Not All Voxels Are Equal: Hardness-Aware Semantic Scene Completion with Self-Distillation

Song Wang[1],   Jiawei Yu[1],   Wentong Li[1],   Wenyu Liu[1],   Xiaolu Liu[1],
Junbo Chen[2]*,   Jianke Zhu[1]*
[1]Zhejiang University   [2]Udeer.ai

Abstract

*Semantic scene completion, also known as semantic occupancy prediction, can provide dense geometric and semantic information for autonomous vehicles, which attracts the increasing attention of both academia and industry. Unfortunately, existing methods usually formulate this task as a voxel-wise classification problem and treat each voxel equally in 3D space during training. As the hard voxels have not been paid enough attention, the performance in some challenging regions is limited. The 3D dense space typically contains a large number of empty voxels, which are easy to learn but require amounts of computation due to handling all the voxels uniformly for the existing models. Furthermore, the voxels in the boundary region are more challenging to differentiate than those in the interior. In this paper, we propose HASSC approach to train the semantic scene completion model with hardness-aware design. The global hardness from the network optimization process is defined for dynamical hard voxel selection. Then, the local hardness with geometric anisotropy is adopted for voxel-wise refinement. Besides, self-distillation strategy is introduced to make training process stable and consistent. Extensive experiments show that our HASSC scheme can effectively promote the accuracy of the baseline model without incurring the extra inference cost. Source code is available at:
https://github.com/songw-zju/HASSC.*

1. Introduction

The accurate 3D perception of the surrounding environment is critical for both autonomous vehicles and robots [38, 40, 43, 60]. Early semantic scene completion works mainly focus on indoor scenes [3, 26, 33, 41]. For outdoor driving scenarios, SemanticKITTI [1] provides the first large benchmark, in which LiDAR-based methods [7, 37, 47, 49] occupy a dominant position with promising performance. Recently, vision-centric methods [16, 28, 30] have made en-
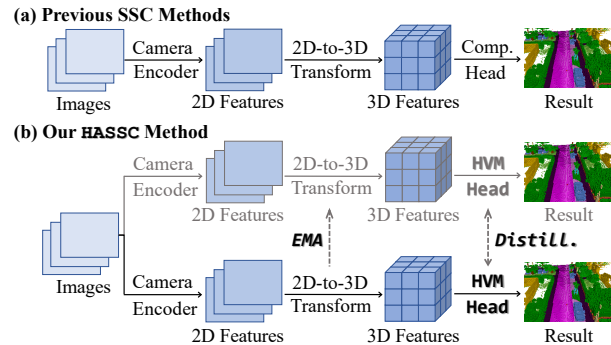
*Corresponding authors

Figure 1. Comparing our proposed hardness-aware semantic scene completion (HASSC) approach against previous semantic scene completion methods. We present an effective hard voxel mining (HVM) head with self-distillation during training.

couraging progress in bird's-eye-view (BEV) perception. Researchers have further started to complete the entire 3D semantic scene with only the camera as input and obtain impressive results [4, 17, 29].

Generally, it is not trivial to infer semantic occupancy information in 3D dense space from the current camera observation alone. For vision-centric semantic scene completion models, the 2D backbone needs high-resolution images as input and consumes most of the GPU memory, as shown in Fig. 1(a). After the forward or backward 2D-to-3D transformation [21, 29, 31], the 3D dense voxel volume features are captured while it is impossible to perform feature extraction under high-resolution in 3D space due to the GPU memory limitation. The 3D backbone adopts convolution [4, 21] or self-attention [29] operators and extracts fine-grained features on reduced-resolution 3D feature maps. As voxel-wise outputs are required at full resolution, the completion head typically derives the final result through trilinear interpolation or transposed convolution directly without considering the hardness for different voxels. Meanwhile, existing methods [4, 29] mainly formulate semantic scene completion as a voxel-wise classification problem and compute the loss for each voxel equally. Such a scheme ignores that the hardness in classifying various voxels in 3D space is quite different. Since more than 90% of the voxel space is empty,

these empty voxels are easy to predict but require a large amount of computation during training. Moreover, voxels inside an object exhibit greater predictability than those located on the boundary.

Building upon the success of hard sample mining in 2D dense prediction like object detection [32, 39] and semantic segmentation [19, 27, 48], we are motivated to design a hard voxel mining strategy in 3D dense space. The difference between 2D pixel space and 3D voxel space is not only the extra computational cost due to dimension upgrading, but also a large number of empty voxels consuming most of memory and computation in 3D voxel space. To alleviate this problem, we propose the hard voxel mining (HVM) head with self-distillation, which select hard voxels via the global hardness and refine them with local hardness, as illustrated in Fig. 1(b). Specifically, the global hardness is based on the uncertainty in predicting each voxel so that we can update the selected voxels dynamically. As most of the voxels selected in such way are empty at the early stages of training, local hardness based on geometric anisotropy is introduced to weight the losses of different voxels. The local geometric anisotropy (LGA) of a voxel is defined as the semantic difference from its neighbors. We adopt the linear mapping of LGA as the local hardness to weight the voxel-wise losses from the hard area and refine their predictions. Furthermore, self-distillation training is introduced to make the model outputs more stable and consistent. The teacher model is optimized by the exponential moving average (EMA) of the student model without extra training process. Our presented self-distillation approach works well with the hard voxel mining head to jointly improve the completion performance.

The main contributions of this work are summarized as follows:

- We propose a hardness-aware semantic scene completion (**HASSC**) scheme that can be easily integrated into existing models without incurring extra cost for inference.
- We take advantage of both the global and local hardness to find the hard voxels so that their predictions can be refined by weighted voxel-wise losses during training.
- A self-distillation training strategy is introduced to improve semantic scene completion through an end-to-end training manner.
- Extensive experiments are conducted to demonstrate the effectiveness of our presented method.

2. Related Work

Semantic Scene Completion. The methods for outdoor semantic scene completion (SSC) can be divided into two categories according to their input: 1) *LiDAR-based methods*. The grid-based approaches [37, 46] employ the occupancy grid voxelized from the sparse LiDAR point cloud as input and achieve fast inference performance with lite-

weight backbone. The point-based methods [7, 49] integrate the point-wise features within the voxel space and improve the model accuracy. Xia *et al.* [47] redesign the completion network architecture and obtain the optimal accuracy with 3D input. 2) *Camera-based methods*. MonoScene [4] is the pioneer work, which explores the SSC with monocular camera image firstly. The subsequent works construct a tri-perspective view plane [17] or design dual-path transformer decoder [56] to improve the performance with single image. VoxFormer [29] estimates the coarse geometry with stereo images first and obtains the non-empty proposals to perform deformable cross-attention [59] on single or multiple monocular images. Another method category utilizes implicit representations rather than voxel-based modeling, indicating the capability for extending SSC with both LiDAR [25, 36] and camera [12]. Since the camera is much cheaper with greater application potentials than LiDAR, we mainly focus on vision-centric methods in this paper.

