LowRankOcc: Tensor Decomposition and Low-Rank Recovery for Vision-based 3D Semantic Occupancy Prediction

Linqing Zhao[1,2], Xiuwei Xu[1], Ziwei Wang[1], Yunpeng Zhang[3], Borui Zhang[1],
Wenzhao Zheng[1], Dalong Du[3], Jie Zhou[1], Jiwen Lu[1]*
[1]Department of Automation, Tsinghua University, China
[2]School of Electrical and Information Engineering, Tianjin University, China
[3]PhiGent Robotics
linqingzhao@tju.edu.cn; {xxw21,wang-zw18,zhang-br21,zhengwz18}@mails.tsinghua.edu.cn;
yunpengzhang97@gmail.com; dalong.du@phigent.ai; {jzhou, lujiwen}@tsinghua.edu.cn

Abstract

In this paper, we present a tensor decomposition and low-rank recovery approach (LowRankOcc) for vision-based 3D semantic occupancy prediction. Conventional methods model outdoor scenes with fine-grained 3D grids, but the sparsity of non-empty voxels introduces considerable spatial redundancy, leading to potential overfitting risks. In contrast, our approach leverages the intrinsic low-rank property of 3D occupancy data, factorizing voxel representations into low-rank components to efficiently mitigate spatial redundancy without sacrificing performance. Specifically, we present the Vertical-Horizontal (VH) decomposition block factorizes 3D tensors into vertical vectors and horizontal matrices. With our "decomposition-encoding-recovery" framework, we encode 3D contexts with only 1/2D convolutions and poolings, and subsequently recover the encoded compact yet informative context features back to voxel representations. Experimental results demonstrate that LowRankOcc achieves state-of-the-art performances in semantic scene completion on the SemanticKITTI dataset and 3D occupancy prediction on the nuScenes dataset.

1. Introduction

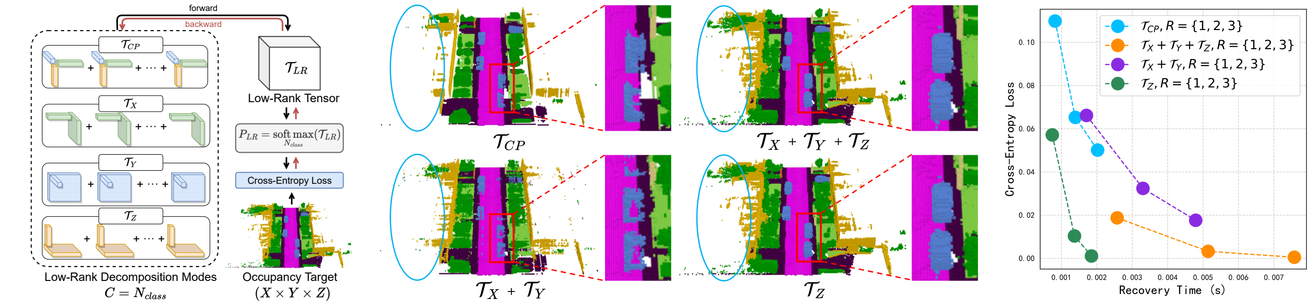
Accurate and comprehensive perception of the 3D environment is pivotal in the applications of autonomous driving systems. Vision-based 3D perception [1, 12, 29, 30, 51] has recently emerged as a promising alternative to LiDAR-based approaches [16, 32, 52], offering an effective means to extract 3D information from 2D images. Despite the absence of direct depth sensing, vision-based models, enhanced by surrounding cameras, exhibit promising performance across a spectrum of 3D perception tasks, including depth estimation [33, 40, 49, 50], semantic map reconstruction [19, 47], and 3D object detection [13, 20].

The primary challenge in vision-based 3D semantic occupancy prediction [15, 21, 24, 26, 38, 39] is the precise capture of the nuanced 3D geometry in real-world scenes. Voxel and Bird's Eye View (BEV) representations emerge as the two most widely adopted methods for tackling this challenge. Voxel representations offer detailed 3D information at the cost of computational complexity, while BEV representations prioritize efficiency and simplicity, albeit with potential information loss, especially in scenarios with complex vertical structures. TPVFormer [14] extends BEV by incorporating three orthometric perspectives: frontal, lateral, and overhead views, aiming to balance efficiency and information preservation.

However, existing methods overlook the spatial redundancy inherent in modeling 3D occupancy data, resulting in capturing noise and specific patterns that do not generalize well to new data. For example, a significant portion (over 95 % in nuScenes [3]) of voxels representing empty space is concentrated in the upper region of the scene. Conventional approaches that densely encode all voxels, regardless of their occupancy status, may inadvertently emphasize irrelevant information, leading to overfitting. This spatial redundancy highlights that not all spatial dimensions or components of 3D occupancy data carry unique and independent information.

To address the overfitting caused by spatial redundancy, in this paper, we present LowRankOcc, a tensor decomposition and low-rank recovery approach for vision-based 3D semantic occupancy prediction. Inspired by the classical tensor decomposition theories, we aim to leverage the low-rank property inherent in 3D outdoor occupancy data.

*Corresponding author.



(a) Illustration of our toy experiment

(b) Semantic predictions of different low-rank tensors ($R = 3$)

(c) Recovery time and loss w.r.t. R

Figure 1. In the toy experiment, we show the low-rank property inherent in 3D outdoor occupancy data without involving a deep neural network. (a) We compare various low-rank decomposition modes by iteratively updating randomly initialized parameters for a fixed number of steps. The optimization objective is to minimize the cross-entropy loss between the prediction of the recovered low-rank tensor and the target occupancy. (b) With the same R (for $\mathcal{T}_{VM}$, $R_X = R_Y = R_Z = R$) and updating steps, $\mathcal{T}_{CP}$ is inferior to $\mathcal{T}_{VM}$, among which $\mathcal{T}_Z$ plays a more significant role compared to $\mathcal{T}_X + \mathcal{T}_Y$. (c) As the value of R increases, $\mathcal{T}_Z$ demonstrates a superior trade-off between performance and forward recovery speed compared to $\mathcal{T}_{VM}$. To mitigate the bias of individual samples, the plot in (c) represents the results obtained by averaging over 100 randomly selected samples in SemanticKITTI.

We factorize voxel representations into low-rank components (e.g. vectors and matrices) to mitigate spatial redundancy without sacrificing performance. To achieve this, we first introduce the Vertical-Horizontal (VH) decomposition block as the cornerstone, factorizing a 3D tensor into vertical vectors and horizontal matrices. The outer product of VH components constitutes a low-rank recovery, effectively summarizing crucial information within the tensor. By stacking multiple VH decomposition blocks, we can represent dense voxel data as the sum of multiple vector-matrix outer products. To prevent VH components from capturing homogeneous information and thus losing representation capability, we introduce a recursive residual decomposition strategy, which encourages VH components to learn diverse frequency contexts while ensuring their complementarity to the original tensor. After the decomposition phase, VH components are reorganized into two temporary mini-batches specifically designed for vector and matrix parallel feature encoding, respectively. This parallel mechanism seamlessly achieves multi-scale feature fusion and propagation, presenting a compelling alternative to the resource-intensive 3D UNet. Finally, we recover multi-scale low-rank tensors and feed them into the head to predict the final occupancy results. Experimental results demonstrate that LowRankOcc outperforms existing methods by a large margin on SemanticKITTI and nuScenes.

