 250m.

Large Scale Self-Supervised Pretraining methods have employed contrastive approaches [9, 13, 23, 51] and masked signal modeling [10, 14, 58, 64]. However, method that rely on pre-training for autonomous driving, which demands semantic understanding, 3D structure, and temporal modeling, have only recently been explored. VoxelMAE [15] extends Masked AutoEncoders to LiDAR data for object detection. UniPAD [67] builds on this by reconstructing color and depth from masked multi-modal inputs. ALSO [3] uses surface reconstruction from present-time LiDAR rays as a pre-training task. ViDAR [70] explores pre-training with temporal modeling by reconstructing future LiDAR from current and past images. Similarly, UnO [1] learns a 4D spatiotemporal occupancy field for the reconstruction of future LiDAR from past LiDAR. Recently, DistillNeRF [56] has demonstrated the distillation of foundation models such as DINOv2 [38] as pretraining for enhancing semantic understanding of scenes. In contrast to existing work, we learn multi-model perception systems and encode 3D structure, semantic understanding, and temporal modelling, focusing on long ranges.

3. LRS4Fusion

In this section, we introduce the proposed **L**ong-**R**ange **S**elf-**S**upervised **S**parse **S**ensor **Fusion** (LRS4Fusion) approach, which is illustrated in Fig. 3. To support prediction distances of up to 250 meters, we rely on a sparse voxel representation that allows us to exploit multi-modal and temporal cues. We train the method with a novel self-supervised training scheme. We first introduce the multi-modal feature extraction and fusion architecture using sparse voxels in Sec. 3.1, then the self-supervised pre-training scheme in Sec. 3.2.

3.1. Sparse Sensor Fusion

The Camera Encoder extracts features from each multi-view image and projected lidar pointcloud. Therefore, we project the lidar pointcloud P onto every camera frame I_i^{RGB} for each camera i, producing a corresponding sparse depth image I_i^D. This results in a 4-channel image defined as $I_i^{\text{RGBD}} = \left[I_i^{\text{RGB}}, I_i^D\right]$, where $I_i^{\text{RGB}} \in \mathbb{R}^{H\times W\times 3}$ and $I_i^D \in \mathbb{R}^{H\times W\times 1}$, yielding $I_i^{\text{RGBD}} \in \mathbb{R}^{H\times W\times 4}$. Image features F are extracted using f_{img} a multi-scale feature pyramid based on Vim [81] as follows,

$$F_1^i, F_2^i, F_3^i, F_4^i = f_{\text{img}}(I_i^{RGBD}). \tag{1}$$

We additionally concatenate a camera embedding, derived from intrinsic and extrinsic calibration matrices, to each flattened patch sequence—small non-overlapping image regions encoded by the Vim [81] backbone—to explicitly incorporate camera-specific geometry into feature extraction. Encoded features are extracted at 4 different Vim depths and then decoded to generate feature maps at multiple scales. The resulting outputs are fed into an FPN network [32] to generate features that are later lifted to 3D.

The Depth Estimation build on top of the multi-scale features $F = \{F_1^i, F_2^i, F_3^i, F_4^i\}$, which are passed to the depth model f_{depth} to predict dense depth D_i for each frame i. The model employs a multi-scale recurrent architecture that iteratively refines depth predictions by integrating sparse LiDAR depth and image features with increasing resolution. For each scale, a small backbone $\mathcal{B}$ extracts context h_t and confidence C_{inp} features which guide the refinement process, that is

$$C_{\text{inp}}, h_t = \mathcal{B}(F) \tag{2}$$

We employ a convolutional update block that uses a Minimal Gated Unit (MGU) [78] to improve the depth map. The MGU, designed with a single forget gate for simplicity, updates depth gradients by adjusting the hidden state based on current depth estimates, whereas more widely applied GRU-based networks [33, 49, 83] typically rely on separate update and reset gates to control state adjustments. This consolidation of gates reduces computational load and parameter count by one third, enhancing efficiency while maintaining effective control for accurate depth predictions. Each iteration refines the depth map by first estimating a depth gradient $\nabla d_t = F_g(h_t)$ from context features, in a depth gradient network F_g, then merging that gradient with the previous depth estimate in a depth integration module. This module balances sparse LiDAR depth with image-

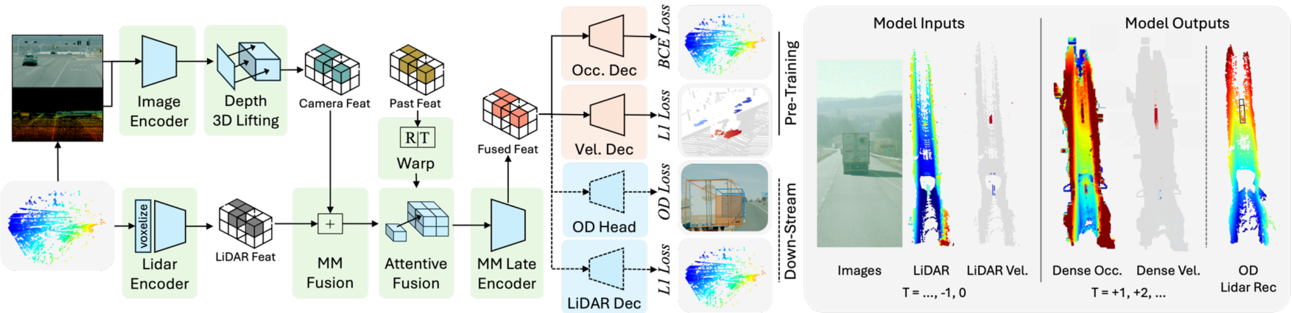

Figure 3. LRS4Fusion: Sparse Multi-Modal Self-Supervision and Fusion. Camera features are lifted to 3D through accurate depth-maps and joined with LiDAR features in a unified sparse representation. The resulting features are fused through a custom sparse attention with past features before being further processed by the Multi-Modal Late Encoder. During the pre-training stage, final features are passed to the custom sparse occupancy decoder and velocity decoder. In contrast to existing SOTA methods that focus on pre-training mono-modality backbones, our approach aims at pre-training multimodal encoders. The method produces depth, occupancy, velocity, and LiDAR at future frames, along with object detection (OD) predictions.