Hard Sample Mining for Dense Prediction. Hard sample mining is firstly explored in 2D object detection [32, 39] as the difficulty in detecting distinct objects from an image is quite different. In 2D image segmentation, Li *et al.* [27] propose a layer cascade method to segment regions with different hardness. Yin *et al.* [53] extract hard regions according to the loss values and re-train these areas for better performance. Kirillov *et al.* [19] start from the common ground of image rendering and segmentation, and refine the edge regions of objects in the image from the feature level. Deng *et al.* [9] use an auxiliary detection network to find hard areas at nighttime and conduct segmentation refinement in both training and inference. Xiao *et al.* [48] propose a pixel hardness learning method by making use of global and historical loss values. The above methods are all designed for images in 2D pixel space. Li *et al.* [22] propose local geometric anisotropy to weight voxel-wise cross-entropy losses, which does not perform well in outdoor scenes. In this paper, we propose to conduct effective hard sample mining in 3D dense voxel space for driving scenes.

Self-Distillation. Knowledge distillation is firstly proposed to learn dark knowledge from well-trained large models for model compression [2, 15]. A series of subsequent works have been presented to improve the learning efficiency and capability of student models [10, 14, 51, 57]. In autonomous driving, knowledge distillation, especially cross-modality distillation, has shown its great potential in improving model accuracy [6, 44, 58] and compressing models [8, 50, 54]. However, these methods usually need to train a stronger teacher with more parameters or other modality at first, which incur additional training costs. Inspired by successful applications of self-distillation in other fields including 2D and 3D semantic segmentation [18, 20, 24, 55], we introduce self-distillation training strategy for semantic scene completion without extra designed models.
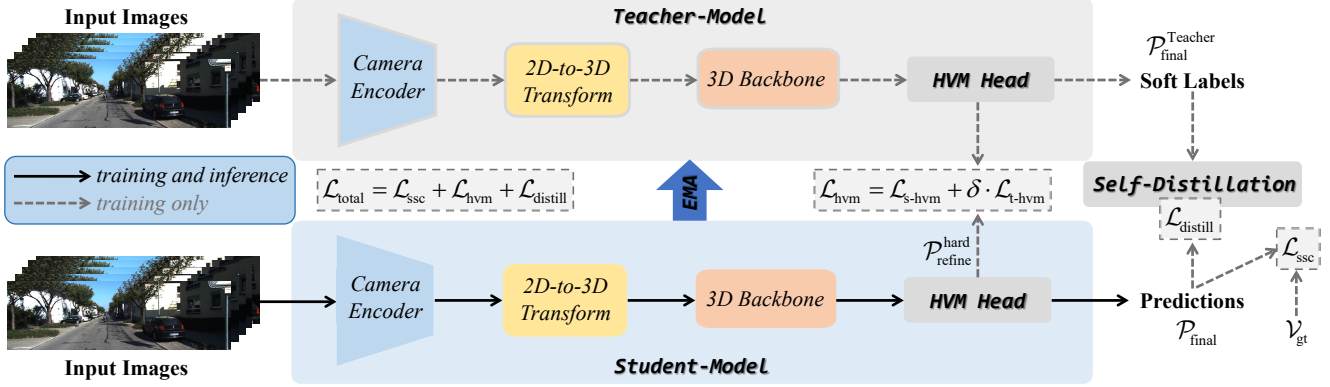
Figure 2. Overview of **Hardness-Aware Semantic Scene Completion (`HASSC`)** pipeline. We take the camera images as input and construct 3D feature volume by *Camera Encoder* and *2D-to-3D Transform*. With the fine-grained features provided by *3D Backbone*, we propose hard voxel mining (HVM) head to make the model concentrate on hard voxels. The teacher-model has the same architecture as student-model, which is updated by the exponential moving average (EMA) of student. The stable predictions can be achieved by taking advantage of both the self-distillation and HVM head.

3. Method

The semantic scene completion task predicts a dense semantic voxel volume $\mathcal{V} \in \mathbb{R}^{X \times Y \times Z}$ in front of the vehicle with only the observation from onboard sensors including camera and LiDAR. (X, Y, Z) represent the length, width and height of the 3D volume, respectively. Each voxel in $\mathcal{V}$ is either empty or occupied with one semantic class. We mainly consider the more challenging 2D input for its low cost and great application potentials.

3.1. Overview

In this work, we aim to provide a hard sample mining solution in voxel space for 3D dense prediction. The overall pipeline of Hardness-Aware Semantic Scene Completion (`HASSC`) is illustrated in Fig. 2. Our proposed hard voxel mining (HVM) head and self-distillation training strategy are independent from specific network, which can be easily integrated into off-the-shelf methods.

The typical semantic scene completion network, *e.g.*, MonoScene [4] and VoxFormer [29], consists of camera encoder, 2D-to-3D transformation, 3D backbone and completion head.

Camera Encoder. The camera encoder is made of an image backbone and a neck, which extracts semantic and geometric feature $\mathcal{F}^{2D} \in \mathbb{R}^{H' \times W' \times D'}$ from images under the perspective view. $H' \times W'$ is the 2D feature resolution, and D' is the dimension. The extracted feature is the basis to construct 3D voxel volume in the following.

2D-to-3D Transformation. The 2D-to-3D transformation in semantic scene completion is similar to view transformation in BEV perception, which can be divided into two paradigms, including forward projection [16, 21, 34] and backward projection [29, 30, 45]. With the extracted image feature $\mathcal{F}^{2D}$, we construct 3D volume feature $\mathcal{F}^{3D} \in$ $\mathbb{R}^{X' \times Y' \times Z' \times D}$ by explicit geometry estimation and query-based 3D-to-2D back projection [29]. To reduce memory consumption and computational cost, $X' \times Y' \times Z'$ as the 3D feature resolution is smaller than $X \times Y \times Z$. D is the 3D feature dimension.

3D Backbone. The 3D Backbone performs self-attention [29] or convolution [4, 21] on the voxel volume feature $\mathcal{F}^{3D}$ from the 2D-to-3D Transformation and obtain fine-grained feature $\mathcal{F}^{3D}_{\text{fine}} \in \mathbb{R}^{X' \times Y' \times Z' \times D}$. Moreover, the completion head uses $\mathcal{F}^{3D}_{\text{fine}}$ to obtain the final prediction $\mathcal{P}_{\text{final}} \in \mathbb{R}^{X \times Y \times Z \times C}$, where C is the number of total classes including empty and semantic categories.

3.2. Hard Voxel Mining Head

The completion head in SSC usually up-samples the fine-grained feature $\mathcal{F}^{3D}_{\text{fine}}$ from the 3D backbone through trilinear interpolation or transposed convolution to obtain the completion result $\mathcal{P}_{\text{final}}$ of full-resolution. Since this process does not take into account the hardness of each voxel, the performance is poor in some difficult regions. Our proposed hard voxel mining (HVM) head is based on the vanilla completion head and selects hard voxels during training to refine their predictions. In the following, we firstly introduce the definitions of global hardness and local hardness, and then explain the working flow of proposed HVM head, as illustrated in Fig. 3.

