

S4C: Self-Supervised Semantic Scene Completion with Neural Fields

Adrian Hayler^{*,1} Felix Wimbauer^{*,1,2} Dominik Muhle^{1,2}
 Christian Rupprecht³ Daniel Cremers^{1,2,3}

¹Technical University of Munich ²MCML ³University of Oxford

{a.hayler, felix.wimbauer, dominik.muhle, cremers}@tum.de chrisr@robots.ox.ac.uk

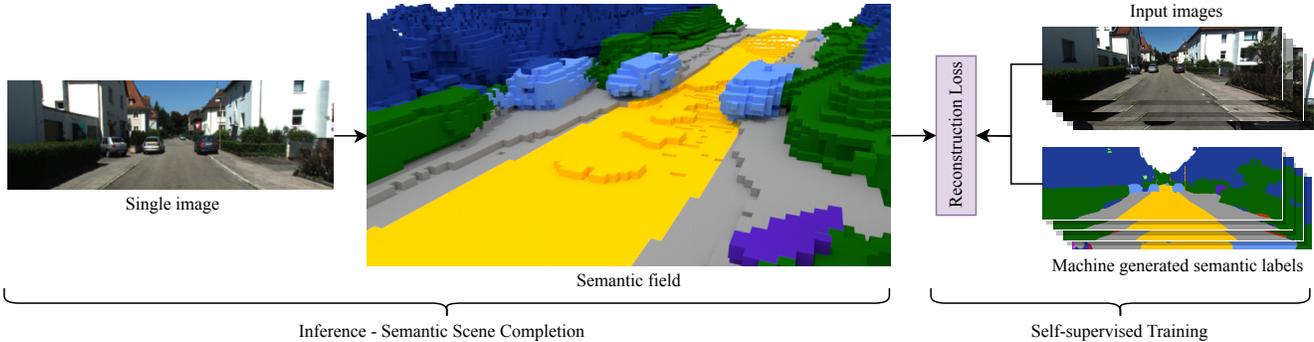


Figure 1. **S4C, predicting a semantic field from a single image.** S4C is the first fully self-supervised semantic scene completion (SSC) method that was trained on image data alone (and camera poses). Our method predicts a volumetric scene representation from a single image, capturing geometric and semantic information even in occluded regions. In contrast to previous SSC methods, we do not require any 3D ground truth information, allowing the use of image-only datasets for training. Despite the lack of ground truth data, our method is competitive to supervised methods with only small differences in performance. For further results, please check out the **video in the supplementary material**.

Abstract

3D semantic scene understanding is a fundamental challenge in computer vision. It enables mobile agents to autonomously plan and navigate arbitrary environments. SSC formalizes this challenge as jointly estimating dense geometry and semantic information from sparse observations of a scene. Current methods for SSC are generally trained on 3D ground truth based on aggregated LiDAR scans. This process relies on special sensors and annotation by hand which are costly and do not scale well. To overcome this issue, our work presents the first self-supervised approach to SSC called **S4C** that does not rely on 3D ground truth data. Our proposed method can reconstruct a scene from a single image and only relies on videos and pseudo segmentation ground truth generated from off-the-shelf image segmentation network during training. Unlike existing methods, which use discrete voxel grids, we represent scenes as implicit semantic fields. This formulation allows querying any point within the camera frustum for occupancy and semantic class. Our architecture is trained through rendering-based self-supervised losses. Nonetheless, our method achieves performance close to fully supervised state-of-the-art methods. Additionally, our method demonstrates strong

generalization capabilities and can synthesize accurate segmentation maps for far away viewpoints.

1. Introduction

A plethora of tasks require holistic 3D scene understanding. Obtaining an accurate and complete representation of the scene, both with regard to geometry and high-level semantic information, enables planning, navigation, and interaction. Obtaining this information is a field of active computer vision research that has become popular with the semantic scene completion (SSC) task [64]. SSC jointly infers the scene geometry and semantics in 3D space from limited observations [39].

Current approaches to SSC either operate on Lidar scans [60, 64] or image data [8, 39, 42, 51, 63] as input. Generally, these methods predict discrete voxel grids that contain occupancy and semantic class information. They are trained on 3D ground truth aggregated from numerous annotated Lidar scans. LiDAR-based methods perform better on the task of SSC but depend on costly sensors compared to cameras [38]. Cameras are readily available and offer a dense repre-

* equal contribution. [Project page](#)

sensation of the world. However, bridging the gap between 2D camera recordings and 3D voxel grids is not straightforward. MonoScene [8] uses line-of-sight projection to lift 2D features into 3D space. However, this disregards information for occluded and empty scene regions [39]. VoxFormer [39] uses a transformer network to simultaneously predict geometry and semantic labels starting from a few query proposals on a voxel grid.