2. Related Works

3D semantic occupancy prediction. Early approaches tackled autonomous driving perception through 3D object detection, generating 3D bounding boxes for objects based on RGB images [1, 12, 13, 20, 27–30, 42, 47, 51] or LIDAR point clouds [16, 32, 52]. While effective, 3D object detection falls short in providing detailed geometry crucial for driving planning. Consequently, recent attention has shifted towards 3D semantic occupancy prediction, aiming to delineate fine-grained spatial occupancy in voxel grids with associated semantic labels. The initial focus of 3D semantic occupancy prediction was on scene completion. SSCNet [34] pioneered research in semantic scene completion (SSC), jointly reasoning about the geometry and semantics of partially observed 3D scenes from RGB-D images. Subsequent works refined geometric representations by incorporating explicit depth information. MonoScene [4] introduced the first SSC method for outdoor scenes using only RGB inputs, leveraging a 3D Unet to process voxel representations back-projected from visual inputs. TPVFormer [14] adopted a unique approach, representing the 3D scene with three orthogonal 2D planes and predicting occupancy for each 3D coordinate through triple-plane interpolation. NDCScene [44] extended 2D feature maps to 3D space by progressively restoring depth dimensions with deconvolution operations. OccFormer [48] proposes to capture the fine-grained details and scene-level layouts with the local and global pathways. SurroundOcc [41] contributed a densely reconstructed occupancy dataset and an effective 3D convolution-based decoder for more nuanced occupancy predictions. However, these methods typically overlook the spatial redundancy inherent in 3D occupancy data, resulting in challenges related to storage and computation.

Tensor decomposition and low-rank tensor recovery. Tensor decomposition theory [5, 6] provides an efficient means of representing tensors through linear combinations of low-rank tensors, acting as principal components. In computer vision, this has been applied for tasks like accelerating convolutions [17, 46], semantic segmentation [7], and model compression [45]. Tucker and CP decompositions are two prevalent methods, with Tucker expressing tensors as matrices and a core tensor, and CP representing tensors as a sum of rank-1 tensors. Low-rank tensor recovery aims to estimate desired tensors under a lower-rank constraint. In this paper, we introduce a "decomposition-encoding-recovery" framework, relying solely on 1/2D convolutions and pooling operations to capture 3D contexts. This approach avoids the computational complexity associated with 3D convolutions.

3. Approach

3.1. Low-Rank 3D Scene Representation

Let $\mathcal{T} \in \mathbb{R}^{X \times Y \times Z \times C}$ be the dense voxel representations, where X, Y, Z are the spatial resolution and C refers to the channel number. In autonomous driving scenarios, the complexity of $\mathcal{O}(XYZ)$ not only incurs excessive storage and computational burdens but also leads to overfitting issues caused by its rich spatial redundancy. This key observation motivates us to explore the tensor low-rank representation as an alternative to dense voxels in the spatial perspective. Without loss of generality, we omit the channel dimension in the following equations.

Decomposing tensors into low-rank components. An intuitive way to reduce the redundancy in high-rank tensors is to employ the classical canonical-polyadic (CP) decomposition [5], which breaks down a tensor into multiple vectors, each representing a compact rank-1 component. Given a 3D tensor $\mathcal{T} \in \mathbb{R}^{X \times Y \times Z}$, CP decomposition is expressed as:

$$\mathcal{T}_{CP} = \sum_{j=1}^{R} \lambda_j \, \mathbf{v}_j^{(X)} \circ \mathbf{v}_j^{(Y)} \circ \mathbf{v}_j^{(Z)}, \tag{1}$$

where $\mathbf{v}_j^{(X)}$, $\mathbf{v}_j^{(Y)}$, and $\mathbf{v}_j^{(Z)}$ are vector factors of the three modes for the j-th component, and the symbol $\circ$ represents the outer product operation. λ_i is a learnable scaling factor.

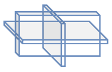
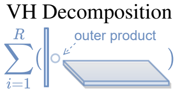
Unlike CP decomposition which utilizes pure vector factors, vector-matrix (VM) decomposition [6] factorizes a tensor into multiple vectors and matrices, incorporating the sum of the three-way vector-matrix components along each direction:

$$\begin{aligned} \mathcal{T}_{VM} =& \mathcal{T}_X + \mathcal{T}_Y + \mathcal{T}_Z \\ =& \sum_{j=1}^{R_X} \mathbf{v}_j^{(X)} \circ \mathbf{M}_j^{(Y,Z)} + \sum_{j=1}^{R_Y} \mathbf{v}_j^{(Y)} \circ \mathbf{M}_j^{(X,Z)} + \sum_{j=1}^{R_Z} \mathbf{v}_j^{(Z)} \circ \mathbf{M}_j^{(X,Y)} \end{aligned} \tag{2}$$

where $\mathbf{M}_j^{(Y,Z)}$, $\mathbf{M}_j^{(X,Z)}$, and $\mathbf{M}_j^{(X,Y)}$ are the matrix factors, allowing two rank-1 vectors to be expanded into matrices of arbitrary rank.

To prove the low-rank property in 3D outdoor occupancy data, we conduct a toy experiment (see Figure 1) to compare various low-rank decomposition modes, i.e., $\mathcal{T}_{CP}$, $\mathcal{T}_X$, $\mathcal{T}_Y$, and $\mathcal{T}_Z$. The qualitative and quantitative comparisons highlight that the z-axis component $\mathcal{T}_Z$ exhibits a superior trade-off between performance and forward recovery speed compared to $\mathcal{T}_{CP}$ and $\mathcal{T}_{VM}$. The reason can be summarized into three aspects. 1) The high compactness of CP decomposition requires a substantial number of components (i.e., rank R) to effectively model complex scenes. When R is relatively small, CP decomposition struggles to fully capture the scene structures. 2) By replacing pure vector factors in CP with matrices along three axes, VM decomposition is able to capture a more intricate structure within a 3D scene. 3) Due to fewer occlusions along the z-axis in autonomous driving scenarios, which has the highest information density, $\mathcal{T}_Z$ contributes the most to VM decomposition. With increasing R, $\mathcal{T}_Z$ achieves multiple times the reconstruction speed of $\mathcal{T}_{VM}$ while maintaining comparable performance. Considering these factors, we design our LowRankOcc method based on tensor decomposition along the z-axis, denoted as Vertical-Horizontal (VH) decomposition, with R as the VH rank.

Table 1. Comparisons of the proposed VH decomposition with other representations for 3D semantic occupancy.