Global Hardness. With the fine-grained feature $\mathcal{F}^{3D}_{\text{fine}}$, we obtain the coarse prediction $\mathcal{P}_{\text{coarse}} \in \mathbb{R}^{X' \times Y' \times Z' \times C}$ by the single layer Multi-Layer Perceptron (MLP) and *softmax* function firstly. Let (i, j, k) denote the voxel index. For the prediction of each voxel $p_{(i,j,k)} \in \mathbb{R}^{1 \times C}$ in $\mathcal{P}_{\text{coarse}}$, we rank the probabilities of each class $\{p^1, p^2, ..., p^C\}$ in decreasing order. The largest probability in C classes is represented as p^a, and the second largest one is denoted as p^b. Then, the
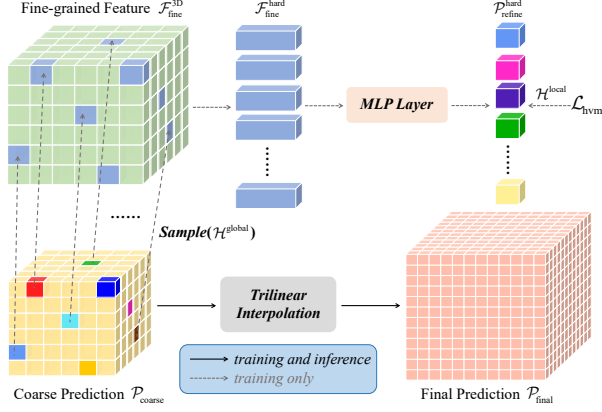
Figure 3. Illustration of **Hard Voxel Mining (HVM) Head**. At training stage, N hard voxels are selected with respect to their global hardness and random sampling. Then, we re-sample the corresponding fine-grained features and employ *MLP Layer* to refine their predictions, which are supervised by the ground truth and local hardness. For inference, we directly utilize *Trilinear Interpolation* to obtain final prediction.

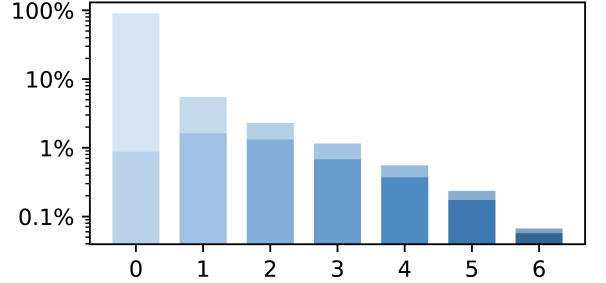global hardness $\mathcal{H}_{i,j,k}^{\text{global}}$ of this voxel is defined as follows

$$\mathcal{H}_{i,j,k}^{\text{global}} = \frac{1}{p^a - p^b}. \quad (1)$$

We obtain $\mathcal{H}^{\text{global}} \in \mathbb{R}^{X' \times Y' \times Z'}$ by computing all the predictions of voxels in $\mathcal{P}_{\text{coarse}}$. $\mathcal{H}^{\text{global}}$ measures the uncertainty of the semantic scene completion prediction between the class a and b, which varies with the optimization of the network. The value of $\mathcal{H}^{\text{global}}$ indicates the global hardness of predicting a certain voxel during the overall training process. In this paper, we mainly employ $\mathcal{H}^{\text{global}}$ to select hard voxels and refine their predictions.

Local Hardness. In 3D dense space, each voxel involves different geometric information, which depends on its various location. Since the voxels in the boundary region pose a greater challenge than those in the interior, the local information is crucial for instructing the model to find genuinely hard voxels. We adopt local geometric anisotropy [22] (LGA) $\mathcal{A} \in \mathbb{R}^{X \times Y \times Z}$ as the basis of the local hardness $\mathcal{H}^{\text{local}} \in \mathbb{R}^{X \times Y \times Z}$ on the selected voxels. For each voxel $v_{(i,j,k)}$ in $\mathcal{V}$, the LGA $\mathcal{A}_{i,j,k}$ is computed with its neighbors $\{v^1, v^2, .., v^M\}$ at M different directions

$$\mathcal{A}_{i,j,k} = \sum_{m=1}^{M} \left(v_{\text{gt}} \oplus v_{\text{gt}}^m \right), \quad (2)$$

where v_{gt} and v_{gt}^m are the semantic labels of v and v^m ($m = 1, ..., M$), respectively. $\oplus$ denotes the exclusive disjunction operation. It returns 0 or 1 when v and v^m have the same semantic label or not. Note that we calculate the LGA of all the voxels including empty ones. In our implementation, we set M to 6 and compute the voxels in the

Figure 4. Illustration of **Local Geometric Anisotropy (LGA)**. The upper figure gives the examples of different LGA values, which are all from real scenarios. The lower figure shows the distribution of LGA values in SemanticKITTI [1]. We use dark and light colors to represent the proportion of non-empty and empty voxels in each LGA value category, respectively.

up/down, front/back, and left/right directions. The examples and distribution information on SemanticKITTI [1] for different LGA values are provided in Fig. 4.

Then, the corresponding local hardness $\mathcal{H}_{i,j,k}$ is defined as below

$$\mathcal{H}_{i,j,k}^{\text{local}} = \alpha + \beta \mathcal{A}_{i,j,k}, \quad (3)$$

where α and β are the coefficients linearly mapping $\mathcal{A}$ onto $\mathcal{H}^{\text{local}}$. $\mathcal{H}^{\text{local}}$ measures the semantic difference between the selected voxel and its neighbors. The value of $\mathcal{H}^{\text{local}}$ reflects the geometric position on the object of the voxel. We adopt local hardness $\mathcal{H}^{\text{local}}$ to weight the selected voxels and make the model focus on more challenging voxel positions.

Hard Voxel Selection. During training, we select N hard voxels from the coarse prediction $\mathcal{P}_{\text{coarse}}$ with the global hardness $\mathcal{H}^{\text{global}}$. Since $N \ll X' \times Y' \times Z'$, selecting N voxels with the largest $\mathcal{H}^{\text{global}}$ in $\mathcal{P}_{\text{coarse}}$ directly may cause the SSC network to fall into over-fitting in the local area at the beginning. Motivated by PointRend [19], we firstly over-generate proposal voxels by randomly sampling tN voxels ($t > 1$) with a homogeneous distribution in 3D dense space. Secondly, ωN hard voxels ($\omega \in [0, 1]$) are selected from tN proposals by sorting the global hardness, which is calculated from the corresponding coarse prediction region $\mathcal{P}_{\text{coarse}}$. Thirdly, remaining $(1 - \omega)N$ voxels are randomly sampled from 3D space to prevent over-fitting in training. Finally, we obtain N hard voxels with the coordinates $\mathcal{V}^{\text{hard}} \in \mathbb{R}^{N \times 3}$.

Voxel-wise Refinement Module. Given N selected hard voxels, we re-sample their corresponding features from the fine-grained 3D volume features $\mathcal{F}_{\text{fine}}^{\text{3D}}$ by the coordinates $\mathcal{V}^{\text{hard}}$ and obtain $\mathcal{F}_{\text{fine}}^{\text{hard}} \in \mathbb{R}^{N \times D}$. Then, the hard voxels are refined with a lightweight network consisting of MLPs like

PointNet [35], since the voxel-wise prediction can be considered as point-wise segmentation problem as below

$$\mathcal{P}_{\text{refine}}^{\text{hard}} = \text{MLP}(\mathcal{F}_{\text{fine}}^{\text{hard}}), \qquad (4)$$

where $\mathcal{P}_{\text{refine}}^{\text{hard}}$ is the refined prediction of the selected hard voxels.