Recently, neural fields have emerged as a versatile representation for 3D scenes. Here, an MLP learns a mapping from encoded coordinates to some output. While initially focused on geometry and appearance, they have since progressed to also incorporate semantic information [18, 31, 62, 70, 86]. One of the major drawbacks of neural fields is that they rely on test time optimization. The network weights are trained by reconstructing different input views of the scene via volume rendering. To enable generalization on scene geometry and appearance, some methods [75, 80] proposed to condition neural fields on pixel-aligned feature maps predicted by trainable image encoders.

In this work, we introduce the first self-supervised approach to semantic scene completion (SSC). Instead of predicting a voxel grid, our method infers a 3D semantic field from a single image. This field holds density and semantic information and allows for volume rendering of segmentation maps and color images (via image-based rendering). By applying reconstruction losses on the rendered output in 2D, we learn the geometry and semantics of the 3D scene. We train our approach on multiple views from the videos captured by a multi-cam setup mounted on a moving vehicle. To make our method as general as possible, we rely on segmentation maps generated by an off-the-shelf image segmentation network rather than hand-annotated ground truth. In order to learn geometry and semantics in the entire camera frustum, we sample frames from the videos at random time offsets to the input image. As the vehicle moves forward, the different cameras capture many areas of the scene, especially those that are occluded in the input image. Our formulation does not require any form of 3D supervision besides camera poses.

We use the KITTI-360 [43] dataset for training and measure the performance of our proposed approach on the new SSCBench [38] dataset, which is defined on top of KITTI-360. Both qualitative and quantitative results show that our method achieves comparable results to fully-supervised approaches for SSC. Further, we demonstrate the beneficial effects of our different loss components and the random frame sampling strategy. Finally, we also test the unique ability of our method to synthesize high-quality segmentation maps from novel views.

Our **contributions** can be summarized as follows:

- We propose the **first SSC training using self-supervision from images without the need for 3D**

ground truth data.

- We achieve close-to state-of-the-art performance compared to fully supervised SSC methods.
- Our method can synthesize high-quality segmentation maps from novel views.
- We release our code upon acceptance to further facilitate research into SSC.

2. Related Work

In the following, we will introduce relevant literature to our proposed method. For semantic scene completion (SSC), we will focus the discussion of related work on camera-based approaches and introduce LiDAR-based methods only briefly. We refer the interested reader to [61] for a broader overview of the topic of SSC.

2.1. Single Image Scene Reconstruction

Scene reconstruction refers to the task of estimating 3D geometry from images. While it has been a topic of active research for two decades [23], the introduction of NeRFs [52] has led to renewed interest in this area. An overview can be found in [22]. In the following, we restrict our review to single-view methods and their distinction from monocular depth estimation methods. Monocular depth estimation [19, 20, 46, 65, 87] reconstructs a 3D environment by predicting a per-pixel depth value. Ground-truth supervision [1, 15, 17, 33, 40, 41, 44, 74], reconstruction losses [19, 20, 82, 88], or a combination thereof [32, 79] have been used to train these networks. In contrast to depth estimation and its restriction to visible surfaces, scene reconstruction aims to also predict geometry in occluded regions. PixelNeRF [80], a NeRF variant with the ability to generalize to different scenes, can predict free space in occluded scene regions only from a single image. As an extension, SceneRF [9] uses a probabilistic sampling and a photometric reprojection loss to additionally supervise depth prediction to improve generalization. BTS [75] combines ideas from generalizable NeRF and self-supervised depth estimation and achieves very accurate geometry estimation, even for occluded regions. In contrast to our approach, the above-mentioned methods do not consider semantics and are therefore not suited for SSC. Another line of work for scene reconstruction leverages massive data to learn object shape priors [13, 16, 68, 77].

2.2. 3D Semantic Segmentation

Given a 3D model, such as mesh, semantic multi-view fusion models [3, 26, 30, 47, 50, 83] project segmentation masks from images into the 3D geometry. Implicit representations such as NeRFs have become popular for semantic 3D modeling [18, 31, 69, 70, 86] to ensure multi-view consistency of segmentation masks. [62] extends this idea

to the panoptic segmentation task. The works of ConceptFusion [28] and OpenScene [55] propose open vocabulary scene understanding by fusing open vocabulary representations into 3D scene representations allowing for segmentation as a downstream task.

In contrast to image segmentation, Lidar segmentation works with 3D data to assign semantic labels to point cloud data. Unlike images, point clouds are a sparse and unordered data collection. To address this different data modality, Lidar segmentation either use point-based [25, 57, 58, 72, 81], voxel-based [48, 56, 71], or projection-based methods [2, 5, 53, 76].