Voxel (Target)	BEV	TPV	VH Decomposition
			
Height modeling	✗	✓	✓
Voxel recovery	✗	✗	✓
Complexity	$\mathcal{O}(XY)$	$\mathcal{O}(XY + YZ + XZ)$	$\mathcal{O}(R(XY + Z))$

Comparison with Voxel, BEV, and TPV representations. Table 1 illustrates the difference between our VH decomposition and other efficient representations. To avoid the cubic complexity ($\mathcal{O}(XYZ)$) of voxel representations, Bird's-Eye-View (BEV) representations discard height information, offering improved efficiency with a complexity of $\mathcal{O}(XY)$. Tri-Perspective View (TPV) representations compress 3D features by projecting them onto three axis-aligned orthogonal planes, reducing the complexity to $\mathcal{O}(XY + YZ + XZ)$. However, both BEV and TPV representations lack theoretical assurance for recovering the original voxel representations. Thanks to tensor decomposition, our VH decomposition can represent voxel features with multiple vectors and matrices, achieving a complexity of $\mathcal{O}(R(XY + Z))$. Given that occupancy data is typically low-rank, R is usually much smaller than Z. Crucially, our representations can be readily converted into low-rank voxel representations through outer product operations, making them versatile for various downstream tasks.

3.2. Tensor Residual Decomposition and Recovery

Given a pre-defined VH rank R, our VH decomposition aims to model a tensor factorization, which is essentially a function $\mathbf{P}$ that maps any tensor $\mathcal{T} \in \mathbb{R}^{X \times Y \times Z \times C}$ to its vector factors $\mathbf{v} = \{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_R\}$, and its matrix factors $\mathbf{M} = \{\mathbf{M}_1, \mathbf{M}_2, \cdots, \mathbf{M}_R\}$, where $\mathbf{v}_i \in \mathbb{R}^{Z \times C}$ is the i-th rank-1 vector component, and $\mathbf{M}_i \in \mathbb{R}^{X \times Y \times C}$ denotes the i-th matrix component. To achieve this decomposition, we introduce a convolution block (see Figure 2) to employ spa-

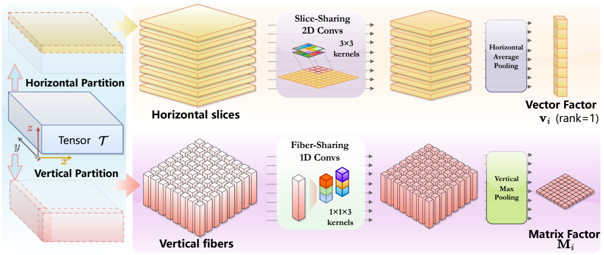


Figure 2. Illustration of the i-th Vertical-Horizontal (VH) decomposition block. By incorporating spatial partitioning alongside 1/2D convolutions and pooling, we can capture 3D contexts without computationally heavy 3D convolutions.

tial partitioning and encoding on voxel features. This partition yields Z horizontal slices and $X \times Y$ vertical fibers. For the horizontal slices $\mathcal{S}_i$, we first leverage two slice-sharing convolution blocks Conv_{2d} with a stride of 2 to effectively summarize the horizontal information from each slice. Subsequently, global average pooling (GAP) is applied to further distill each slice into a single point. Consequently, we obtain the vector component $\mathbf{v}_i$ by abstracting information from all slices:

$$\mathbf{v}_i = \text{GAP}(\text{Conv}_{2d}(\mathcal{S}_i)). \tag{3}$$

Concerning the vertical fibers $\mathcal{F}_i$, each can be regarded as a vector with a spatial shape of $1 \times 1 \times Z$. Hence, we can straightforwardly apply two fiber-sharing 1D convolutions Conv_{1d} to enhance the vertical receptive field. Given the relatively low information density in the vertical direction, we perform max pooling (GMP) to compress each fiber into a single point, enabling the extraction of the matrix component $\mathbf{M}_i$:

$$\mathbf{M}_i = \text{GMP}(\text{Conv}_{1d}(\mathcal{F}_i)). \tag{4}$$

By extending this subblock, we can effectively model the i-th decomposition $\mathbf{P}_i$ with spatial partition and 1/2D convolutions and poolings, thereby avoiding computationally heavy 3D convolutions.

To acquire R pairs of vector and matrix components, an intuitive way is to employ R distinct VH decompositions $\mathbf{P}$ on $\mathcal{T}$. The i-th reconstructed tensor, $\mathcal{T}_i$, is given by:

$$\{\mathbf{v}_i, \mathbf{M}_i\} = \mathbf{P}_i(\mathcal{T}), \quad \mathcal{T}_i = \mathbf{v}_i \circ \mathbf{M}_i, \tag{5}$$

where $\mathbf{v}_i$ is aligned with the z-axis, and $\mathbf{M}_i$ is vertical with the z-axis. This strategy is termed tensor concurrent decomposition and recovery (TCDR). However, there is no guarantee that the sum of all recovered tensors, $\sum_{i=1}^{R} \lambda_i \mathcal{T}_i$, can effectively reconstruct $\mathcal{T}$ since they might capture homogeneous information.

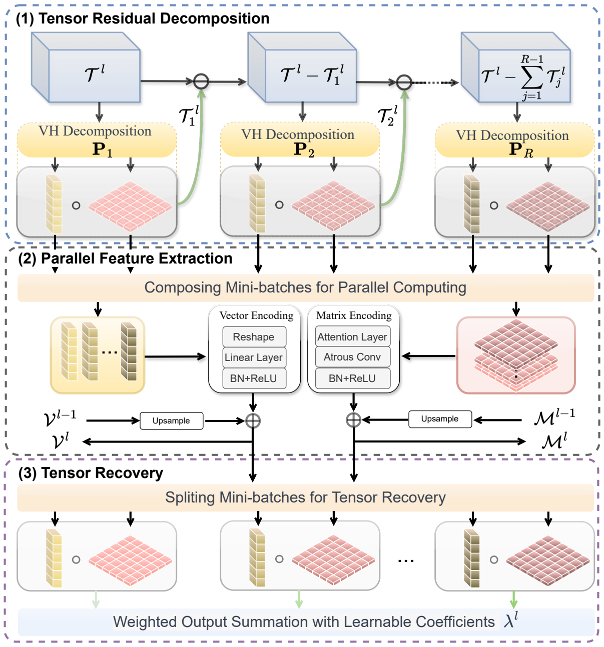


Figure 3. Illustration of the proposed Tensor Residual Decomposition and Recovery (TRDR) module. The input tensor undergoes an initial recursive factorization into R pairs of VH components. Subsequently, these VH components are regrouped into temporary mini-batches to facilitate the efficient extraction of contextual priors. Finally, the recovered low-rank tensor is computed through a weighted summation of the updated VH components.