In training process, we need to re-sample the corresponding semantic labels of hard voxels from the ground truth. As the majority of 3D dense space is empty, we observe that the selected hard voxels are also predominantly empty ones at the beginning. In fact, part of these empty voxels are not inherently challenging to distinguish. If we treat the selected hard voxels equally, it may cause the SSC network to focus on some samples that are not actually difficult in the early stages of training. Therefore, we adopt the predefined local hardness $\mathcal{H}^{\text{local}}$ to weight the selected hard voxels and make the SSC network with the auxiliary MLP-based refinement head concentrate on the harder samples at local position. Specifically, the local hardness of selected hard voxels is computed by Eq. 2 and Eq. 3. Then, the hard voxel mining loss is calculated as follows

$$\mathcal{L}_{\text{s-hvm}} = \frac{1}{N} \sum_{n=1}^{N} \mathcal{H}_n^{\text{local}} \cdot \text{CE}(v_{\text{refine}}^n, v_{\text{gt}}^n), \qquad (5)$$

where $\text{CE}(\cdot, \cdot)$ denotes the cross entropy loss function. v_{refine}^n and v_{gt}^n are the refined prediction and the semantic label of the selected n-th hard voxel, respectively.

At inference stage, we directly use trilinear interpolation to obtain the final completion result $\mathcal{P}_{\text{final}}$ without introducing extra computational burden, as illustrated in Fig. 3.

3.3. Self-Distillation

To further train a robust model with higher performance without extra well-trained teacher model, we propose to perform self-distillation with the hard voxel mining head. The teacher model in our **HASSC** scheme has the same network architecture as the student model, as shown in Fig. 2. Following [13, 42], we construct a mean teacher by exponential moving average (EMA) to achieve better stability and consistency between iterations. During training process, the parameters of teacher model share the same initialization as student at step 0 and update at next steps as below

$$\theta_{t+1}^{\text{Teacher}} = \gamma \theta_t^{\text{Teacher}} + (1 - \gamma) \theta_{t+1}^{\text{Student}}, \qquad (6)$$

where $\gamma = \min\left(1 - \frac{1}{t+1}, 0.99\right)$. $\theta_t^{\text{Teacher}}$ and $\theta_t^{\text{Student}}$ are the learnable parameters of teacher and student model at step t. We only optimize parameters of the student model θ^{Student} and update the teacher network by Eq. 6.

Since the teacher network has the same hard voxel mining head as student, N hard voxels $\mathcal{V}^{\text{hard-T}} \in \mathbb{R}^{N \times 3}$ can be

obtained during training. We use $\mathcal{V}^{\text{hard-T}}$ to sample the final result $\mathcal{P}_{\text{final}}$ from the student branch. Then, the teacher-guided hard voxel mining loss $\mathcal{L}_{\text{t-hvm}}$ is computed by Eq. 5 with the corresponding local hardness, final prediction and ground truth. In case of the large number of voxels, we employ $\mathcal{L}_{\text{t-hvm}}$ to make the hard voxel selection by HVM head in student model more stable and consistent.

Moreover, we adopt Kullback–Leibler divergence ($\boldsymbol{D}_{\text{KL}}$) to instruct student model to learn from the online soft labels $\mathcal{P}_{\text{final}}^{\text{Teacher}}$ provided by teacher-branch as follows

$$\mathcal{L}_{\text{distill}} = \lambda e^{\mu} \cdot \boldsymbol{D}_{\text{KL}}\left(\mathcal{P}_{\text{final}}^{\text{Teacher}} \| \mathcal{P}_{\text{final}}\right), \qquad (7)$$

where λ is the weight coefficient for distillation. $\mu \in [0, 1]$ is the mean intersection over union (mIoU) value between the prediction of the current frame with teacher model $\mathcal{P}_{\text{fine}}^{\text{Teacher}}$ and the corresponding ground truth $\mathcal{V}_{\text{gt}}$.

3.4. Training and Inference

Overall Loss Function for Training. In this work, we follow the common settings [4, 29] and treat the semantic scene completion (SSC) as a voxel-wise classification problem. Overall, the total training loss of our proposed **HASSC** is composed of three terms as below

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{ssc}} + \mathcal{L}_{\text{hvm}} + \mathcal{L}_{\text{distill}}, \qquad (8)$$

where $\mathcal{L}_{\text{ssc}}$ is the commonly used loss for SSC. $\mathcal{L}_{\text{ssc}}$ consists of weighted cross entropy loss $\mathcal{L}_{\text{wce}}$ and scene-class affinity losses as follows

$$\mathcal{L}_{\text{ssc}} = \mathcal{L}_{\text{wce}} + \mathcal{L}_{\text{sem}} + \mathcal{L}_{\text{geo}}, \qquad (9)$$

where $\mathcal{L}_{\text{sem}}$ and $\mathcal{L}_{\text{geo}}$ are scene-class affinity losses optimized for semantics and geometry, respectively.

Additionally, the hard voxel mining loss $\mathcal{L}_{\text{hvm}}$ is made of $\mathcal{L}_{\text{s-hvm}}$ and $\mathcal{L}_{\text{t-hvm}}$:

$$\mathcal{L}_{\text{hvm}} = \mathcal{L}_{\text{s-hvm}} + \delta \cdot \mathcal{L}_{\text{t-hvm}}, \qquad (10)$$

where δ is the trade-off coefficient between student and teacher.

Inference. The student-branch is well optimized during training, which not only digs out hard samples and refine them with fine-grained features but also makes use of the soft labels provided by the teacher-branch. During the inference process, we only need to preserve the student-branch without incurring the extra computational cost.

4. Experiments

4.1. Setup

Dataset. The SemanticKITTI dataset [1] is the first large semantic scene completion benchmark for outdoor scenes, which contains LiDAR scans and front camera images from