2.3. Semantic Scene Completion

Semantic scene completion (SSC) extends the task of completing the scene geometry by jointly predicting scene semantics. It was first introduced in [64] and has gained significant attention in recent years [61]. This contrasts the separate treatment of the tasks in early works [21, 67]. The first approaches on SSC either focused on indoor settings from image data [7, 10, 34–36, 45, 84, 85] or outdoor scenes with LiDAR-based methods [12, 37, 59, 60, 78]. MonoScene [8] was the first to present a unified camera approach to indoor and outdoor scenarios using a line-of-sight projection and a novel frustum proportion loss. VoxFormer [39] uses deformable cross-attention and self-attention on voxels from image features. OccDepth [51] utilizes stereo images and stereo depth supervision. Another line of work uses birds-eye-view and temporal information for 3D occupancy prediction [42, 63]. This idea was extended to a Tri-Perspective View in [27]. SSCNet first tackled the problem of SSC from an image and a depth map and used a 3D convolutional network to output occupancy and labels in a voxel grid [64]. LSMCNet combines 2D convolutions with multiple 3D segmentation heads at multiple resolutions to reduce network parameters [60]. Overall, Lidar methods tend to outperform camera approaches on outdoor scenes. All the above-mentioned methods require annotated 3D ground truth for training, which is costly to collect. The need for accurate 3D ground truth data restricts the evaluation of SSC methods to only a few datasets, such as SemanticKITTI [4]. SSCBench [38] is a recently introduced benchmark that includes annotated ground truth for SSC on KITTI-360 [43], nuScenes [6], and Waymo [66].

In contrast, we present **S4C**, a fully self-supervised training strategy that allows our model to be trained from posed images only, lifting the restriction on the expensive datasets with Lidar data.

3. Method

In the following, we describe our approach to predict the geometry and semantics of a scene from a single image \mathbf{I}_1 to tackle the task of Semantic Scene Completion, as shown

in Fig. 2. We first cover how we represent a scene as a continuous semantic field, and then propose a fully self-supervised training scheme that learns 3D geometry and semantics from 2D supervision only.

3.1. Notation

Let $\mathbf{I}_1 \in [0, 1]^{3 \times H \times W} = (\mathbb{R}^3)^\Omega$ be the input image which is defined on a lattice $\Omega = \{1, \dots, H\} \times \{1, \dots, W\}$. During training, we have access to $N = \{\mathbf{I}_1, \dots, \mathbf{I}_n\}$ additional views of the scene beside the input image. Through an off-the-shelf semantic segmentation network $\Phi(\mathbf{I})$, we obtain segmentation maps $\mathbf{L}_i \in \{0, \dots, c - 1\}^\Omega$ for all images $\mathbf{I} \in \{\mathbf{I}_1\} \cup N$. c denotes the number of different classes. Camera poses and projection matrix of the images are given as $T_i \in \mathbb{R}^{4 \times 4}$ and $K_i \in \mathbb{R}^{3 \times 4}$, respectively. Points in world coordinates are denoted as $\mathbf{x} \in \mathbb{R}^3$. Projection into the image plane of camera k is given by $\pi_k(\mathbf{x}) = K_k T_k \mathbf{x}$ in homogeneous coordinates.

3.2. Predicting a Semantic Field

Current approaches to SSC typically involve the prediction and manipulation of discrete voxel grids, which come with various limitations. Firstly, these grids have restricted resolution due to their cubic memory requirements and processing constraints. Second, when reconstructing a scene from an image, voxels are not aligned with the pixel space. Consequently, methods have to rely on complex multi-stage approaches to lift information from 2D to 3D [8, 39].

As an alternative, we propose a simple architecture that predicts an *implicit* and *continuous semantic field* to overcome these shortcomings. [86] A semantic field maps every point \mathbf{x} within the camera frustum to both volumetric density $\sigma \in [0, 1]$ and a semantic class prediction $l \in \{0, \dots, c - 1\}$. Inspired by [75, 80], we use a high capacity encoder-decoder network to predict a dense pixel-aligned feature map $\mathbf{F} \in (\mathbb{R}^3)^\Omega$ from the input image \mathbf{I}_1 . The feature $f_{\mathbf{u}}$ at pixel location \mathbf{u} describes the semantic and geometric structure of the scene captured along a ray through that pixel. We follow [75] and do not store color in the neural field. This improves generalization capabilities and robustness.

To query the semantic field at a specific 3D point $\mathbf{x} \in \mathbb{R}^3$ within the camera frustum, we first project \mathbf{x} onto the camera plane to obtain the corresponding pixel location \mathbf{u} . First, \mathbf{x} is projected onto the image plane $\mathbf{u}_1 = \pi_1(\mathbf{x})$. The corresponding feature vector is extracted from the feature map with bilinear interpolation $f_{\mathbf{u}} = \mathbf{F}(\mathbf{u})$. Together with positional encodings $\gamma(d)$ for distance d [52] and pixel position $\gamma(\mathbf{u}_1)$, the feature vectors is then decoded to the density

$$\sigma_{\mathbf{x}} = \phi_D(f_{\mathbf{u}_1}, \gamma(d), \gamma(\mathbf{u}_1)) \quad (1)$$

and semantic prediction

$$l_{\mathbf{x}} = \phi_S(f_{\mathbf{u}_1}, \gamma(d), \gamma(\mathbf{u}_1)). \quad (2)$$