Recursive residual decomposition. To address this challenge, we propose that the initial reconstructed tensor components focus on low-frequency information in the scene, such as the scene layout and larger objects. Once these low-frequency details are sufficiently captured, the subsequent tensor components shift their focus to the remaining high-frequency information, including finer details of object shapes and smaller objects. To achieve this, we introduce a recursive residual decomposition (RRD) strategy (refer to Figure 3) to iteratively decompose the residual part of the tensor. Thus, VH components can learn discriminative low-rank tensors of different frequencies. Specifically, we apply the first VH decomposition block $\mathbf{P}_1$ to the input tensor $\mathcal{T}$ to generate $\mathbf{v}_1$ and $\mathbf{M}_1$, from which the recovered tensor can be computed as $\mathcal{T}_1 = \mathbf{v}_1 \circ \mathbf{M}_1$. Instead of using the input tensor $\mathcal{T}$ as the decomposition target for $\mathbf{P}_2$, we then extract the residual part of the tensor, i.e., $\mathcal{T} - \mathcal{T}_1$, which can be considered as the information with a higher frequency that $\mathcal{T}_1$ fails to restore. Following this strategy, the decomposition target for $\mathbf{P}_3$ is $\mathcal{T} - \mathcal{T}_1 - \mathcal{T}_2$. Formally, given the pre-defined VH rank R, the factorized vector and matrix components in Equation 5 can be rewritten as:

$$\{\mathbf{v}_i, \mathbf{M}_i\} = \begin{cases} \mathbf{P}_i(\mathcal{T}), & \text{if } i = 1, \\ \mathbf{P}_i(\mathcal{T} - \sum_{j=1}^{i-1} \mathbf{v}_j \circ \mathbf{M}_j), & \text{if } i > 1, \end{cases} \tag{6}$$

where each VH decomposition block is tasked with generating components containing distinctive frequency contexts while ensuring their sufficient complementarity for preserving crucial information of the input tensor.

Parallel feature encoding. Having obtained all vector factors $\mathbf{v} = \{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_R\}$ and matrix factors $\mathbf{M} = \{\mathbf{M}_1, \mathbf{M}_2, \cdots, \mathbf{M}_R\}$, our objective is to extract features both vertically and horizontally to capture richer contextual priors. This offers two advantages compared to directly extracting features from 3D tensors. Firstly, the computational cost of increasing the receptive field on 3D features is quite high. However, with 1D or 2D features, we can more economically enhance the long-range semantic modeling, thereby better capturing macro-level information such as scene structure and road shapes. For 1D vectors, we utilize a linear layer, while for 2D matrices, we employ a combination of a windowed self-attention layer and a dilated convolution to encode contexts, as shown in Figure 3 (2). Secondly, the decomposed vectors and matrices can be combined into temporary mini-batches before feature encoding:

$$\{\mathbf{v}_1, \mathbf{M}_1\}, \cdots, \{\mathbf{v}_R, \mathbf{M}_R\} \longrightarrow \{\mathbf{v}_1, \cdots \mathbf{v}_R\}, \{\mathbf{M}_1, \cdots \mathbf{M}_R\}. \quad (7)$$

The two obtained mini-batches will undergo parallel processing and then be split back for tensor recovery. The advantages of these temporary mini-batches will be further discussed in the network architecture (see Section 3.3).

Low-rank tensor recovery. After the decomposition-based 3D feature encoding and cross-scale interaction, the recovered low-rank tensor $\mathcal{T}_{LR}$ can be expressed as follows:

$$\mathcal{T}_{LR} = \sum_{i=1}^{R} \lambda_i \mathcal{T}_i = \sum_{i=1}^{R} \lambda_i \cdot \mathbf{v}'_i \circ \mathbf{M}'_i, \quad (8)$$

where $\mathbf{v}'$ and $\mathbf{M}'$ are the updated VH components, and λ_i represents a parameter with gradients, allowing the network to learn it autonomously. By employing tensor parallel decomposition and recovery modules, the 3D tensor can be transformed into a low-rank tensor of identical size using a ”decomposition-encoding-recovery” manner, all without the need for 3D convolutions.

3.3. Network Architecture

Image encoding and feature lifting. The overall pipeline of LowRankOcc is illustrated in Figure 4. Comprising a backbone network for extracting multi-scale features and a neck for further fusion, the image encoder produces a fused feature map with a resolution reduced to 1/16 of the input. We denote the extracted features as $\mathcal{F}_{2d} \in \mathbb{R}^{N \times Cimg \times H \times W}$, where N represents the number of cameras, C_{img} is the channel number, and H and W refer to the image resolution. To transform 2D image features into 3D tensors, we employ the widely-used LSS [28] paradigm for image-to-3D lifting. Specifically, the encoded image features $\mathcal{F}_{2d}$ are processed to generate the context feature $\mathcal{F}_{con} \in \mathbb{R}^{N \times C_{con} \times H \times W}$ and a multi-bin depth distribution $\mathcal{D} \in \mathbb{R}^{N \times D \times H \times W}$, where D is the number of discrete bins. Each element in $\mathcal{D}$ represents a probability within the range of $[0, 1]$. Then we combine the context feature and depth distribution via an outer product to create a dense point cloud representation $\mathcal{P} = \mathcal{F}_{con} \circ \mathcal{D}$. Finally, we conduct voxel-pooling to generate the 3D feature volume $\mathcal{T} \in \mathbb{R}^{X \times Y \times Z \times C_{con}}$. The multi-scale 3D features can be generated by applying several downsampling convolutions.

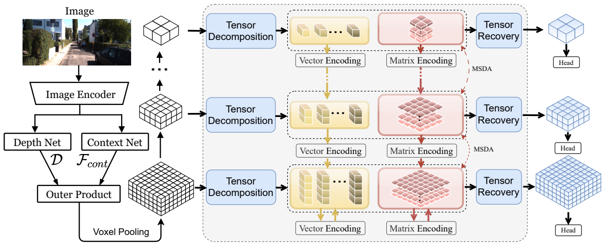


Figure 4. The pipeline of our LowRankOcc. The image encoder first extracts multi-scale features, which are then lifted to multi-scale 3D feature volumes through depth predictions and voxel-pooling. The 3D features undergo further factorization into vector and matrix components, followed by processing through the decoder composed of multi-scale TRDR modules. Finally, we recover and feed multi-scale low-rank tensors into the head to predict the final results. For simplicity, skip connections are omitted.

3D encoding via multi-scale TRDR modules. State-of-the-art methods commonly utilize feature pyramids to generate representations at different scales. With the advantage of regrouped temporary mini-batches, we can effortlessly achieve cross-scale feature fusion and propagation. This flexibility enables our TRDR module to be easily extended as an UNet-like architecture, presenting a viable alternative to the 3D UNet [41] designed for voxel representations. Specifically, we construct a 1D UNet and a 2D UNet to capture 3D contexts from both vertical and horizontal directions. Concerning multi-scale matrix components, given that each feature level highlights distinct aspects of both low-level details and high-level semantics, we utilize multi-scale deformable attention [53] (MSDA) to enhance interactions within and across scales effectively.