derived gradients, producing a refined map that incorporates both sensor measurements and visual context. Formally, the updated depth is given by:

$$d_{t+1} = d_t - \Delta d, \tag{3}$$

where the correction term Δd is computed as

$$\Delta d = f_{\text{update}} \left(\nabla d_t - g,\, (d_t - s_d) \odot M,\, C_{dg},\, C_{\text{inp}} \right). \tag{4}$$

Here, ∇d_t is the gradient of the current depth estimate d_t, g is the predicted depth gradient, s_d is the sparse depth measurement, M is the valid sparse mask, C_{dg} is the confidence in the depth gradient, and f_{update} represents the convolutional operations within the integration module.

By integrating depth and gradient discrepancies with confidence information across multiple scales, the network incrementally refines the depth map over successive passes, initially ensuring global consistency and later incorporating high-frequency details while smoothing out inaccuracies. The confidence information is predicted by $C_{dg} = f_{\text{conf}}(h_t)$, where f_{conf} is a learned confidence head applied to the hidden state h_t in the MGU update block.

Lifting 2D Features into 3D is performed as first step by projecting each image frame using the camera matrix K and predicted depth D_i to 3D points $\mathbf{X}_C$ from pixel coordinates (u, v), as $\mathbf{X}_C = D_i(u, v)K^-1(u, v, 1)$. The predicted coordinates are casted into sparse voxels and features are aggregated per voxel cell, that results in a sparse representation in the form of $F_C^i = [\mathbf{F}_C, \mathbf{X}_C]^i$ for $i = 1, ..., N$ and $\mathbf{F}_C \in \mathbb{R}^{N,F}, \mathbf{X}_C \in \mathbb{N}^{N,3}$, where N is the number of hidden features and F the hidden feature dimension.

The LiDAR Encoder processes the LiDAR scan P, which is first voxelized and encoded using f_{lid}. We implement f_{lid} as voxel-wise PointNet [44]. Features are then extracted through sparse convolutions, followed by a sparse U-Net [45]. Similar to the camera branch, this process yields a sparse representation $F_L = [\mathbf{F}_L, \mathbf{X}_L]$, where $\mathbf{F}_L \in \mathbb{R}^{M,F}$ and $\mathbf{X}_L \in \mathbb{N}^{M,3}$, with M being the number of occupied voxels and F the feature dimension.

Camera-LiDAR Fusion combines the sparse voxels from both modalities in a unified sparse voxel space by concatenating the features from both modalities in the Sparse Fusion Module f_{MM}. Features are first fed into batch norm layers to normalize values from different encoding branches. During the concatenation, for voxels that are empty in either of the two modalities, zeros are appended. After the concatenation the features are passed into a sparse convolution module as a first fusion step. The resulting fused features form a single sparse representation $F_{LC} = [\mathbf{F}_{LC}, \mathbf{X}_{LC}]$ and $\mathbf{F}_{LC} \in \mathbb{R}^{Q,F}, \mathbf{X}_{LC} \in \mathbb{N}^{Q,3}$ where $Q = M + N - O$ is the number of occupied voxels, O is the number of overlapping Camera and LiDAR features and F the hidden feature dimension.

Late Sparse Encoding is applied as subsequent step, where the resulting features F_{LC} are processed through a sequence of Completion Blocks and Contextual Aggregation Blocks f_{con}, following [48]. A pyramid of features is then assembled, where the final multi-scale features are fused within the sparse representation. Compared to previous work [48], the *proposed representation remains sparse along all scales*, further reducing the memory footprint and enabling the use of finer-grained discretization. The resulting latent embedding consists of sparse voxels at 4 different scales $V = [V_1, V_2, V_3, V_4]$. Multiple scales captures coarse and fine-grained information separately, preserving details while maintaining global context. This enables specialized processing before strategic fusion, while larger voxels help fill gaps and propagate information across sparse areas for better long-range perception.

Temporal Sparse Fusion is applied within the Late Sparse Encoding to the second smallest V_2 Voxel representation only. Therefore we utilize the current t_0 and last t_{-1} timestamp. For abbreviation we drop the ^{2} exponent and write V^{t_0} and $V^{t_{-1}}$. To align the voxel maps we need both the rigid body transformation $R \mid T^{t_{-1} \rightarrow t_1}$ between the timestamps and the velocity per voxel to correct the vehicle movements. The velocity are observed directly from the FMCW LiDAR measurements and we accumulate the velocity per voxel $v_q^{t_{-1}}$. The $R \mid T^{t_{-1} \rightarrow t_1}$ are obtained from the vehicle odometry. Past features $V^{t_{-1}} = [\mathbf{V}^{t_{-1}}, \mathbf{X}^{\mathbf{t_{-1}}}]$ are warped to $V^{t'_0} = [\mathbf{V}^{t_{-1}}, \mathbf{X}^{t'_0}]$, where the new positions are computed as

$$\mathbf{X}_q^{t'_0} = (\mathbf{X}_q^{t_{-1}} + \mathbf{v}_q^{t_{-1}} dt)\mathbf{R}|\mathbf{T}^{t_{-1} \rightarrow t_0}. \tag{5}$$

Here, $q = 1, \ldots, Q$ represents the number of past occupied voxels cells.

To maintain sparsity transformed features can not be concatenated with the current features as it we would sequentially add more and more occupied voxels to the latent representation over time, completely neglecting the memory efficiency of the sparse representation.