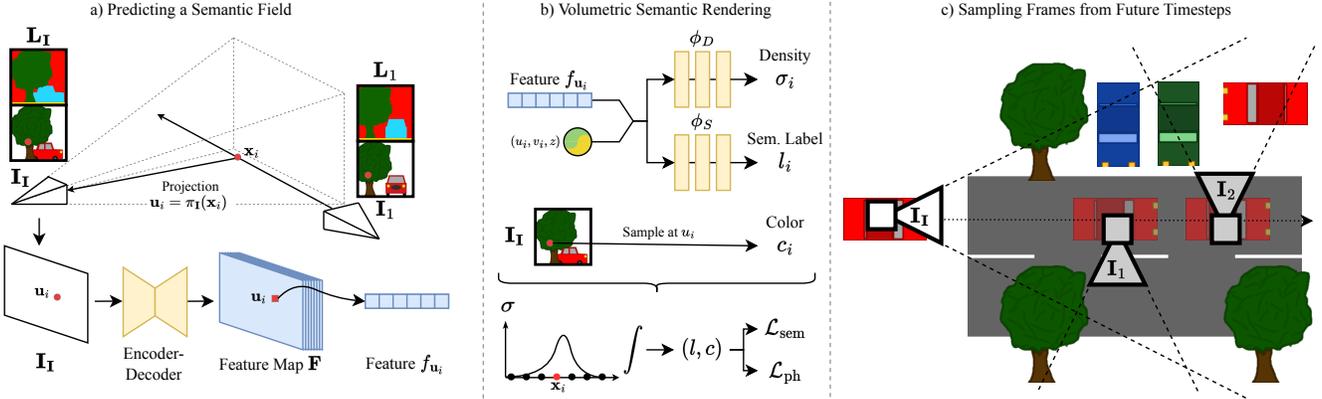


Figure 2. **Overview.** **a)** From an input image \mathbf{I}_I , an encoder-decoder network predicts a pixel-aligned feature map \mathbf{F} describing a semantic field in the frustum of the image. The feature f_{u_i} of pixel u_i encodes the semantic and occupancy distribution on the ray cast from the optical center through the pixel. **b)** The semantic field allows rendering novel views and their corresponding semantic segmentation via volumetric rendering. A 3D point \mathbf{x}_i is projected into the input image and therefore \mathbf{F} to sample f_{u_i} . Combined with positional encoding of \mathbf{x}_i , two MLPs decode the density of the point σ_i and semantic label l_i , respectively. The color c_i for novel view synthesis is obtained from other images via color sampling. **c)** To achieve best results, we require training views to cover as much surface of the scene as possible. Therefore, we sample side views from random future timesteps, that observe areas of the scene that are occluded in the input frame.

Both ϕ_D and ϕ_S are small multi-layer perceptron (MLP) networks. Note that ϕ_S predicts semantic logits. To obtain a class distribution or label, we apply $\text{Softmax}_c(l_{\mathbf{x}})$ or $\arg \max_c(l_{\mathbf{x}})$, respectively.

3.3. Volumetric Semantic Rendering

The goal of this paper is to develop a method to perform 3D SSC from a single image while being trained from 2D supervision alone. The continuous nature of our scene representation allows us to use volume rendering [29, 49] to synthesize high-quality novel views. As shown in [86], volumetric rendering can be extended from color to semantic class labels. The differentiable rendering process allows us to back-propagate a training signal from both color and semantic supervision on rendered views to our scene representation.

To render segmentation masks from novel viewpoints, we cast rays from the camera for every pixel. Along each ray, we integrate the semantic class labels over the probability of the ray ending at a certain distance. To approximate this integral, density $\sigma_{\mathbf{x}_i}$ and label $l_{\mathbf{x}_i}$ are evaluated at m discrete steps \mathbf{x}_i along the ray.

Since we consider segmentation in 3D space, we apply Softmax normalization at every query point individually, *before* we integrate along the ray. The intuition behind this is that regions with low density should not be able to “overpower” high-density regions by predicting very high scores for classes. Thus, this technique makes rendering semantics from the neural field more robust. [62]

$$\alpha_i = 1 - \exp(-\sigma_{\mathbf{x}_i} \delta_i) \quad T_i = \prod_{j=1}^{i-1} (1 - \alpha_j) \quad (3)$$

$$\hat{l} = \sum_{i=1}^m T_i \alpha_i \cdot \text{Softmax}_c(l_{\mathbf{x}_i}) \quad (4)$$

Here, δ_i denotes the distance between the points \mathbf{x}_i and \mathbf{x}_{i+1} , α_i the probability of the ray ending between the points \mathbf{x}_i and \mathbf{x}_{i+1} , and T_i the probability of \mathbf{x}_i being not occluded and therefore visible in the image. \hat{l} is the final, normalized distribution predicted for a pixel. We construct a per-pixel depth d_i , the distance between \mathbf{x}_i , and the ray origin using the expected ray termination depth.