Occupancy head and loss. The recovered multi-scale low-rank tensors are then input into the occupancy heads to produce the final outputs. Following OccFormer [48], we employ a vanilla Mask2Former [9] as the 3D semantic occupancy head. This involves bipartite matching between the predicted and ground-truth segments, focusing only on the sampled positions. The matching cost contains the class loss and the binary mask loss. Using the Hungarian algorithm, we compute the optimal matching and derive the mask classification loss L_{mc} based on the matching cost. Additionally, the depth distribution $\mathcal{D}$ for view transformation is supervised by LiDAR points with BCE loss L_d. The final training loss is a summation of two terms: $L = L_{mc} + L_d$.

Table 2. Semantic scene completion results on SemanticKITTI test set. * represents these methods are adapted for the RGB inputs, which are implemented and reported in MonoScene [4]. Our method outperforms all published monocular methods for semantic scene completion in both the SC IoU and the SSC mIoU.

Method	Input Modality	SC IoU	SSC mIoU	road (15.30%)	sidewalk (11.13%)	parking (1.12%)	other-grnd (0.56%)	building (14.1%)	car (3.92%)	truck (0.16%)	bicycle (0.03%)	motorcycle (0.03%)	other-veh. (0.20%)	vegetation (39.3%)	trunk (0.51%)	terrain (9.17%)	person (0.07%)	bicyclist (0.07%)	motorcyclist. (0.05%)	fence (3.90%)	pole (0.29%)	traf.-sign (0.08%)
LMSCNet* [31]	Camera	31.38	7.07	46.70	19.50	13.50	3.10	10.30	14.30	0.30	0.00	0.00	0.00	10.80	0.00	10.40	0.00	0.00	0.00	5.40	0.00	0.00
3DSketch* [8]	Camera	26.85	6.23	37.70	19.80	0.00	0.00	12.10	17.10	0.00	0.00	0.00	0.00	12.10	0.00	16.10	0.00	0.00	0.00	3.40	0.00	0.00
AICNet* [18]	Camera	23.93	7.09	39.30	18.30	19.80	1.60	9.60	15.30	0.70	0.00	0.00	0.00	9.60	1.90	13.50	0.00	0.00	0.00	5.00	0.10	0.00
JS3C-Net* [43]	Camera	34.00	8.97	47.30	21.70	19.90	2.80	12.70	20.10	0.80	0.00	0.00	4.10	14.20	3.10	12.40	0.00	0.20	0.20	8.70	1.90	0.30
MonoScene [4]	Camera	34.16	11.08	54.70	27.10	24.80	5.70	14.40	18.80	3.30	0.50	0.70	**4.40**	14.90	2.40	19.50	1.00	1.40	0.40	11.10	3.30	2.10
TPVFormer [14]	Camera	34.25	11.26	55.10	27.20	27.40	6.50	14.80	19.20	**3.70**	1.00	0.50	2.30	13.90	2.60	20.40	1.10	**2.40**	0.30	11.00	2.90	1.50
OccFormer [48]	Camera	34.53	12.32	55.90	**30.30**	**31.50**	6.50	15.70	**21.60**	1.20	1.50	1.70	3.20	16.80	3.90	**21.30**	2.20	1.10	0.20	11.90	3.80	3.70
SurroundOcc [41]	Camera	34.72	11.86	**56.90**	28.30	30.20	6.80	15.20	20.60	1.40	1.60	1.20	**4.40**	14.90	3.40	19.30	1.40	2.00	0.10	11.30	3.90	2.40
LowRankOcc (ours)	Camera	**38.47**	**13.56**	52.80	27.20	25.10	**8.80**	**22.10**	20.90	2.90	**3.30**	**2.70**	**4.40**	**22.90**	**8.90**	20.80	**2.40**	1.70	**2.30**	**14.40**	**7.00**	**7.00**

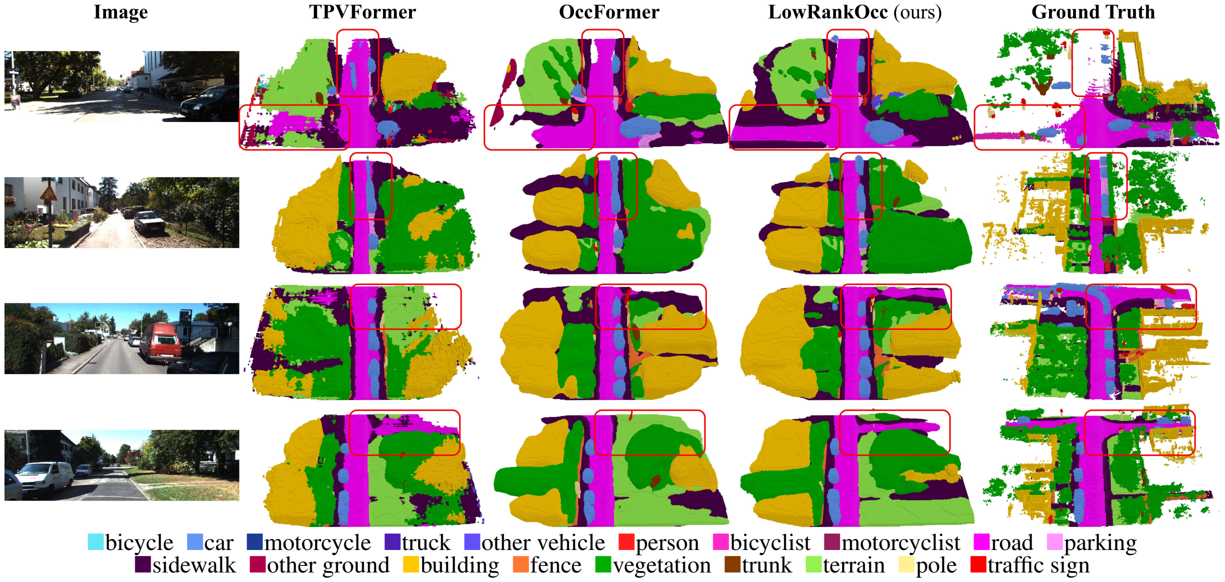


Figure 5. Qualitative results on SemanticKITTI validation set. For the convenience of comparing results from different methods, we have highlighted areas with significant differences using red boxes.

4. Experiments

4.1. Experimental Setup

Datasets. SemanticKITTI [2] is an extension of the well-known KITTI Odometry Benchmark [10]. We evaluate our method with the monocular left images as input following MonoScene [4]. The ground-truth semantic occupancy is expressed through $256 \times 256 \times 32$ voxel grids, each voxel measuring 0.2m in size. These grids are annotated with 21 semantic classes, comprising 19 specific semantics, one for free space, and one for unknown regions. We follow the official data split for comprehensive assessment. The ablation study is conducted on the SeamnticKITTI validation set. **nuScenes** [3] is known as a large-scale autonomous driving dataset, captured in Boston and Singapore. In our evaluation, we leverage the dense annotations from SurroundOcc. The occupancy prediction range for our method spans $[-50m, 50m]$ along the X and Y axes, and $[-5m, 3m]$ along the Z axis. The ground-truth semantic occupancy is expressed through $200 \times 200 \times 16$ voxel grids, each voxel measuring 0.5m in size. The input image resolution is 1600x900. To ensure consistency, we adopt the ground truth provided by SurroundOcc [41] and adhere to the official data split.