Therefore, we introduce a novel sparse windowed attention layer, see Fig. 4. Inspired by local attention mechanism, each occupied voxel at the current timestamp t_0 attends to a 3D window of voxels in the previous warped timestamp t_{-1}. The operation directly operates in the sparse representation, without the need of converting the features to a dense BEV grid or dense voxel volume. This approach significantly reduces computational cost while enabling the windowed attention to capture information beyond local neighborhoods, effectively capturing moving actors and correcting misalignments between past voxels transformed into the current reference. Hence, sparse history enhanced features are calculated as

$$V_* = \sum_{V^{t'_0} \in J_s} \text{softmax}\left(\frac{V^{t_0}\,(V^{t'_0})^T}{\sqrt{d}}\right) V^{t'_0}, \tag{6}$$

where Js is the set of neighboring voxels inside the attention sampling window and d the softmax normalization factor representing the dimensionality of the hidden features. Note, basing the mechanism on top of the occupied voxels V^{t_0} the queries don't include unoccupied cells ensuring that the amount of occupied voxels does not explode.

3.2. Self-Supervision

Due to the quadratically shrinking resolution for long ranges and the diminished number of objects at very long distances (Fig. 2), training requires an extensive amount of data. To tackle this challenge, we self-supervise the embedding by a loss composed of a reconstruction loss for sparse occupancy and sparse velocity.

Sparse Occupancy and Velocity Decoder are auxiliary prediction heads predicting a dense geometric and dynamic representation from the voxel embeddings and are shown Fig. 5. Both can be trained at scale from self supervision alone from arbitrary driving recordings without ground truth labels. Both take as input a four-dimensional query point Q (x, y, z, t), where x, y, z are spatial coordinates and t is time, and its outputs are both density and velocity for each queried point. If the query point lies in the past or future, the current voxel representation may not accurately reflect the scene. Rather than transforming the entire voxel space, we only transform the queried point [1], building on top of an existing rigid transformation between the current encoding and both past and future timestamp. This allows us to account for the movements of the ego-vehicle and dynamic actors. A lightweight neural network f_{pose} predicts the new position of the query point, whether in the past or future, by using the query (x, y, z, t) and tri-linearly interpolated voxels at the (x, y, z) location. We interpolate the N nearest neighbors at the query point intersection with the voxel grid, creating a new voxel that captures the precise sub-voxel location. The new query position is calculated as $(x', y', z') = (x, y, z) + f_{\text{pose}}(V^*(x, y, z), Q)$. At the new position, voxel features are again interpolated with N nearest neighbors at all latent scales. The two sets of interpolated voxel features, from both the current and new positions, are stacked and used to predict occupancy $\hat{o}$ and velocity $\hat{v}$ through two lightweight heads f_{occ} and f_{vel}, as

$$\begin{aligned} \hat{o} &= f_{\text{occ}}(V^*(x, y, z), V^*(x', y', z'), Q), \\ \hat{v} &= f_{\text{vel}}(V^*(x, y, z), V^*(x', y', z'), Q). \end{aligned} \tag{7}$$

GT values for self-supervision can be estimated from the recorded LiDAR scan. Occupied labels and non-zero velocities can be directly inferred by the presence and measured velocity of a LiDAR point, while all positions along one LiDAR ray are taken as unoccupied labels as the LiDAR ray travels through free space. More details on occupancy and velocity decoder are provided in the supplemental material.

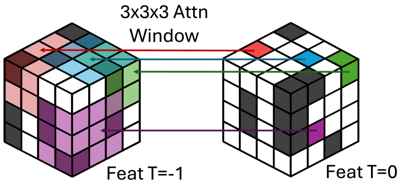

Figure 4. Sparse Window Attention. Occupied voxels in the current frame $T = 0$ attends to a window of occupied voxels in the previous timestamp $T = -1$. The attention is computed between each query in the current frame and all occupied key,values inside the attention window centered at the same location but at the previous timestamp. The method ensure that the final number of occupied voxels does not explode when the number of occupied previous voxels is high. In the example, we consider an attention window of 3x3x3, Red Query attends to 3 voxels in the past frame, Blue to 3, Green to 2 and Purple to 4.

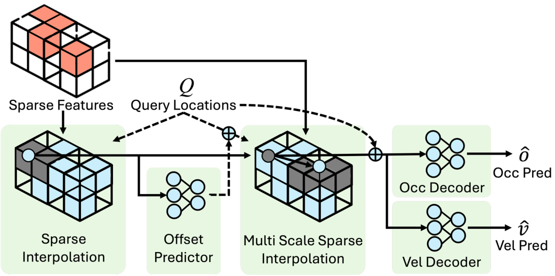

Figure 5. Occupancy and Velocity Decoder. The proposed decoder takes a 4D query point, interpolates nearby occupied voxels to refine the interpolation position based on temporal information and decodes the features from the first and second sampling to generate occupancy and velocity predictions.

3.3. Training

Our training strategy is divided into three stages. In the first stage, we train the image feature encoder and depth prediction modules using a combination of image reconstruction, depth supervision, and feature distillation losses. The second stage involves training the complete model with supervision for past, current, and future frames, covering occupancy and velocity reconstruction. Lastly, we train the object detection on top. Additional training details are provided in the Supplemental Material.

4. Dataset

In order to overcome the range limitations of existing LiDAR-based datasets [5, 8, 11, 46] with sensors restricted to 80 meters, we capture a new multi-modal dataset specifically tailored to heavy-duty trucking scenarios, shown in Fig. 2. We equip a semi-truck with 5 synchronized OnSemi AR0820 cameras, each featuring a $1/2$-inch CMOS sensors that captures raw RCCB data at a resolution of 3848 × 2168 pixels, and arrange them to record a near 360° view at 5 Hz. The system is complemented by Aeva Aeries 4D LiDARs, capturing 3D point clouds up to $400m$ as well as radial velocity measurements at $10Hz$, synchronized with the camera feeds. The dataset includes recordings from diverse locations in Texas, New Mexico and Virginia, spanning highway and urban environments. The captures feature a variety of natural lighting conditions and include $60,000$ unlabeled frames and $36,000$ manually annotated frames for object detection. From these, we curate $25,000$ at $5Hz$ for our training split, capturing seven object classes with the following distribution: "Passenger-Car" ($194,807$ instances, 38.21%), "Vehicle" ($105,116$, 20.62%), "SemiTruck-Trailer" ($70,495$, 13.83%), "Road-Obstruction" ($95,862$, 18.80%), "SemiTruck-Cab" ($40,982$, 8.04%), "Person" ($2,448$, 0.48%), and "Bike" (157, 0.03%). Further information is in the Supplemental Material.