$$\hat{d} = \sum_{i=1}^m T_i \alpha_i d_i \quad (5)$$

From the semantic field, we can also synthesize novel appearance views by applying image-based rendering [75]. Image-based rendering does not predict color information from a neural field but rather queries the color from other images. We sample the color by projecting point \mathbf{x}_i into frame k to the pixel position $\mathbf{u}_{i,k} = \pi_k(\mathbf{x}_i)$ and then bilinearly interpolating the color value $c_{i,k} = \mathbf{I}_k(\mathbf{u}_{i,k})$ at the projected pixel position. The color sample can come from any other image k except the one we want to render. With volumetric rendering, we obtain the pixel color with queries from image k as

$$\hat{c}_k = \sum_{i=1}^m T_i \alpha_i c_{i,k}, \quad (6)$$

where we obtain a different color prediction from every frame k .

3.4. Training for Semantic Multi-View Consistency

All existing methods for SSC rely on 3D ground truth data for training. Such data is generally obtained from annotated LiDAR scans, which are very difficult and costly to acquire. In contrast to 3D data, images with semantic labels are abundantly available. We propose to leverage this available 2D data to train a neural network for 3D semantic scene completion. To make our approach as general as possible, we use pseudo-semantic labels generated from a pre-trained semantic segmentation network. This allows us to train our architecture only from posed images without the need for either 2D or 3D ground truth data in a fully self-supervised manner.

In this paper, we consider SSC in an autonomous driving setting. Generally, there are forward and sideways-facing cameras mounted on a car that is moving. We train our method on multiple posed images. In addition to the source image \mathbf{I} from which a network produces the feature map \mathbf{F} , frames \mathbf{I}_i are aggregated from the main camera, stereo, and side-view cameras over multiple timesteps of a video sequence.

We randomly sample a subset of the available frames and use them as reconstruction targets for novel view synthesis. Our pipeline reconstructs both colors and semantic labels - based on our semantic field and color samples from other frames. The discrepancy between the pseudo-ground truth semantic masks and the image is used as the training signal.

As supervision from a view only gives training signals in areas that are observed by this camera, it is important to select training views strategically. Especially sideways-facing camera views give important cues for regions that are occluded in the input image. In order to best cover the entire camera frustum we aim to reconstruct, we, therefore, select sideways-facing views with a random offset to the input image for training. This increases the diversity and improves prediction quality especially for further away regions, which are often difficult to learn with image-based reconstruction methods.

In practice, we only reconstruct randomly sampled patches P_i that we reconstruct from different render frames k as $\hat{P}_{i,k}$ for color. For semantic supervision, we reconstruct the semantic labels S_i as \hat{S}_i . To train for SSC, *i.e.* scene reconstruction and semantic labelling, we use a combination of semantic and photometric reconstruction loss. While the photometric loss gives strong training signals for the general scene geometry, the semantic loss is important for differentiating objects and learning rough geometry. Furthermore, it guides the model to learn sharper edges around objects.

We use weighted binary cross-entropy loss to apply supervision on the density field from our pseudo ground truth

segmentation labels \mathcal{L}_i .

$$\mathcal{L}_{\text{sem}} = \text{BCE} \left(S_i, \hat{S}_i \right) \quad (7)$$

The photometric loss employs a combination of L1, SSIM [73] and an edge-aware smoothness (eas) term loss as proposed in [20]. We follow the strategy of [20, 75] to take only the per-pixel *minimum* into account when aggregating the costs.

$$\mathcal{L}_{\text{ph}} = \min_{k \in N_{\text{render}}} \left(\lambda_{\text{L1}} \text{L1}(P_i, \hat{P}_{i,k}) + \lambda_{\text{SSIM}} \text{SSIM}(P_i, \hat{P}_{i,k}) \right) \quad (8)$$

$$\mathcal{L}_{\text{eas}} = |\delta_x d_i^*| e^{-|\delta_x P_i|} + |\delta_y d_i^*| e^{-|\delta_y P_i|} \quad (9)$$

Our final loss is then computed as a weighted sum:

$$\mathcal{L} = \mathcal{L}_{\text{sem}} + \lambda_{\text{ph}} \mathcal{L}_{\text{ph}} + \lambda_{\text{eas}} \mathcal{L}_{\text{eas}} \quad (10)$$

4. Experiments

To demonstrate the capabilities of our proposed method, we conduct a wide range of experiments. First, we evaluate our method using the new SSCBench dataset [38] on KITTI-360 [43] and achieve performance that closely rivals state-of-the-art, but supervised, methods. We also conduct ablation studies to justify our design choices and to demonstrate synergistic effects between semantic and geometric training. Second, we show that our method can synthesize high-quality segmentation maps for novel viewpoints.

4.1. Implementation Details

For the architecture, we rely on a ResNet-50 [24] pre-trained on ImageNet as the backbone and a prediction head based on [20]. The feature vectors $f_{\mathbf{u}}$ have a dimension of 64. Both MLPs ϕ_D and ϕ_S are very lightweight with two fully-connected layers of 64 hidden nodes each. They do not need more capacity, as all information is already contained in the feature vector $f_{\mathbf{u}}$ and the MLPs are tasked to decode the contained information at a certain distance.