Implementation details. We employ EfficientNetB7 [4] for SemanticKITTI and ResNet101-DCN [11, 53], initialized with weights from FCOS3D [36], as the backbone to extract image features. The overall network architecture incorporates $L = 4$ levels to capture multi-scale image features. Each level comprises $R = 4$ VH decomposition blocks to generate low-rank components. Notably, the vector and matrix components share the same spatial (ex-

Table 3. 3D semantic occupancy prediction results on the nuScenes validation set. * represents these methods are adapted for the outdoor multi-camera settings, which are implemented and reported in SurroundOcc [41]. Our method outperforms all published multi-camera methods for semantic scene completion in both the SC IoU and the SSC mIoU.

Method	SC IoU	SSC mIoU	barrier ■	bicycle ■	bus ■	car ■	const. veh. ■	motorcycle ■	pedestrian ■	traffic cone ■	trailer ■	truck ■	drive. suf. ■	other flat ■	sidewalk ■	terrain ■	manmade ■	vegetation ■
MonoScene* [4]	23.96	7.31	4.03	0.35	8.00	8.04	2.90	0.28	1.16	0.67	4.01	4.35	27.72	5.20	15.13	11.29	9.03	14.86
Atlas* [25]	28.66	15.00	10.64	5.68	19.66	24.94	8.90	8.84	6.47	3.28	10.42	16.21	34.86	15.46	21.89	20.95	11.21	20.54
BEVFormer [22]	30.50	16.75	14.22	6.58	23.46	28.28	8.66	10.77	6.64	4.05	11.20	17.78	37.28	18.00	22.88	22.17	13.80	22.21
TPVFormer [14]	30.86	17.10	15.96	5.31	23.86	27.32	9.79	8.74	7.09	5.20	10.97	19.22	38.87	21.25	24.26	23.15	11.73	20.81
OccFormer [48]	31.39	19.03	18.65	10.41	23.92	30.29	10.31	14.19	13.59	10.13	12.49	20.77	38.78	19.79	24.19	22.21	13.48	21.35
SurroundOcc [41]	31.49	20.30	20.59	11.68	28.06	30.86	**10.70**	**15.14**	**14.09**	12.06	14.38	22.26	37.29	**23.70**	24.49	22.77	14.89	21.86
LowRankOcc (ours)	**32.78**	**21.51**	**22.49**	**12.45**	**30.32**	**33.63**	10.35	14.31	13.67	**12.40**	**15.09**	**25.99**	**39.52**	23.21	**26.67**	**25.19**	**16.23**	**22.66**

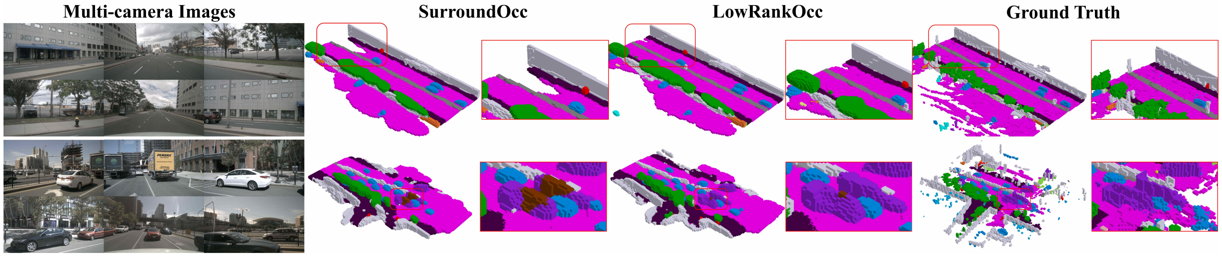


Figure 6. Qualitative results on nuScenes validation set. For the convenience of comparing results from different methods, we have highlighted areas with significant differences using red boxes and provided enlarged views.

cluding the compressed dimension) and channel dimensions with the decomposed voxel features. For matrix components, we apply multi-scale deformable self-attention with 6 layers. Voxel-pooling is applied to generate the initial 3D feature volume, with dimensions $\frac{X}{2} \times \frac{Y}{2} \times \frac{Z}{2}$. The occupancy head aligns closely with the OccFormer implementation [48], featuring 384 channels. Model training spans 30 epochs for SemanticKITTI and 25 epochs for nuScenes. We utilize the AdamW [23] optimizer with an initial learning rate of 1e-4 and weight decay of 0.01, employing a multi-step scheduler for learning rate decay. All models are trained with a batch size of 8. We borrow the data augmentation settings of OccFormer, including random resize, rotation, and flip for the image space, along with 3D flipping for the volume space. It's worth noting the exclusion of random flip augmentation along the z-axis for stationary training. Our evaluation metric is consistent with MonoScene [4].

4.2. Main Results

SemanticKITTI. As shown in Table 2, we present a comprehensive quantitative comparison of various monocular methods tackling the semantic scene completion task on the SemanticKITTI test set. Notably, LowRankOcc emerges as the frontrunner, surpassing all existing competitors, particularly excelling in the more challenging aspect of semantic scene completion. A notable performance improvement is observed compared to recent approaches such as TPVFormer [14], OccFormer [48], and SurroundOcc [41]. These observations also prove the valuable application of low-rank representation in 3D occupancy prediction. This is because low-rank representation effectively mitigates overfitting issues arising from spatial redundancy, thereby enhancing the model's generalization capability. In Figure 5, we present qualitative comparisons with state-of-the-art methods. In contrast to TPVFormer and OccFormer, our LowRankOcc not only captures finer details of object shapes (e.g., in the 2nd row) but also generates a more realistic and holistic perception of the scene. Notably, in distant regions (e.g., the 3rd and 4th rows), and even in regions not visible in the input image (1st row), our method accurately predicts intricate structures such as crossroads. This capability can be attributed to our low-rank recovery framework, which can efficiently increase the receptive field to capture long-range scene priors. This is especially advantageous for methods based on dense voxel representations, given their cubic complexity in computations.

nuScenes. Thanks to recent works [35, 37, 41] that provide densely annotated labels, we are empowered to directly compare occupancy prediction against other methods. Table 3 shows the quantitative comparison of various multi-camera methods tackling the 3D semantic occupancy prediction task on the nuScenes validation set. Notably, our LowRankOcc outperforms state-of-the-art methods by a significant margin, achieving at least a 1% improvement in both SC IoU and SSC mIoU. The quantitative results prove that the "decomposition-encoding-recovery" framework is

Table 4. Ablation on the VH decomposition-based 3D encoding.