Table 1. The proposed depth model brings boost in speed and reduction in memory footprint while achieving the same accuracy to SOTA methods. Evaluation on accumulated LiDAR point cloud from 0 to 250m, more details to the depth evaluation are provided in the supplemental material.

Method	MAE ↓	RMSE ↓	Runtime [ms] ↓	MEM [GB] ↓
Completion Former [77]	4.98	12.36	188	2.1
OGNI-DC [83]	4.76	13.16	364	2.4
LRS4Fusion	**3.46**	**9.21**	**64**	**1.3**

Table 2. 3D Object Detection performances on Long Range Dataset. ‡ is with ViDAR [70] pre-train.

Method	Modality	mAP ↑	NDS ↑
PointPillars [26]	L	39.31	41.52
BEVFormer [30]	C	23.67	37.99
BEVFormer [30] (w/ Pre-train‡)	C	24.51	38.93
BEVFusion [36]	L + C	40.10	48.43
SAMFusion [39]	L + C	41.55	52.44
LRS4Fusion (w/o Pre-train)	L + C	49.58	**59.12**
LRS4Fusion	L + C	**52.61**	58.06

5. Experiments

In this section, we validate the proposed method. Specifically, we assess the quality of the depth predictions in Sec. 5.1, object detection in Sec. 5.2, and LiDAR forecasting tasks in Sec. 5.3. Additionally, we report ablation studies validating the design choices of the proposed method. Further details on the experimental setup can be found in the Supplementary Material.

5.1. Depth Evaluation

We first evaluate depth prediction of the proposed method and recent existing methods [77] and [83] compared to accumulated LiDAR ground truth. The experimental setup is detailed in the Supplemental Material. The role of the depth network is to reduce inference time and reduce memory footprint without compromising accuracy. Tab. 1 reports how the proposed architecture achieves the lowest MAE and MSE among the three methods. The proposed architecture improves MAE by 27% and MSE by 25%, it achieves the lowest inference time of 0.064s and 1.3GB of memory.

5.2. Object Detection

We compare the proposed method on the OD task against recent existing single-modality and multi-modality methods on the common Mean Average Precision (mAP) and NuScenes Detection (ND) Score metrics. We project sparse features at the smallest scale into a BEV grid and pass them to a CenterPoint [73] head. The camera-only method BEVFormer [30] ($23.67mAP$) fails to achieve the performance of LiDAR models. Pre-training the feature extractor on ViDAR [70] increases the detection accuracy by 3.55% ($24.51mAP$), reinforcing the importance of the pre-training step in the long-range scenarios. The LiDAR-

Table 3. LiDAR Forecasting on Long Range Dataset. We consider a ROI of $[+100m, -100m]$ on the Y axis of the Ego Vehicle and $[+250m, -100m]$ on the X axis.

History Horizon	Method	Modality	1s		3s	
			CD ↓	L1 (m)↓	CD ↓	L1 (m)↓
1s	4DOcc [24]	L	18.933	4.685	-	-
	ViDAR [70]	C	58.228	20.209	51.720	20.688
	LRS4Fusion	L + C	**15.821**	**3.313**	**39.031**	**3.825**
3s	4DOcc [24]	L	23.580	2.996	47.81	4.293
	ViDAR [70]	C	57.284	20.141	56.200	20.531
	LRS4Fusion	L + C	**16.382**	**2.493**	**42.932**	**4.051**

Table 4. LiDAR Forecasting on NuScenes. ROI of $51.2m$ on all sides around the Ego Vehicle. † denotes results as reported in [70].

History Horizon	Method	Modality	Chamfer Distance ↓						
			0s	0.5s	1s	1.5s	2.0s	2.5s	3s
0s	HERMES [79]	C	0.59	-	0.78	-	**0.95**	-	**1.17**
	LRS4Fusion	L + C	**0.087**	**0.348**	**0.566**	**0.748**	0.963	1.205	1.510
1s	4DOcc† [24]	L	-	1.26	1.88	-	-	-	-
	ViDAR [70]	C	-	1.11	1.25	1.40	1.57	1.76	1.97
	LRS4Fusion	L + C	**0.06**	**0.31**	**0.48**	**0.64**	**0.79**	**0.99**	**1.25**
3s	4DOcc† [24]	L	-	0.91	1.13	1.30	1.53	1.72	2.11
	ViDAR [70]	C	-	1.01	1.12	1.25	1.38	1.54	1.73
	LRS4Fusion	L + C	**0.11**	**0.33**	**0.47**	**0.61**	**0.77**	**0.97**	**1.23**

Table 5. LiDAR Forecasting on NuScenes. [1] evaluation setting.

Method	SPFNet [60]	S2Net [61]	RayTracing [24]	4D-OCC [24]	UnO [1]	**LRS4Fusion**
NFCD ↓	2.50	2.06	1.66	1.40	0.89	**0.72**
CD ↓	4.14	3.47	3.59	4.31	1.80	**0.88**

only method PointPillars achieves $39.31mAP$. The fusion method BEVFusion [36] performs only marginally better (+2.01%) than the LiDAR-only approach, $40.10mAP$, confirming the drawbacks of the LSS approach. SAMFusion [39] methods, based on depth-based 3d lifting of camera features, improves by 5.70% over the LiDAR only baseline. The proposed method (52.61) achieves state-of-the-art quality in the long-range dataset, improving the mAP by 26.6% over the second-best method, SAMFusion, on the long-range dataset. Finally, we report that the proposed occupancy-velocity self-supervision improves performance by +6.11% over the proposed model without self-supervision (49.58). Fig. 6 reports qualitative detections at long ranges for vehicles and road debris.