We implement our architecture entirely on PyTorch [54] and train with a batch size of 16 on a single Nvidia A40 GPU with 48GB memory. For every input image, we sample 32 patches of size 8×8 for RGB and semantics reconstruction each. For further technical details, please refer to the supplementary material.

4.2. Data

For training and testing our method, we use the established KITTI-360 [43] dataset, which consists of video sequences captured by multiple cameras mounted on top of a moving vehicle. Besides a pair of forward-facing stereo cameras, KITTI-360 provides recordings from two fisheye cameras facing sideways left and right. The fisheye cameras allow us to gather geometric and semantic information in parts of

the scene occluded in the source view. We are interested in an area approximately 50 meters in front of the car. Based on this and considering an average speed of 30-50kph, we sample fisheye views between 1 to 4 seconds into the future. The recordings in KITTI-360 have a frame rate of 10Hz which translates to an offset of 10 - 40 timesteps. In total, we use a total of eight images per sample during training: Four forward-facing views (out of which one is the input image), and four sideways-facing views.

To generate the pseudo segmentation labels, we run the off-the-shelf segmentation network Panoptic-Deeplab [11] trained on Cityscapes [14].

4.3. SSC Performance on SSCBench

To compare the performance of our method with supervised approaches, we evaluate the predicted semantic fields on the new SSCBench dataset [38] which is defined on KITTI-360 [43]. This dataset aggregates Lidar scans over entire driving sequences and builds voxel grids with occupancy and semantic information. These voxel grids can be used to compute both occupancy (geometry) and semantic-focused performance metrics. SSCBench follows the setup of SemanticKITTI [4] to use a voxel resolution of 0.2m on scenes of size $51.2m \times 51.2m \times 6.4m$, $25.6m \times 25.6m \times 6.4m$, and $12.8m \times 12.8m \times 6.4m$. Evaluation of SSCBench happens at three scales of $12.8m \times 12.8m \times 6.4m$, $25.6m \times 25.6m \times 6.4m$, and $51.2m \times 51.2m \times 6.4m$. SSCBench also provides invalid masks for voxels that do not belong to the scene. As these are quite coarse, we use slightly refined invalid masks. For fairness, we rerun all related methods on this refined data and observe a minor performance increase for all approaches. For further details, please refer to the supplementary material.

It is important to note that evaluating on SSCBench means bridging a non-trivial domain gap for our method. Our approach is trained via 2D supervision, which means that scene geometry must be pixel-level accurate. Occupancy is predicted for areas *within* objects. On the other hand, SSCBench ground truth was collected by aggregating Lidar point clouds. Here, a voxel is considered occupied when a Lidar measurement point lies within the voxel, but Lidar measurement points only capture the *surface* of an object. Therefore, the voxel ground truth of SSCBench tends to grow objects in size.

We use two techniques during the discretization of our implicit semantic field into a voxel grid to best align our predictions with the way the ground truth was captured. First, we leverage the unique advantage of our architecture to be not bound to the coarse resolution of discrete voxel grids. For every voxel, we query *multiple* different points distributed evenly within the voxel. The final prediction is obtained by taking the maximum occupancy value among the points and weighting the class predictions accordingly.

The intuition behind this is that we want to check whether there is a surface anywhere in the voxel. However, we observe that volumetric rendering encourages the occupancy to blur at object borders. In a second step, we, therefore, perform a neighbourhood check. A voxel is considered occupied when volumetric occupancy is observed in at least one of the immediate neighbouring voxels.

Our method is the first to approach SSC in a self-supervised manner without relying on 3D ground truth. We compare against several fully supervised, state-of-the-art approaches, namely MonoScene [8], LMSCNet [60], and SSCNet [64]. While MonoScene takes single images as input at test time, LMSCNet and SSCNet require Lidar scans even at test time.

As can be seen in Tab. 1, even though we tackle a significantly more challenging task, our proposed **S4C** method achieves occupancy IoU and segmentation mean IoU results that closely rival these of MonoScene [8], which is trained with annotated 3D Lidar ground truth. Furthermore, the results are not far off from the methods that take Lidar inputs at test time. Even though further away regions (25.6m and 51.2m) are usually more challenging for self-supervised approaches, as fewer camera views observe them, the performance of our method stays stable even when evaluating further away distances. We attribute this to the strategy of sampling side views at a random time offset. While the occupancy precision of our method is lower than other methods, our occupancy recall is significantly higher. A possible reason for this is that our method tends to place occupied voxels in areas that are unobserved and that cannot be hallucinated in a meaningful way from the observations in the image. Methods trained on inherently more sparse voxel ground truth tend to place unoccupied voxels in such regions.