Vector encoding	Matrix encoding	Params	Memory	IoU↑	mIoU↑
✓		16.4M	5.2G	30.01	7.14
	✓	80.0M	7.8G	36.18	12.39
✓	✓	89.5M	8.4G	**37.85**	**14.21**
3D UNet [41]		138.2M	11.9G	37.52	13.42
3D TPV + point query [14]		88.9M	6.7G	35.11	11.53
CP Decompositon [5]		16.2M	5.2G	31.10	7.32
VM Decompositon [6]		113.4M	9.4G	37.69	13.25

Table 5. Ablation on the VH rank and VH decomposition strategy.

VH rank R	Decomposition Strategy	Memory	IoU↑	mIoU↑
1	direct	5.7G	34.4	9.9
2	recursive	6.9G	36.41	12.76
3	recursive	7.6G	36.97	13.24
4	recursive	8.4G	**37.85**	**14.21**
4	concurrent	8.4G	36.15	12.46
5	recursive	9.3G	36.77	13.77

a general contribution applicable to both monocular and multi-camera settings. In Figure 6, we present qualitative comparisons with the pure 3D convolution-based method SurroundOcc [41]. Specifically, we present two types of samples: one displaying a clear view in a smoothly flowing traffic scene (1st row), and the other illustrating crowded scenes with multiple vehicles (2nd row). In the first sample, low-frequency information includes structurally simple scene layouts and coarse-grained object shapes. Our Recursive Residual Decomposition strategy effectively captures this low-frequency context, contributing to obtaining more precise drivable areas. In the second sample, high-frequency information involves higher-level semantic relationships and fine-grained details of object shapes.

4.3. Ablation Studies

VH decomposition-based 3D encoding. In Table 4, we ablate the VH decomposition-based 3D encoding and compare it with other baseline methods. The last two rows of Table 4 are obtained by altering the VH block to CP and VM blocks. For CP and VM, we computed three orthonormal vector factors and three-way vector-matrix pairs, respectively. Firstly, the utilization of 1D vertical vectors and 2D horizontal matrices in encoding proves to offer distinct contextual information. Combining these elements into a unified 3D encoding yields a further performance boost. Secondly, compared with dense 3D convolutions and TPV-based point querying, we demonstrate that the VH decomposition-based 3D encoding strikes a favorable balance between model complexity and performance. Thirdly, compared with CP and VM decomposition, vertical-horizontal representation proved to be a more efficient decomposition strategy.

VH rank R and VH decomposition strategy. Table 5 shows an ablation study focusing on VH rank and VH decomposition strategy. The IoU scores demonstrate that a higher VH rank contributes to enhanced scene understanding, but the performance gains diminish beyond a certain rank (i.e., $R > 4$). Additionally, the chosen decomposition strategy plays a crucial role in influencing the ability of low-rank tensors to capture 3D contexts. We provide a visual comparison between different values of R in Figure 7. With the increase in R, our method shows a progression from weak to strong representational capacity. This reveals the substantial spatial redundancy in voxel representations, successfully addressed by our low-rank decomposition.

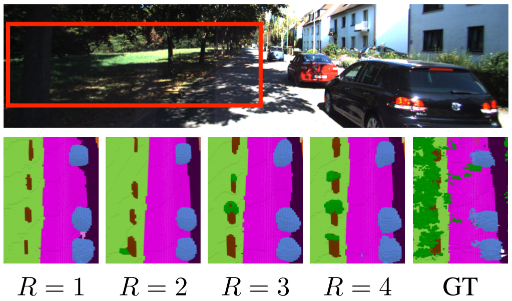


Figure 7. Ablation study on the value of VH rank R.

Table 6. Ablation on parallel multi-scale (MS) feature interaction.

Method	Tem. mini-batch	Latency	Memory	IoU↑	mIoU↑
w/o MS inter.	✗	0.45s	8.0G	37.21	13.87
w/o MS inter.	✓	0.28s	8.0G	37.21	13.87
w. MS inter.	✗	0.48s	8.4G	37.85	14.21
w. MS inter.	✓	0.32s	8.4G	37.85	14.21

Parallel multi-scale feature interaction. Table 6 shows that employing multi-scale feature interaction is beneficial for improving low-level and high-level feature fusion, with a reasonable increase in computational cost. Temporary mini-batches significantly accelerated this process, which is also an advantage brought by our VH decomposition.

5. Conclusion

We propose LowRankOcc to address spatial redundancy in 3D semantic occupancy prediction, leveraging the inherent low-rank property of occupancy data. VH decomposition, complemented by a residual learning strategy, enables compact yet informative 3D context encoding with minimal computational overhead. Experimental results demonstrate that LowRankOcc achieves state-of-the-art performances in semantic scene completion on the SemanticKITTI dataset and 3D occupancy prediction on the nuScenes dataset.

Acknowledgement The authors express their sincere gratitude to Prof. Zhanjie Song for many valuable suggestions that helped to improve the paper. This work was supported in part by the National Key Research and Development Program of China under Grant 2022ZD0160102, and in part by the National Natural Science Foundation of China under Grant 62125603, Grant 62321005, Grant 62336004.

References

[1] Adil Kaan Akan and Fatma Güney. Stretchbev: Stretching future instance prediction spatially and temporally. In *ECCV*, 2022. 1, 2

[2] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *ICCV*, 2019. 6

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

[4] Anh-Quan Cao and Raoul de Charette. Monoscene: Monocular 3d semantic scene completion. In *CVPR*, 2022. 2, 6, 7

[5] J Douglas Carroll and Jih-Jie Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of "eckart-young" decomposition. *Psychometrika*, 35(3):283–319, 1970. 2, 3, 8

[6] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *ECCV*, pages 333–350, 2022. 2, 3, 8

[7] Wanli Chen, Xinge Zhu, Ruoqi Sun, Junjun He, Ruiyu Li, Xiaoyong Shen, and Bei Yu. Tensor low-rank reconstruction for semantic segmentation. In *ECCV*, pages 52–69, 2020. 2

[8] Xiaokang Chen, Kwan-Yee Lin, Chen Qian, Gang Zeng, and Hongsheng Li. 3d sketch-aware semantic scene completion via semi-supervised structure prior. In *CVPR*, 2020. 6

[9] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 5

[10] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012. 6

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 6

[12] Anthony Hu, Zak Murez, Nikhil Mohan, Sofía Dudas, Jeffrey Hawke, Vijay Badrinarayanan, Roberto Cipolla, and Alex Kendall. Fiery: future instance prediction in bird's-eye view from surround monocular cameras. In *ICCV*, 2021. 1, 2

[13] Junjie Huang, Guan Huang, Zheng Zhu, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021. 1, 2

[14] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. *arXiv preprint arXiv:2302.07817*, 2023. 1, 2, 6, 7, 8

[15] Haoyi Jiang, Tianheng Cheng, Naiyu Gao, Haoyang Zhang, Wenyu Liu, and Xinggang Wang. Symphonize 3d semantic scene completion with contextual instance queries. *arXiv preprint arXiv:2306.15670*, 2023. 1