5.3. LiDAR Forecasting

We analyze the performance of our second training step by evaluating the LiDAR reconstruction accuracy at $1s$ and $3s$ into the future. Following [24, 70], we report Chamfer distance (CD) between predicted and ground truth point cloud at $1s$, $3s$ into the future and with $0s$, $1s$, and $3s$ preceding historical horizon. We compare two LiDAR forecasting methods [24, 79] and the camera method [70].

Table 6. Depth ablations. All methods use two iterative refinement steps. “DI” = depth-integration.

Method	Backbone	DINOv2 dist.	Update Block	DI Module	Cam Token	MAE↓	RMSE↓	Time↓	MEM↓
OGNI-DC	CF [77]	false	GRU	OGNI-DC	No	4.76	13.16	364ms	2.4GB
Ours	Vim [81]	No	GRU	OGNI-DC	No	3.83	11.14	245ms	1.3GB
Ours	Vim [81]	Yes	GRU	OGNI-DC	No	3.61	10.32	245ms	1.3GB
Ours	Vim [81]	Yes	MGU	OGNI-DC	No	3.58	10.41	213ms	1.3GB
Ours	Vim [81]	Yes	MGU	Ours	No	3.51	9.31	63ms	1.3GB
Ours	ResNet50	Yes	MGU	Ours	No	3.53	9.28	**58ms**	2.9GB
Ours	Vim [81]	Yes	MGU	Ours	Yes	**3.46**	**9.21**	63ms	**1.3GB**

Table 7. Ablation Experiments. (a) Backbone Ablation, (b) History Horizon Ablation, (c) Self-Supervision Decoders.

(a) Backbone Ablation

Backbone	**Pre-train**	**mAP ↑**
ResNet50	stage 2	50.22
VisionMamba	stage 2	**52.61**

(b) Self-Supervision Decoders

Occupancy Decoder	**Velocity Decoder**	**CD - 1s ↓**	**CD - 3s ↓**
Yes	No	16.592	42.152
Yes	Yes	**15.821**	**39.031**

(c) History Horizon Ablation

History H	**Pre-train**	**mAP ↑**
0s	stage 1	50.75
1s	stage 1	49.58
	stage 2	52.61
	Improvement	**+6.11%**
3s	stage 1	51.16
	stage 2	51.86
	Improvement	**+1.37%**

NuScenes Evaluation We evaluates the LiDAR forecasting task on the NuScenes dataset [5] following the more comprehensive protocol of [70] in Tab. 4 and following [1], condensing the performance into a single metric, in 5. The proposed method achieves state-of-the-art results, with a Chamfer distance of 0.48 (+61.6% over the second best model) on the $1s$ in, $1s$ out tasks, 1.25 (+36.5%) on the $1s$ in $3s$ out tasks, 0.47 (+58.0%) on the $3s$ in $1s$ out tasks and 1.23 (+28.9%) on the $3s$ in $3s$ out tasks. We also compare the performance of the proposed method on the no-history ($0s$ history horizon) task with the recent Hermes [79] work based on large Vision Language Model (VLM): we achieve state-of-the-art results - 0.566 on $1s$ forecasting - with future horizon less then $2s$, where the complex reasoning capabilities of VLMs are able to better forecast, but also compute-heavy.

Long-range Evaluation Tab. 3 evaluates the method on the proposed long-range dataset. 4D-Occ struggle to extract temporal information from the history horizon due to the large motion between frame in the highway scenarios; this is visible by the drop in performance between $1s$, CD 16.87, and $3s$, CD 23.58, history horizon. ViDAR [70] is less affected by large scene movements due to its use of expensive deformable attention and multi-frame reasoning but still struggles with accurate long-range 3D geometry, as monocular surround cameras lack the necessary depth cues and geometric constraints for estimating distant structures. This is reflected by the large CD, 56.2 for $3s$ input $3s$ output, 58.23 for $1sec$ in $1sec$ out, in the full range setting. Instead, the proposed method is able to exploit the multi-modal input and to effectively extract temporal clues from the history horizon, achieving CD of 15.821 on $1sec$ in $1sec$ out and 42.932 on $3sec$ in $3sec$ out.

5.4. Ablation Experiments

We conduct a series of ablation studies to evaluate the design choices of the proposed architecture.

Depth Model. Table 6 reports ablation experiments validating the proposed architectural components in terms of accuracy gains, inference time reduction, and memory footprint. Starting from [83], replacing their backbone with our

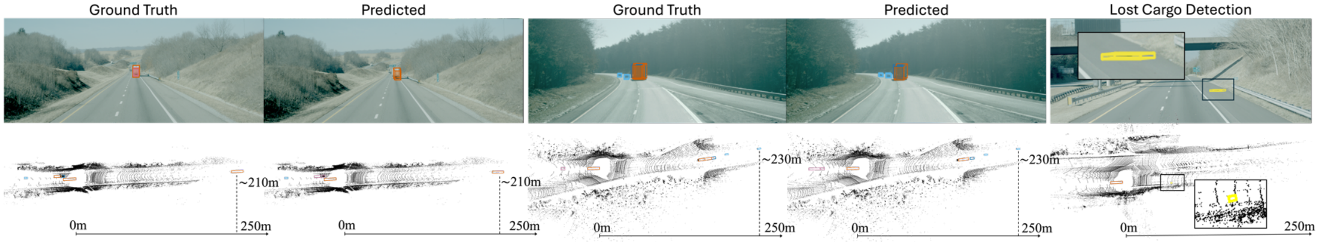

Figure 6. Object Detection. The method detects vehicles and small lost cargo objects at long distances beyond 100m. 3D bounding boxes of Car, Truck-Cab, Truck-Trailer, Road Obstruction with cyan, blue, orange, yellow color, respectively. Please zoom in for details.