When visualizing the predicted voxel grids, as shown in Fig. 3, we can see that our method is able to accurately reconstruct the given scene and to assign correct class labels. The general structure of the scene is recovered at a large scale and smaller objects like cars are clearly separated from the rest of the scene. Even for further distances, which are more difficult for camera-based methods, our method still provides reasonable reconstructions. As mentioned before, our method can be observed to place more occupied in unseen, ambiguous regions. Generally, the qualitative difference between the reconstructions by our method and supervised approaches is very low. This suggests that self-supervised training is a viable alternative to costly fully-supervised approaches to SSC.

4.4. Ablation studies

To give insights into the effect of our design choices, we conduct thorough ablation studies using the SSCBench ground truth.

Method	S4C (Ours)			MonoScene [8]			LMSCNet [60]			SSCNet [64]		
Supervision	Camera only			Lidar training			Lidar training + testing					
Range	12.8m	25.6m	51.2m	12.8m	25.6m	51.2m	12.8m	25.6m	51.2m	12.8m	25.6m	51.2m
IoU	54.64	45.57	39.35	58.61	48.15	40.66	66.74	58.48	47.93	74.93	66.36	55.81
Precision	59.75	50.34	43.59	71.79	67.02	64.79	80.58	76.75	76.87	83.65	77.85	75.41
Recall	86.47	82.79	80.16	76.15	63.11	52.20	79.54	71.07	56.00	87.79	81.80	68.22
mIoU	16.94	13.94	10.19	20.44	16.42	12.34	22.01	19.81	15.36	26.64	24.33	19.23
car	22.58	18.64	11.49	36.05	29.19	20.87	39.6	32.48	20.63	52.72	45.93	31.89
bicycle	0.00	0.00	0.00	2.69	1.07	0.49	0.00	0.00	0.00	0.00	0.00	0.00
motorcycle	0.00	0.00	0.00	4.70	1.44	0.59	0.00	0.00	0.00	1.41	0.41	0.19
truck	7.51	4.37	2.12	19.81	14.14	8.48	0.62	0.44	0.23	16.91	14.91	10.78
other-vehicle	0.00	0.01	0.06	8.81	5.61	2.78	0.00	0.00	0.00	1.45	1.00	0.60
person	0.00	0.00	0.00	2.26	1.30	0.87	0.00	0.00	0.00	0.36	0.16	0.09
road	69.38	61.46	48.23	82.94	73.32	58.23	84.60	81.24	69.06	87.81	85.42	73.82
sidewalk	45.03	37.12	28.45	56.51	43.53	32.70	60.73	51.28	36.71	67.19	60.34	46.96
building	26.34	28.48	21.36	39.17	38.02	31.79	48.59	51.55	41.22	53.93	54.55	44.67
fence	9.70	6.37	3.64	12.36	6.70	3.83	1.64	0.62	0.26	14.39	10.73	6.42
vegetation	35.78	28.04	21.43	38.26	31.51	25.67	51.17	46.90	38.70	56.66	51.77	43.30
terrain	35.03	22.88	15.08	38.05	27.30	19.29	43.23	32.59	23.54	43.47	36.44	27.83
pole	1.23	0.94	0.65	10.41	9.25	7.34	0.00	0.00	0.00	1.03	1.05	0.62
traffic-sign	1.57	0.83	0.36	9.20	7.98	5.68	0.00	0.00	0.00	1.01	1.22	0.70
other-object	0.00	0.00	0.00	6.62	5.17	3.44	0.00	0.00	0.00	1.20	0.97	0.58

Table 1. **Quantitative evaluation on SSCBench-KITTI-360.** We report the performances with respect to different ranges (12.8m, 25.6m, and 51.2m). We provide both **geometric** (IoU, Precision, Recall) and **semantic** (mIoU, per class IoU) metrics. As we use refined invalid masks on SSCBench, we rerun all methods with the provided checkpoints. MonoScene [8] is trained with Lidar but also uses a single image at test time. LMSCNet [60] and SSCNet [64] are trained on Lidar data and require a sparse Lidar scan at test time.