[16] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *CVPR*, 2019. 1, 2

[17] Vadim Lebedev, Yaroslav Ganin, Maksim Rakhuba, Ivan Oseledets, and Victor Lempitsky. Speeding-up convolutional neural networks using fine-tuned cp-decomposition. *arXiv preprint arXiv:1412.6553*, 2014. 2

[18] Jie Li, Kai Han, Peng Wang, Yu Liu, and Xia Yuan. Anisotropic convolutional networks for 3d semantic scene completion. In *CVPR*, 2020. 6

[19] Tianyu Li, Li Chen, Xiangwei Geng, Huijie Wang, Yang Li, Zhenbo Liu, Shengyin Jiang, Yuting Wang, Hang Xu, Chunjing Xu, et al. Topology reasoning for driving scenes. *arXiv preprint arXiv:2304.05277*, 2023. 1

[20] Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. *arXiv preprint arXiv:2206.10092*, 2022. 1, 2

[21] Yiming Li, Zhiding Yu, Christopher Choy, Chaowei Xiao, Jose M Alvarez, Sanja Fidler, Chen Feng, and Anima Anandkumar. Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion. In *CVPR*, pages 9087–9098, 2023. 1

[22] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. *arXiv preprint arXiv:2203.17270*, 2022. 7

[23] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 7

[24] Qihang Ma, Xin Tan, Yanyun Qu, Lizhuang Ma, Zhizhong Zhang, and Yuan Xie. Cotr: Compact occupancy transformer for vision-based 3d occupancy prediction. *arXiv preprint arXiv:2312.01919*, 2023. 1

[25] Zak Murez, Tarrence Van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images. In *ECCV*, pages 414–431, 2020. 7

[26] Wenzhe Ouyang, Xiaolin Song, Bailan Feng, and Zenglin Xu. Octocc: High-resolution 3d occupancy prediction with octree. In *AAAI*, pages 4369–4377, 2024. 1

[27] Lang Peng, Zhirong Chen, Zhangjie Fu, Pengpeng Liang, and Erkang Cheng. Bevsegformer: Bird's eye view semantic segmentation from arbitrary camera rigs. *arXiv preprint arXiv:2203.04050*, 2022. 2

[28] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *ECCV*, 2020. 5

[29] Cody Reading, Ali Harakeh, Julia Chae, and Steven L Waslander. Categorical depth distribution network for monocular 3d object detection. In *CVPR*, 2021. 1

[30] Thomas Roddick and Roberto Cipolla. Predicting semantic map representations from images using pyramid occupancy networks. In *CVPR*, 2020. 1, 2

[31] Luis Roldao, Raoul de Charette, and Anne Verroust-Blondet. Lmscnet: Lightweight multiscale 3d semantic completion. In *3DV*, 2020. 6

[32] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *CVPR*, 2019. 1, 2

[33] Yunxiao Shi, Hong Cai, Amin Ansari, and Fatih Porikli. Egadepth: Efficient guided attention for self-supervised multicamera depth estimation. In *CVPR*, pages 119–129, 2023. 1

[34] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *CVPR*, 2017. 2

[35] Xiaoyu Tian, Tao Jiang, Longfei Yun, Yue Wang, Yilun Wang, and Hang Zhao. Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. *arXiv preprint arXiv:2304.14365*, 2023. 7

[36] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *ICCV*, 2021. 6

[37] 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. *arXiv preprint arXiv:2303.03991*, 2023. 7

[38] Yu Wang and Chao Tong. H2gformer: Horizontal-to-global voxel transformer for 3d semantic scene completion. In *AAAI*, pages 5722–5730, 2024. 1

[39] Yuqi Wang, Yuntao Chen, Xingyu Liao, Lue Fan, and Zhaoxiang Zhang. Panoocc: Unified occupancy representation for camera-based 3d panoptic segmentation. *arXiv preprint arXiv:2306.10013*, 2023. 1

[40] Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Yongming Rao, Guan Huang, Jiwen Lu, and Jie Zhou. Surrounddepth: Entangling surrounding views for self-supervised multi-camera depth estimation. In *CoRL*, pages 539–549, 2023. 1

[41] Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving. In *ICCV*, pages 21729–21740, 2023. 2, 5, 6, 7, 8

[42] Xiuwei Xu, Chong Xia, Ziwei Wang, Linqing Zhao, Yueqi Duan, Jie Zhou, and Jiwen Lu. Memory-based adapters for online 3d scene perception. *arXiv preprint arXiv:2403.06974*, 2024. 2

[43] Xu Yan, Jiantao Gao, Jie Li, Ruimao Zhang, Zhen Li, Rui Huang, and Shuguang Cui. Sparse single sweep lidar point cloud segmentation via learning contextual shape priors from scene completion. In *AAAI*, 2021. 6

[44] Jiawei Yao, Chuming Li, Keqiang Sun, Yingjie Cai, Hao Li, Ouyang Wanli, and Hongsheng Li. Ndc-scene: Boost monocular 3d semantic scene completion in normalized device coordinates space. *arXiv preprint arXiv:2309.14616*, 2023. 2

[45] Xiyu Yu, Tongliang Liu, Xinchao Wang, and Dacheng Tao. On compressing deep models by low rank and sparse decomposition. In *CVPR*, pages 7370–7379, 2017. 2

[46] Shipeng Zhang, Lizhi Wang, Lei Zhang, and Hua Huang. Learning tensor low-rank prior for hyperspectral image reconstruction. In *CVPR*, pages 12006–12015, 2021. 2

[47] Yunpeng Zhang, Zheng Zhu, Wenzhao Zheng, Junjie Huang, Guan Huang, Jie Zhou, and Jiwen Lu. Beverse: Unified perception and prediction in birds-eye-view for vision-centric autonomous driving. *arXiv preprint arXiv:2205.09743*, 2022. 1, 2

[48] Yunpeng Zhang, Zheng Zhu, and Dalong Du. Occformer: Dual-path transformer for vision-based 3d semantic occupancy prediction. In *ICCV*, 2023. 2, 5, 6, 7

[49] Linqing Zhao, Wenzhao Zheng, Yueqi Duan, Jie Zhou, and Jiwen Lu. Sptr: Structure-preserving transformer for unsupervised indoor depth completion. *TCSVT*, 2023. 1

[50] Linqing Zhao, Yi Wei, Jiaxin Li, Jie Zhou, and Jiwen Lu. Structure-aware cross-modal transformer for depth completion. *TIP*, 33:1016–1031, 2024. 1

[51] Brady Zhou and Philipp Krähenbühl. Cross-view transformers for real-time map-view semantic segmentation. In *CVPR*, 2022. 1, 2

[52] Benjin Zhu, Zhengkai Jiang, Xiangxin Zhou, Zeming Li, and Gang Yu. Class-balanced grouping and sampling for point cloud 3d object detection. *arXiv preprint arXiv:1908.09492*, 2019. 1, 2

[53] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *ICLR*, 2021. 5, 6