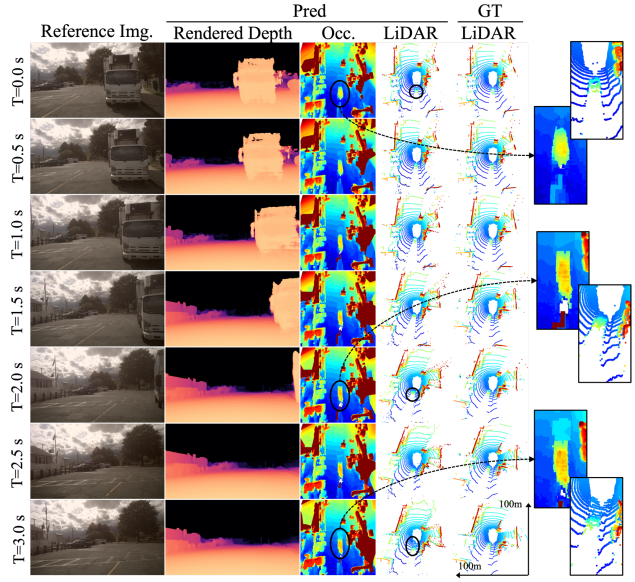

Figure 7. Future predictions up to 3 seconds of depth, occupancy, and LiDAR. The method captures fine details in the occupancy decoder, such as fine structures on the truck, forecasts the motion of other agents (evident in the trailing car), and expresses uncertainty through a gradual spread of occupancy. Please zoom in for details.

proposed Vim [81] encoding leads to a notable reduction of memory of 45% while boosting accuracy in MAE by 24%. Further replacing the GRU update block with MGU improves inference time by 41% over [83], while integrating our proposed Depth Integration Module further reduces it to a total improvement of 82%. Finally, adding Camera Tokens in the image encoding leads to a final improvement of 27% in MAE over the baseline. Using the for BEV tasks commonly deployed ResNet50 instead of Vim decreases inference time marginally but doubles the memory footprint. As scaling to long ranges requires a lightweight backbone, we adopt the Vim-based model.

Camera Backbone. Tab. 7a reports mAP performance on the OD task with Vim [81] backbone and with ResNet50 backbone. Our model with Vim improves mAP by 4.75% over the proposed model with ResNet image backbone.

History Horizon. In Tab. 7c we ablate the input history horizon. Departing from existing work on urban scenarios [30], we empirically find that using a horizon of 1 second on the long-range dataset leads to better Object Detection performance $(+1.44\%)$ than 3 seconds. This can be explained by the higher speeds involved in highway scenarios: higher velocities imply larger motions between frames during which instances can fall out of ROI and, in general, make the feature alignment harder. A car at highway speeds travels in 3 seconds almost the entire region of interest of the NuScenes dataset [5] ($\sim 100m$).

Velocity Decoder. In Tab. 7b, we ablate the velocity decoder during pre-training. Training stage 2 with a 1-s input shows that velocity supervision improves LiDAR forecasting by 4.6%, improving the encoding of moving objects, leading to more accurate predictions of dynamic actors.

6. Conclusion

We introduce a long-range camera-LiDAR BEV method that learns a sparse voxel representation for efficient and spatio-temporal 3D scene understanding. To tackle the need for a large amount of training data, we devise a self-supervised pretraining approach that integrates temporal cues from past frames, enabling the model to predict future occupancy through supervision from raw sensor data. While existing BEV perception methods have been limited to $100m$, the method achieves long-range object detection up to $250m$, improving mAP by *26.6%* (*+11.06 mAP*) at distances up to $250m$. Our method also sets a new state-of-the-art in LiDAR forecasting, reducing Chamfer Distance by up to *30.5%* across the full $[-100m : +250m]$ range and outperforming existing work on the hardest 3-second future horizon in NuScenes by *29%*. We find that tackling highway scenarios for trucking is fundamentally different than urban perception, thus opening the avenue for a new line of work, specializing in achieving real-time long-range perception high-resolution sensors.

7. Acknowledgments

Felix Heide was supported by an NSF CAREER Award (2047359), a Packard Foundation Fellowship, a Sloan Research Fellowship, a Sony Young Faculty Award, a Project X Innovation Award, a Amazon Science Research Award, and a Bosch Research Award.

References

[1] Ben Agro, Quinlan Sykora, Sergio Casas, Thomas Gilles, and Raquel Urtasun. Uno: Unsupervised occupancy fields for perception and forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14487–14496, 2024. 1, 2, 3, 5, 7

[2] Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. *arXiv preprint arXiv:2410.02073*, 2024. 3

[3] Alexandre Boulch, Corentin Sautier, Björn Michele, Gilles Puy, and Renaud Marlet. Also: Automotive lidar self-supervision by occupancy estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13455–13465, 2023. 3

[4] Samuel Brucker, Stefanie Walz, Mario Bijelic, and Felix Heide. Cross-spectral gated-rgb stereo depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21654–21665, 2024. 3

[5] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving, 2020. 1, 2, 6, 7, 8

[6] Hongxiang Cai, Zeyuan Zhang, Zhenyu Zhou, Ziyin Li, Wenbo Ding, and Jiuhua Zhao. Bevfusion4d: Learning lidar-camera fusion under bird's-eye-view via cross-modality guidance and temporal aggregation. *arXiv preprint arXiv:2303.17099*, 2023. 2

[7] Anh-Quan Cao and Raoul De Charette. Monoscene: Monocular 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3991–4001, 2022. 2

[8] Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, and James Hays. Argoverse: 3d tracking and forecasting with rich maps. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8740–8749, 2019. 6

[9] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 3

[10] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3

[11] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3354–3361. IEEE, 2012. 6

[12] Junru Gu, Chenxu Hu, Tianyuan Zhang, Xuanyao Chen, Yilun Wang, Yue Wang, and Hang Zhao. Vip3d: End-to-end visual trajectory prediction via 3d agent queries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5496–5506, 2023. 2

[13] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 3

[14] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 3

[15] Georg Hess, Johan Jaxing, Elias Svensson, David Hagerman, Christoffer Petersson, and Lennart Svensson. Masked autoencoder for self-supervised pre-training on lidar point clouds. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 350–359, 2023. 3