Methods	VoxFormer-S [29]			HASSC VoxFormer-S			VoxFormer-T [29]			HASSC VoxFormer-T			StereoScene† [21]			HASSC StereoScene		
Modality	Camera			Camera			Camera			Camera			Camera			Camera		
Range	S	M	L	S	M	L	S	M	L	S	M	L	S	M	L	S	M	L
IoU (%)↑	65.35	57.54	44.02	**65.54**	**57.99**	**44.82**	65.38	57.69	44.15	**66.05**	**58.01**	**44.58**	65.70	56.84	43.66	65.52	**57.01**	**44.55**
mIoU (%)↑	17.66	16.48	12.35	**18.98**	**17.95**	**13.48**	21.55	18.42	13.35	**24.10**	**20.27**	**14.74**	23.27	21.15	15.24	**24.43**	**22.17**	**15.88**
■ car (3.92%)	39.78	35.24	25.79	**42.37**	**36.78**	**27.23**	44.90	37.46	26.54	**45.79**	**37.70**	**27.33**	47.05	43.52	31.15	46.47	43.02	30.64
■ bicycle (0.03%)	3.04	1.48	0.59	2.72	**2.26**	**0.92**	5.22	2.87	1.28	4.23	2.11	1.07	2.38	2.15	1.05	**4.20**	**2.63**	**1.20**
■ motorcycle (0.03%)	2.84	1.10	0.51	**4.49**	**1.63**	**0.86**	2.98	1.24	0.56	**5.64**	**2.03**	**1.14**	4.78	2.84	1.55	**5.26**	**3.34**	0.91
■ truck (0.16%)	7.50	7.47	5.63	6.25	**11.00**	**9.91**	9.80	10.38	7.26	**22.89**	**21.90**	**17.06**	18.72	22.48	17.55	**24.94**	**34.73**	**23.72**
■ other-veh. (0.20%)	8.71	4.98	3.77	**14.77**	**8.85**	**5.61**	17.21	10.61	7.81	**22.71**	**13.52**	**8.83**	17.33	13.79	9.26	**20.61**	**14.24**	7.77
■ person (0.07%)	4.10	3.31	1.78	**5.11**	**4.89**	**2.80**	4.44	3.50	1.93	**5.12**	**4.18**	**2.25**	6.31	4.37	2.17	6.06	3.58	1.79
■ bicyclist (0.07%)	6.82	7.14	3.32	**6.87**	**8.57**	**4.71**	2.65	3.92	1.97	**4.09**	**6.58**	**4.09**	7.70	4.75	2.30	**8.22**	**5.65**	**2.47**
■ motorcyclist (0.05%)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
■ road (15.30%)	72.40	65.74	54.76	**74.49**	**68.04**	**57.05**	75.45	66.15	53.57	**78.51**	**70.02**	**57.23**	79.24	74.16	61.86	**80.61**	**75.53**	**62.75**
■ parking (1.12%)	10.79	18.49	15.50	15.49	**21.23**	**15.90**	21.01	23.96	19.69	**29.43**	**26.69**	**19.89**	21.33	21.19	17.02	**25.21**	**25.95**	**20.20**
■ sidewalk (11.13%)	39.35	33.20	26.35	**42.69**	**36.32**	**28.25**	45.39	34.53	26.52	**51.69**	**38.83**	**29.08**	50.71	41.86	30.58	**52.68**	**43.61**	**32.40**
■ other-grnd (0.56%)	0.00	1.54	0.70	**0.02**	**2.38**	**1.04**	0.00	0.76	0.42	0.00	**1.55**	**1.26**	0.00	1.12	0.85	0.00	0.18	0.51
■ building (14.10%)	17.91	24.09	17.65	**22.78**	**27.30**	**19.05**	25.13	29.45	19.54	**27.99**	**30.81**	**20.19**	26.98	32.52	22.71	**29.09**	31.68	**22.90**
■ fence (3.90%)	12.98	10.63	7.64	9.81	8.70	6.58	16.17	11.15	7.31	**17.09**	**11.65**	**7.95**	22.50	14.26	8.73	20.88	13.32	8.67
■ vegetation (39.3%)	40.50	34.68	24.39	40.49	**35.53**	**25.48**	43.55	38.07	26.10	**44.68**	**38.93**	**27.01**	40.20	36.10	24.81	**40.29**	**36.44**	**26.27**
■ trunk (0.51%)	15.81	10.64	5.08	14.93	**11.25**	**6.15**	21.39	12.75	6.10	**22.22**	**14.11**	**7.71**	21.45	15.28	7.17	**21.65**	14.92	7.14
■ terrain (9.17%)	32.25	35.08	29.96	**36.66**	**38.28**	**32.94**	42.82	39.61	33.06	**47.04**	**41.37**	**33.95**	45.75	43.67	34.87	**48.50**	**46.95**	**38.10**
■ pole (0.29%)	14.47	11.95	7.11	**15.25**	**12.48**	**7.68**	20.66	15.56	9.15	18.95	14.76	**9.20**	20.43	18.95	10.66	18.67	16.34	9.00
■ traf.-sign (0.08%)	6.19	6.29	4.18	5.52	5.61	4.05	10.63	8.09	4.94	9.89	**8.44**	4.81	9.21	8.91	5.19	**10.88**	**9.08**	**5.23**

Table 1. Quantitative comparisons against the selected baseline methods on the **validation set** of SemanticKITTI [1]. † denotes the results are reproduced from the original implementation. "S", "M" and "L" represent the short range ($12.8 \times 12.8 \times 6.4m^3$), middle range ($25.6 \times 25.6 \times 6.4m^3$) and long/full range ($51.2 \times 51.2 \times 6.4m^3$), respectively. The improved results compared to the corresponding baselines are marked in **blue**.

KITTI Odometry Benchmark [11]. The ground truth is generated from the accumulated LiDAR semantic segmentation labels, which is represented as the $256 \times 256 \times 32$ voxel grids with a resolution of 0.2m. Each voxel grid is annotated as one of 19 semantic classes or 1 empty class. We adopt the same setting as in [1, 11] and split the total 22 sequences into (00-07, 09-10) / (08) / (11-21) for training/validation/test sets.

Evaluation Metrics. The mean intersection over union (mIoU) on 19 semantic classes is reported to evaluate the quality of semantic scene completion (SSC). Moreover, we adopt intersection over union (IoU) to measure the performance of class-agnostic scene completion (SC), which reflects the 3D geometric quality with 2D camera images as input. Besides, we calculate the IoU and mIoU at different ranges from the ego car on the validation set, which include the volume of $12.8 \times 12.8 \times 6.4m^3$ (short range, S), $25.6 \times 25.6 \times 6.4m^3$ (middle range, M), and $51.2 \times 51.2 \times 6.4m^3$ (long/full range, L). In practical, the perception results at closer range are more critical to vehicle safety.

Implementation Details. Our proposed **HASSC** method is designed as a generic training scheme to improve the performance of existing methods in hard regions. To demonstrate the efficacy of our approach, we choose the state-of-the-art methods including VoxFormer-S [29], VoxFormer-T [29][1]

and StereoScene [21][2] as our baseline models. VoxFormer-S only adopts the current frame from left camera as input while VoxFormer-T combines the previous 4 images. StereoScene uses both the left and right camera images to train the model. The input image size is set to 1220×370 and 1280×384 for VoxFormer and StereoScene, respectively. Other training settings are keep the same as the corresponding baselines. The number of selected hard voxels during training (N) is set to 4096. The coefficients (α, β) of linear transformation from $\mathcal{A}$ to $\mathcal{H}^{local}$ are set to 0.2 and 1.0, respectively. The distillation weight λ is set to 48. The trade-off coefficient δ is set to 0.1.

All the models in experiments are trained on four GeForce RTX 4090 GPUs with 24G memory, and the inference speed is reported by a single GeForce RTX 4090 GPU. More implementation details with different baselines are given in our supplementary material.

4.2. Performance

Qualitative Comparisons. We firstly present the quantitative comparison with the our baseline models on the validation set of SemanticKITTI. As shown in Tab. 1, **HASSC** effectively improves the accuracy over baseline methods including VoxFormer-S (+1.13%mIoU, +0.80%IoU), VoxFormer-T (+1.39%mIoU, +0.43%IoU) and Stere-

---
[1] https://github.com/NVlabs/VoxFormer
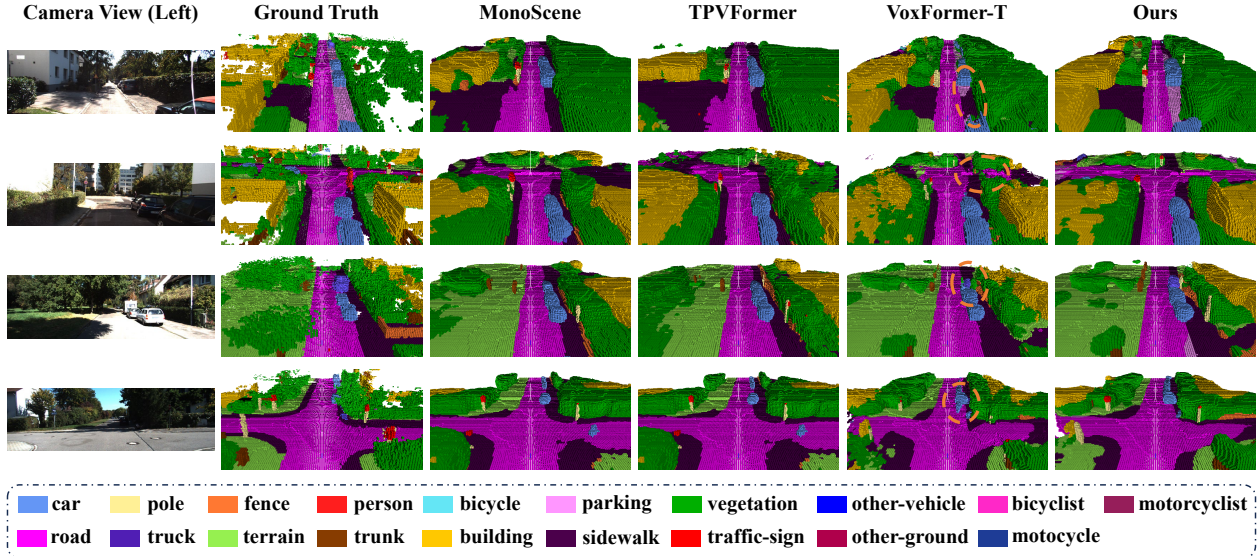
[2] https://github.com/Arlo0o/StereoScene

Figure 5. Visual results of our method (**HASSC**-VoxFormer-T) and the state-of-the-art camera-based methods on the validation set of SemanticKITTI. The left shows the perspective view image from left camera, which is the input for model training and inference. The right is the ground truth and the corresponding predicted semantic scene from these methods.

oScene (+0.64%mIoU, +0.89%IoU) at full range. Our **HASSC**-VoxFormer-T obtains a more obvious improvement at the closer range including short (+2.55%mIoU) and middle (+1.85%mIoU) to ensure the safety of autonomous vehicles. Besides, it is worthy of noting that **HASSC**-VoxFormer-S with single image input even outperforms VoxFormer-T with 5 images (13.48% *v.s.* 13.35%).

Then, we submit our prediction results to the website of SemanticKITTI for the online evaluation of hidden test set. In Tab. 2, we compare our approach against the state-of-the-art camera-based methods. **HASSC** shows consistent improvements with both VoxFormer-S (+1.14%mIoU) and VoxFormer-T (+0.97%mIoU). Note that **HASSC**-VoxFormer-S outperforms all the camera-based methods with single image input. Comparisons on detailed semantic categories and further discussions about our method are provided in the supplementary material.

Qualitative Comparisons. To further investigate the effectiveness of our proposed **HASSC**, we visualize the predictions of different models on the validation set of SemanticKITTI. As shown in Fig. 5, our method (**HASSC**-VoxFormer-T) performs better at complex classes junctions (*e.g.*, road, sidewalk and truck) compared to other camera-based approaches. This also corresponds to the improvements in quantitative evaluation (road +3.66%mIoU, sidewalk +2.56%mIoU, truck +9.80%mIoU) of our method, which demonstrates the effectiveness of the proposed hard voxel mining scheme.

4.3. Ablation Studies

In this section, we perform exhaustive ablation experiments on the validation set of SemanticKITTI with VoxFormer-

Methods	SSC Input	Pub.	IoU (%)↑	mIoU (%)↑
LMSCNet[*] [37]	$\hat{x}_{3D}^{occ}$	3DV 2020	31.38	7.07
3DSketch[*] [5]	$x^{rgb}, \hat{x}^{TSDF}$	CVPR 2020	26.85	6.23
AICNet[*] [23]	$x^{rgb}, \hat{x}^{depth}$	CVPR 2020	23.93	7.09
JS3C-Net[*] [49]	$\hat{x}^{pts}$	AAAI 2021	34.00	8.97
MonoScene [4]	x^{rgb}	CVPR 2022	34.16	11.08
TPVFormer [17]	x^{rgb}	CVPR 2023	34.25	11.26
OccFormer [56]	x^{rgb}	ICCV 2023	34.53	12.32
NDC-Scene [52]	x^{rgb}	ICCV 2023	36.19	12.58
VoxFormer-S [29]	x^{rgb}	CVPR 2023	42.95	12.20
VoxFormer-T [29]	$x^{rgb} \times 5$	CVPR 2023	43.21	13.41
HASSC-VoxFormer-S	x^{rgb}	-	**43.40**	13.34
HASSC-VoxFormer-T	$x^{rgb} \times 5$	-	42.87	**14.38**

Table 2. Quantitative comparisons with the state-of-the-art camera-based methods on the **hidden test set** of SemanticKITTI. ∗ denotes that the method is converted to the camera-based model by MonoScene [4].

T [29] as the baseline model for fair comparison.

Ablation on HASSC Scheme. Firstly, we provide the ablation of the proposed **HASSC** scheme. As illustrated in Tab. 3, the first row is the result reproduced with the original implementation of VoxFormer-T. It can be observed that using global hardness $\mathcal{H}^{global}$ and local hardness $\mathcal{H}^{local}$ individually obtains the limited performance improvements. Only the combination of $\mathcal{H}^{global}$ and $\mathcal{H}^{local}$ can effectively improve model performance. With the teacher-guided hard voxel mining (T-HVM), **HASSC** achieves stable improvements in both semantics and geometry. The self-distillation (T-Distill) from teacher-branch can provide consistent supervision with reliable soft labels and further improve the model accuracy when coupled with HVM head. Furthermore, we visualize the sum of the local hardness of N se-
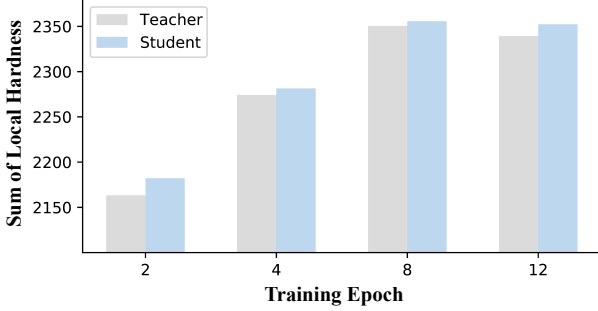
Figure 6. Visualization of the sum of the local hardness change during training on both student and teacher branches.