[16] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17853–17862, 2023. 1, 2

[17] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9223–9232, 2023. 2

[18] Yuanhui Huang, Wenzhao Zheng, Borui Zhang, Jie Zhou, and Jiwen Lu. Selfocc: Self-supervised vision-based 3d occupancy prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19946–19956, 2024. 2

[19] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Gaussianformer: Scene as gaussians for vision-based 3d semantic occupancy prediction. *arXiv preprint arXiv:2405.17429*, 2024. 2

[20] Xiaosong Jia, Yulu Gao, Li Chen, Junchi Yan, Patrick Langechuan Liu, and Hongyang Li. Driveadapter: Breaking the coupling barrier of perception and planning in end-to-end autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7953–7963, 2023. 2

[21] Xiaosong Jia, Penghao Wu, Li Chen, Jiangwei Xie, Conghui He, Junchi Yan, and Hongyang Li. Think twice before driving: Towards scalable decoders for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21983–21994, 2023. 2

[22] Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8350, 2023. 1

[23] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020. 3

[24] Tarasha Khurana, Peiyun Hu, David Held, and Deva Ramanan. Point cloud forecasting as a proxy for 4d occupancy forecasting, 2023. 7

[25] Jason Ku, Ali Harakeh, and Steven L Waslander. In defense of classical image processing: Fast depth completion on the cpu. In *2018 15th Conference on Computer and Robot Vision (CRV)*, pages 16–22. IEEE, 2018. 3

[26] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12697–12705, 2019. 2, 6

[27] Qi Li, Yue Wang, Yilun Wang, and Hang Zhao. Hdmapnet: An online HD map construction and evaluation framework. *CoRR*, abs/2107.06307, 2021. 2

[28] Yanwei Li, Yilun Chen, Xiaojuan Qi, Zeming Li, Jian Sun, and Jiaya Jia. Unifying voxel-based representation with transformer for 3d object detection. *Advances in Neural Information Processing Systems*, 35:18442–18455, 2022. 2

[29] Yingyan Li, Lue Fan, Yang Liu, Zehao Huang, Yuntao Chen, Naiyan Wang, and Zhaoxiang Zhang. Fully sparse fusion for 3d object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 2

[30] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European conference on computer vision*, pages 1–18. Springer, 2022. 1, 2, 3, 6, 8

[31] Tingting Liang, Hongwei Xie, Kaicheng Yu, Zhongyu Xia, Zhiwei Lin, Yongtao Wang, Tao Tang, Bing Wang, and Zhi Tang. Bevfusion: A simple and robust lidar-camera fusion framework. *Advances in Neural Information Processing Systems*, 35:10421–10434, 2022. 2

[32] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944, 2017. 3

[33] Lahav Lipson, Zachary Teed, and Jia Deng. Raft-stereo: Multilevel recurrent field transforms for stereo matching. In *2021 International Conference on 3D Vision (3DV)*, pages 218–227. IEEE, 2021. 3

[34] Haisong Liu, Yao Teng, Tao Lu, Haiguang Wang, and Limin Wang. Sparsebev: High-performance sparse 3d object detection from multi-camera videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18580–18590, 2023. 2

[35] Yingfei Liu, Junjie Yan, Fan Jia, Shuailin Li, Aqi Gao, Tiancai Wang, and Xiangyu Zhang. Petrv2: A unified framework for 3d perception from multi-camera images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3262–3272, 2023. 2

[36] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2774–2781. IEEE, 2023. 1, 2, 3, 6, 7

[37] Jiageng Mao, Minzhe Niu, Chenhan Jiang, Hanxue Liang, Jingheng Chen, Xiaodan Liang, Yamin Li, Chaoqiang Ye, Wei Zhang, Zhenguo Li, Jie Yu, Hang Xu, and Chunjing Xu. One million scenes for autonomous driving: Once dataset, 2021. 2

[38] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 3

[39] Edoardo Palladin, Roland Dietze, Praveen Narayanan, Mario Bijelic, and Felix Heide. Samfusion: Sensor-adaptive multimodal fusion for 3d object detection in adverse weather. In *Proceedings of the IEEE European Conference on Computer Vision (ECCV)*, 2024. 1, 2, 3, 6, 7

[40] Jinsun Park, Kyungdon Joo, Zhe Hu, Chi-Kuei Liu, and In So Kweon. Non-local spatial propagation network for depth completion. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pages 120–136. Springer, 2020. 3

[41] Lang Peng, Zhirong Chen, Zhangjie Fu, Pengpeng Liang, and Erkang Cheng. Bevsegformer: Bird's eye view semantic segmentation from arbitrary camera rigs, 2022. 2

[42] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, pages 194–210. Springer, 2020. 1, 3

[43] Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segu, Siyuan Li, Luc Van Gool, and Fisher Yu. Unidepth: Universal monocular metric depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10106–10116, 2024. 3

[44] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 4

[45] Shaoshuai Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network, 2020. 4

[46] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Sheng Zhao, Shuyang Cheng, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset, 2020. 6

[47] Jie Tang, Fei-Peng Tian, Wei Feng, Jian Li, and Ping Tan. Learning guided convolutional network for depth completion. *IEEE Transactions on Image Processing*, 30:1116–1129, 2020. 3

[48] Pin Tang, Zhongdao Wang, Guoqing Wang, Jilai Zheng, Xiangxuan Ren, Bailan Feng, and Chao Ma. Sparseocc: Rethinking sparse latent representation for vision-based semantic occupancy prediction. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, pages 15035–15044, 2024. 2, 4

[49] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 402–419. Springer, 2020. 3

[50] Izzeddin Teeti, Salman Khan, Ajmal Shahbaz, Andrew Bradley, Fabio Cuzzolin, and Lud De Raedt. Vision-based intention and trajectory prediction in autonomous vehicles: A survey. In *IJCAI*, pages 5630–5637, 2022. 2

[51] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 776–794. Springer, 2020. 3

[52] Wenwen Tong, Chonghao Sima, Tai Wang, Li Chen, Silei Wu, Hanming Deng, Yi Gu, Lewei Lu, Ping Luo, Dahua Lin, et al. Scene as occupancy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8406–8415, 2023. 1

[53] Sourabh Vora, Alex H Lang, Bassam Helou, and Oscar Beijbom. Pointpainting: Sequential fusion for 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4604–4612, 2020. 2

[54] Stefanie Walz, Mario Bijelic, Andrea Ramazzina, Amanpreet Walia, Fahim Mannan, and Felix Heide. Gated stereo: Joint depth estimation from gated and wide-baseline active stereo cues. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13252–13262, 2023. 3

[55] Chunwei Wang, Chao Ma, Ming Zhu, and Xiaokang Yang. Pointaugmenting: Cross-modal augmentation for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11794–11803, 2021. 2

[56] Letian Wang, Seung Wook Kim, Jiawei Yang, Cunjun Yu, Boris Ivanovic, Steven L. Waslander, Yue Wang, Sanja Fidler, Marco Pavone, and Peter Karkus. Distillnerf: Perceiving 3d scenes from single-glance images by distilling neural fields and foundation model features, 2024. 2, 3

[57] Xiaofeng Wang, Zheng Zhu, Wenbo Xu, Yunpeng Zhang, Yi Wei, Xu Chi, Yun Ye, Dalong Du, Jiwen Lu, and Xingang Wang. Openoccupancy: A large scale benchmark for surrounding semantic occupancy perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17850–17859, 2023. 2

[58] Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14668–14678, 2022. 3

[59] Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21729–21740, 2023. 2

[60] Xinshuo Weng, Jianren Wang, Sergey Levine, Kris Kitani, and Nicholas Rhinehart. Inverting the pose forecasting pipeline with spf2: Sequential pointcloud forecasting for sequential pose forecasting, 2020. 7

[61] Xinshuo Weng, Junyu Nan, Kuan-Hui Lee, Rowan McAllister, Adrien Gaidon, Nicholas Rhinehart, and Kris M. Kitani. S2net: Stochastic sequential pointcloud forecasting. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVII*, page 549–564, Berlin, Heidelberg, 2022. Springer-Verlag. 7

[62] Xinshuo Weng, Boris Ivanovic, Yan Wang, Yue Wang, and Marco Pavone. Para-drive: Parallelized architecture for real-time autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15449–15458, 2024. 1, 2

[63] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting, 2023. 2

[64] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9653–9663, 2022. 3

[65] Gangwei Xu, Xianqi Wang, Xiaohuan Ding, and Xin Yang. Iterative geometry encoding volume for stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21919–21928, 2023. 3

[66] Shaoqing Xu, Dingfu Zhou, Jin Fang, Junbo Yin, Zhou Bin, and Liangjun Zhang. Fusionpainting: Multimodal fusion with adaptive attention for 3d object detection. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 3047–3054. IEEE, 2021. 2

[67] Honghui Yang, Sha Zhang, Di Huang, Xiaoyang Wu, Haoyi Zhu, Tong He, Shixiang Tang, Hengshuang Zhao, Qibo Qiu, Binbin Lin, et al. Unipad: A universal pre-training paradigm for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15238–15250, 2024. 3

[68] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10371–10381, 2024. 3

[69] Zeyu Yang, Jiaqi Chen, Zhenwei Miao, Wei Li, Xiatian Zhu, and Li Zhang. Deepinteraction: 3d object detection via modality interaction. *Advances in Neural Information Processing Systems*, 35:1992–2005, 2022. 1

[70] Zetong Yang, Li Chen, Yanan Sun, and Hongyang Li. Visual point cloud forecasting enables scalable autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14673–14684, 2024. 1, 2, 3, 6, 7

[71] Junbo Yin, Jianbing Shen, Runnan Chen, Wei Li, Ruigang Yang, Pascal Frossard, and Wenguan Wang. Is-fusion:

Instance-scene collaborative fusion for multimodal 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14905–14915, 2024. 2

[72] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Multimodal virtual point 3d detection. *Advances in Neural Information Processing Systems*, 34:16494–16507, 2021. 2

[73] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Center-based 3d object detection and tracking, 2021. 6

[74] Wei Yin, Chi Zhang, Hao Chen, Zhipeng Cai, Gang Yu, Kaixuan Wang, Xiaozhi Chen, and Chunhua Shen. Metric3d: Towards zero-shot metric 3d prediction from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9043–9053, 2023. 3

[75] Jin Hyeok Yoo, Yecheol Kim, Jisong Kim, and Jun Won Choi. 3d-cvf: Generating joint camera and lidar features using cross-view spatial feature fusion for 3d object detection. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*, pages 720–736. Springer, 2020. 2

[76] Tianyuan Zhang, Xuanyao Chen, Yue Wang, Yilun Wang, and Hang Zhao. Mutr3d: A multi-camera tracking framework via 3d-to-2d queries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4537–4546, 2022. 1

[77] Youmin Zhang, Xianda Guo, Matteo Poggi, Zheng Zhu, Guan Huang, and Stefano Mattoccia. Completionformer: Depth completion with convolutions and vision transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18527–18536, 2023. 3, 6, 7

[78] Guo-Bing Zhou, Jianxin Wu, Chen-Lin Zhang, and Zhi-Hua Zhou. Minimal gated unit for recurrent neural networks. *International Journal of Automation and Computing*, 13(3): 226–234, 2016. 3

[79] Xin Zhou, Dingkang Liang, Sifan Tu, Xiwu Chen, Yikang Ding, Dingyuan Zhang, Feiyang Tan, Hengshuang Zhao, and Xiang Bai. Hermes: A unified self-driving world model for simultaneous 3d scene understanding and generation, 2025. 7

[80] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4490–4499, 2018. 2

[81] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv preprint arXiv:2401.09417*, 2024. 3, 7, 8

[82] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 2

[83] Yiming Zuo and Jia Deng. Ogni-dc: Robust depth completion with optimization-guided neural iterations. *Proceedings of the IEEE European Conference on Computer Vision (ECCV)*, 2024. 3, 6, 7, 8