Global	Local	T-HVM	T-Distill	IoU (%)↑	mIoU (%)↑
				44.16	13.33
✓				43.89	13.30
	✓			44.00	13.40
✓	✓			43.98	13.91
✓	✓	✓		44.12	14.03
			✓	44.38	13.65
✓	✓	✓	✓	**44.58**	**14.74**

Table 3. Ablation study on our proposed **HASSC** scheme.

Methods	VoxFormer-T	HASSC-VoxFormer-T
Params (M)	57.91	58.43
Inference Speed (ms)	724.05	720.84
IoU (%)↑	44.16	**44.58**
mIoU (%)↑	13.33	**14.74**

Table 4. Comparison with baseline model on the training and inference efficiency.

Voxel Numbers (N)	0	1024	2048	4096	8192
IoU (%)↑	**44.16**	44.01	43.92	44.12	44.09
mIoU (%)↑	13.33	13.52	13.64	**14.03**	13.74

Table 5. Ablation study on the number of hard voxel selection.

Distill Weight (λ)	0	12	24	48	96
IoU (%)↑	44.12	44.06	44.25	**44.58**	44.51
mIoU (%)↑	14.03	13.80	14.23	**14.74**	14.39

Table 6. Ablation study on the weight of self-distillation from teacher model.

Methods	Hardness	IoU (%)↑	mIoU (%)↑
PALNet [22]	Local	44.28	13.28
PointRend [19]	Global	44.29	13.57
Xiao *et al.* [48]	Global	44.10	13.33
Ours	Global & Local	**44.58**	**14.74**

Table 7. Comparison with other hard sample mining schemes. We re-implement them with VoxFormer-T for fair comparison.

lected voxels during training. As shown in Fig. 6, the sum of local hardness continues to increase on both the student and teacher branches, which means the process of selecting voxels based on the learned global hardness $\mathcal{H}^{\text{global}}$ is consistent with local one $\mathcal{H}^{\text{local}}$.

Comparison of Training and Inference Efficiency. The model complexity analysis regarding training and inference is provided in Tab. 4. Compared with the vanilla VoxFormer-T, it can be seen that our proposed **HASSC** only introduces minimal overheads (+0.90% parameters) during training but promotes 10.58% relative performance without incurring extra cost for inference (724.05ms *v.s.* 720.84ms).

Ablation on Hard Voxel Selection. As there are totally $262,144$ ($128 \times 128 \times 16$) voxels in coarse prediction $\mathcal{P}_{\text{coarse}}$, we need to find an appropriate number (N) of hard voxels for refinement. Therefore, the ablation experiments on the number (N) of hard voxel selection are shown in Tab. 5. These experiments are conducted without distillation loss $\mathcal{L}_{\text{distill}}$ from teacher-branch. A small N yields an minimal improvement while a large one may include relatively easy voxels resulting in erroneous optimization. When N is set to 4096, the HVM head achieves the best performance.

Ablation on Self-Distillation. We present ablation on the weight (λ) of self-distillation loss $\mathcal{L}_{\text{distill}}$. As illustrated in

Tab. 6, inappropriate distillation loss weight may lead to the inferior model performance when $\lambda = 12$. We set λ to 48 in order to better integrate it into the hard voxel mining head.

Comparison with Other Schemes. Finally, we provide comprehensive comparison with existing hard sample mining schemes. We re-implement PALNet [22], PointRend [19] and [48] with VoxFormer-T [29] to facilitate a fair comparison. PALNet [22] is originally for indoor semantic scene completion and only considers local geometry. PointRend [19] and [48] are designed for 2D image segmentation and just use the globally updated information from the network optimization process. As shown in Tab. 7, they obtain marginal performance enhancements in 3D space of large-scale scenes. Our proposed **HASSC** outperforms all the reference methods.

5. Conclusion

In this paper, we adhere to the principle of *not all voxels are equal* and propose hardness-aware semantic scene completion (**HASSC**). The hard voxel mining head consists of hard voxel selection and voxel-wise refinement module, which combines global and local hardness to optimize the network on difficult regions. Additionally, self-distillation training strategy is introduced to improve the stability and consistency of completion. We have conducted extensive experiments to demonstrate that **HASSC** can effectively promote the existing semantic scene completion models without incurring the overheads during inference.

Acknowledgments

References

[1] 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*, pages 9297–9307, 2019. 1, 4, 5, 6

[2] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *SIGKDD*, pages 535–541, 2006. 2

[3] Yingjie Cai, Xuesong Chen, Chao Zhang, Kwan-Yee Lin, Xiaogang Wang, and Hongsheng Li. Semantic scene completion via integrating instances and scene in-the-loop. In *CVPR*, pages 324–333, 2021. 1

[4] Anh-Quan Cao and Raoul de Charette. MonoScene: Monocular 3D semantic scene completion. In *CVPR*, pages 3991–4001, 2022. 1, 2, 3, 5, 7

[5] 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*, pages 4193–4202, 2020. 7

[6] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qinhong Jiang, and Feng Zhao. Bevdistill: Cross-modal bev distillation for multi-view 3d object detection. In *ICLR*, 2022. 2

[7] Ran Cheng, Christopher Agia, Yuan Ren, Xinhai Li, and Liu Bingbing. S3CNet: A sparse semantic scene completion network for LiDAR point cloud. In *CoRL*, pages 2148–2161, 2021. 1, 2

[8] Hyeon Cho, Junyong Choi, Geonwoo Baek, and Wonjun Hwang. itkd: Interchange transfer-based knowledge distillation for 3d object detection. In *CVPR*, pages 13540–13549, 2023. 2

[9] Xueqing Deng, Peng Wang, Xiaochen Lian, and Shawn Newsam. Nightlab: A dual-level architecture with hardness detection for segmentation at night. In *CVPR*, pages 16938–16948, 2022. 2

[10] Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. In *ICML*, pages 1607–1616, 2018. 2

[11] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *CVPR*, pages 3354–3361, 2012. 6

[12] Adrian Hayler, Felix Wimbauer, Dominik Muhle, Christian Rupprecht, and Daniel Cremers. S4c: Self-supervised semantic scene completion with neural fields. In *3DV*, 2024. 2

[13] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9729–9738, 2020. 5

[14] Byeongho Heo, Jeesoo Kim, Sangdoo Yun, Hyojin Park, Nojun Kwak, and Jin Young Choi. A comprehensive overhaul of feature distillation. In *ICCV*, pages 1921–1930, 2019. 2

[15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 2

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

[17] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3D semantic occupancy prediction. In *CVPR*, pages 9223–9232, 2023. 1, 2, 7

[18] Mingi Ji, Seungjae Shin, Seunghyun Hwang, Gibeom Park, and Il-Chul Moon. Refine myself by teaching myself: Feature refinement via self-knowledge distillation. In *CVPR*, pages 10664–10673, 2021. 2

[19] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *CVPR*, pages 9799–9808, 2020. 2, 4, 8

[20] Shiyi Lan, Zhiding Yu, Christopher Choy, Subhashree Radhakrishnan, Guilin Liu, Yuke Zhu, Larry S Davis, and Anima Anandkumar. Discobox: Weakly supervised instance segmentation and semantic correspondence from box supervision. In *ICCV*, pages 3406–3416, 2021. 2

[21] Bohan Li, Yasheng Sun, Xin Jin, Wenjun Zeng, Zheng Zhu, Xiaoefeng Wang, Yunpeng Zhang, James Okae, Hang Xiao, and Dalong Du. Stereoscene: Bev-assisted stereo matching empowers 3d semantic scene completion. *arXiv preprint arXiv:2303.13959*, 2023. 1, 3, 6

[22] Jie Li, Yu Liu, Xia Yuan, Chunxia Zhao, Roland Siegwart, Ian Reid, and Cesar Cadena. Depth based semantic scene completion with position importance aware loss. *IEEE RA-L*, 5(1):219–226, 2019. 2, 4, 8

[23] Jie Li, Kai Han, Peng Wang, Yu Liu, and Xia Yuan. Anisotropic convolutional networks for 3d semantic scene completion. In *CVPR*, pages 3351–3359, 2020. 7

[24] Jiale Li, Hang Dai, and Yong Ding. Self-distillation for robust lidar semantic segmentation in autonomous driving. In *ECCV*, pages 659–676, 2022. 2

[25] Pengfei Li, Ruowen Zhao, Yongliang Shi, Hao Zhao, Jirui Yuan, Guyue Zhou, and Ya-Qin Zhang. Lode: Locally conditioned eikonal implicit scene completion from sparse lidar. In *ICRA*, 2023. 2

[26] Siqi Li, Changqing Zou, Yipeng Li, Xibin Zhao, and Yue Gao. Attention-based multi-modal fusion network for semantic scene completion. In *AAAI*, pages 11402–11409, 2020. 1

[27] Xiaoxiao Li, Ziwei Liu, Ping Luo, Chen Change Loy, and Xiaoou Tang. Not all pixels are equal: Difficulty-aware semantic segmentation via deep layer cascade. In *CVPR*, pages 3193–3202, 2017. 2

[28] 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. In *AAAI*, pages 1477–1485, 2023. 1

[29] 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, 2, 3, 5, 6, 7, 8

[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 *ECCV*, pages 1–18, 2022. 1, 3

[31] Zhiqi Li, Zhiding Yu, David Austin, Mingsheng Fang, Shiyi Lan, Jan Kautz, and Jose M Alvarez. Fb-occ: 3d occupancy prediction based on forward-backward view transformation. *arXiv preprint arXiv:2307.01492*, 2023. 1

[32] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pages 2980–2988, 2017. 2

[33] Shice Liu, Yu Hu, Yiming Zeng, Qiankun Tang, Beibei Jin, Yinhe Han, and Xiaowei Li. See and think: Disentangling semantic scene completion. In *NeurIPS*, 2018. 1

[34] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *ECCV*, pages 194–210, 2020. 3

[35] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, pages 652–660, 2017. 5

[36] Christoph B Rist, David Emmerichs, Markus Enzweiler, and Dariu M Gavrila. Semantic scene completion using local deep implicit functions on lidar data. *T-PAMI*, 44(10):7205–7218, 2021. 2

[37] Luis Roldao, Raoul de Charette, and Anne Verroust-Blondet. LMSCNet: Lightweight multiscale 3D semantic completion. In *3DV*, pages 111–119, 2020. 1, 2, 7

[38] Luis Roldao, Raoul De Charette, and Anne Verroust-Blondet. 3d semantic scene completion: A survey. *IJCV*, 130(8):1978–2005, 2022. 1

[39] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *CVPR*, pages 761–769, 2016. 2

[40] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *CVPR*, pages 1746–1754, 2017. 1

[41] Jiaxiang Tang, Xiaokang Chen, Jingbo Wang, and Gang Zeng. Not all voxels are equal: Semantic scene completion from the point-voxel perspective. In *AAAI*, pages 2352–2360, 2022. 1

[42] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, 2017. 5

[43] Song Wang, Jianke Zhu, and Ruixiang Zhang. Meta-rangeseg: Lidar sequence semantic segmentation using multiple feature aggregation. *IEEE RA-L*, 7(4):9739–9746, 2022. 1

[44] Song Wang, Wentong Li, Wenyu Liu, Xiaolu Liu, and Jianke Zhu. Lidar2map: In defense of lidar-based semantic map construction using online camera distillation. In *CVPR*, pages 5186–5195, 2023. 2

[45] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *CoRL*, pages 180–191, 2022. 3

[46] Joey Wilson, Jingyu Song, Yuewei Fu, Arthur Zhang, Andrew Capodieci, Paramsothy Jayakumar, Kira Barton, and Maani Ghaffari. Motionsc: Data set and network for real-time semantic mapping in dynamic environments. *IEEE RA-L*, 7(3):8439–8446, 2022. 2

[47] Zhaoyang Xia, Youquan Liu, Xin Li, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, and Yu Qiao. SCPNet: Semantic scene completion on point cloud. In *CVPR*, pages 17642–17651, 2023. 1, 2

[48] Xin Xiao, Daiguo Zhou, Jiagao Hu, Yi Hu, and Yongchao Xu. Not all pixels are equal: Learning pixel hardness for semantic segmentation. *arXiv preprint arXiv:2305.08462*, 2023. 2, 8

[49] 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*, pages 3101–3109, 2021. 1, 2, 7

[50] Jihan Yang, Shaoshuai Shi, Runyu Ding, Zhe Wang, and Xiaojuan Qi. Towards efficient 3d object detection with knowledge distillation. In *NeurIPS*, pages 21300–21313, 2022. 2

[51] Zhendong Yang, Zhe Li, Mingqi Shao, Dachuan Shi, Zehuan Yuan, and Chun Yuan. Masked generative distillation. In *ECCV*, pages 53–69, 2022. 2

[52] Jiawei Yao, Chuming Li, Keqiang Sun, Yingjie Cai, Hao Li, Wanli Ouyang, and Hongsheng Li. Ndc-scene: Boost monocular 3d semantic scene completion in normalized device coordinates space. In *ICCV*, pages 9455–9465, 2023. 7

[53] Jin Yin, Pengfei Xia, and Jingsong He. Online hard region mining for semantic segmentation. *Neural Processing Letters*, 50:2665–2679, 2019. 2

[54] Jia Zeng, Li Chen, Hanming Deng, Lewei Lu, Junchi Yan, Yu Qiao, and Hongyang Li. Distilling focal knowledge from imperfect expert for 3d object detection. In *CVPR*, pages 992–1001, 2023. 2

[55] Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *ICCV*, pages 3713–3722, 2019. 2

[56] Yunpeng Zhang, Zheng Zhu, and Dalong Du. OccFormer: Dual-path transformer for vision-based 3D semantic occupancy prediction. In *ICCV*, 2023. 2, 7

[57] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *CVPR*, pages 11953–11962, 2022. 2

[58] Shengchao Zhou, Weizhou Liu, Chen Hu, Shuchang Zhou, and Chao Ma. Unidistill: A universal cross-modality knowledge distillation framework for 3d object detection in bird's-eye view. In *CVPR*, pages 5116–5125, 2023. 2

[59] 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. 2

[60] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3D convolution networks for lidar segmentation. In *CVPR*, pages 9939–9948, 2021. 1