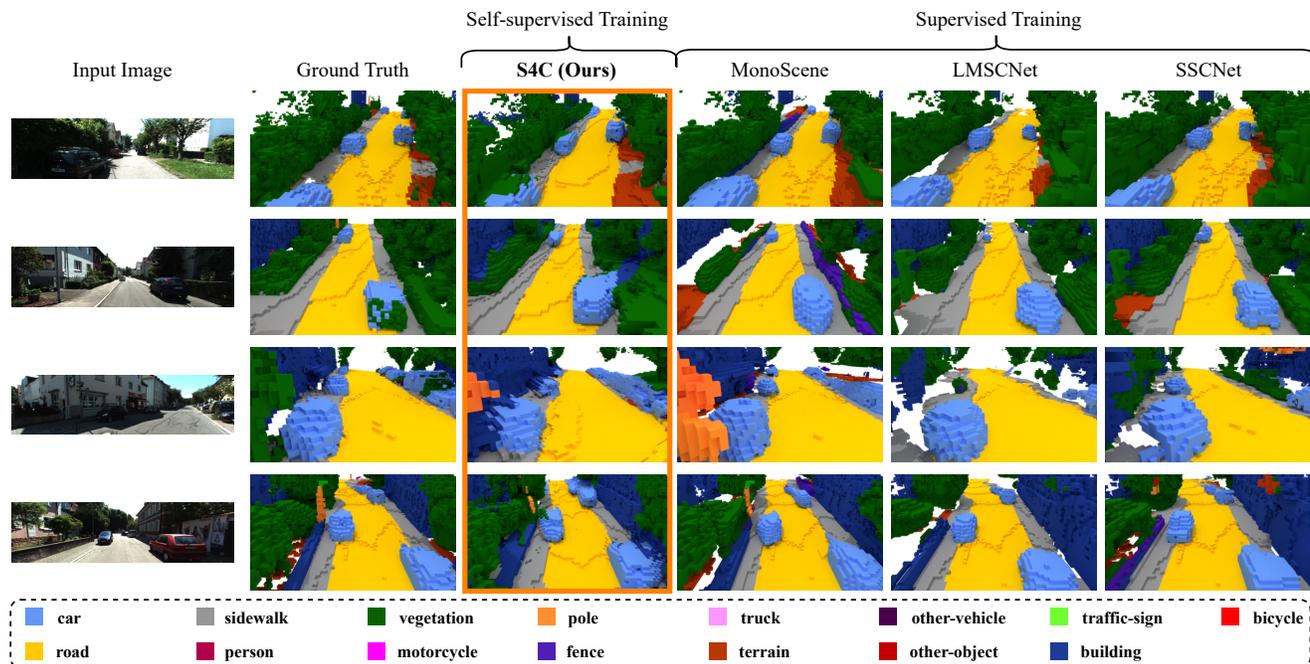


Figure 3. **Predicted voxel grids for SSCBench-KITTI-360.** The qualitative evaluation of our method on occupancy maps shows that our method is able to accurately reconstruct and label the scene. Especially a comparison to other image based methods like MonoScene shows, that **S4C** is able to recover details such as the driveway on the right in image 1. The resulting voxel occupancy from **S4C** shows fewer holes than for Lidar based training, which reproduce holes found in the ground truth.

First, to investigate the effect of the different 2D supervision signals, we train our architecture with the different loss terms turned off and report the results in Tab. 2. The photometric loss alone can already give a clear training signal and allows the network to recover accurate scene geometry.

Despite the semantic loss being very high-level and not providing a signal for smaller geometric details, it is enough to guide the network to predict rough geometry correctly. As shown in Fig. 4, the model can synthesize depth maps that clearly depict a geometric understanding of the scene.

Semantic	Photometric	View Offset	12.8m	25.6m	51.2m
✓	✗	1s-4s	31.11	32.04	27.88
✗	✓	1s-4s	50.26	41.94	37.40
✓	✓	1s	52.00	41.96	36.67
✓	✓	1s-4s	54.64	45.57	39.35

Table 2. **Ablation studies on SSCBench-KITTI-360.** We report IoU for occupancy. The full model achieves the best performance.

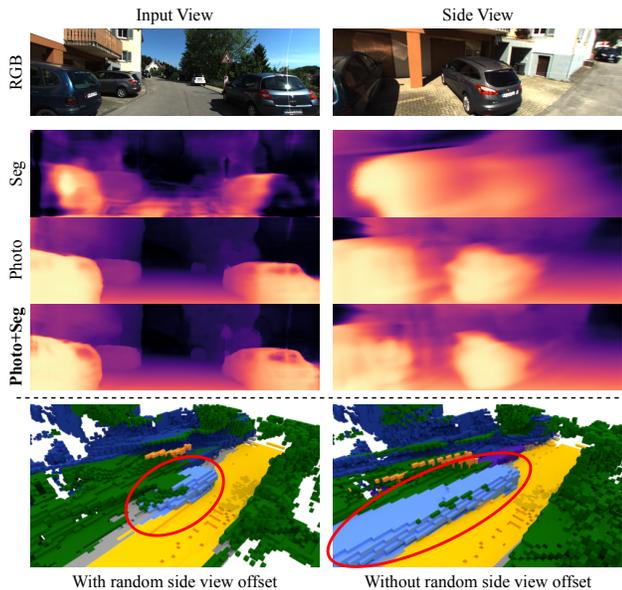


Figure 4. **Effects of different loss terms.** We show expected ray termination depth for the input image and a corresponding side view for different loss configurations. The full model produces sharpest results. We also show reconstruction with and without random offsetting the side views. This technique helps to correctly capture objects that are further away and reduce trailing effects.

Best occupancy results are achieved when training relies on both losses. We hypothesize that the object boundaries in segmentation maps, which are much clearer compared to regular images, help the model to learn sharper geometry.

Second, we investigate the impact of our view sampling strategy. When sampling sideways-facing views only at a fixed distance (1s into the future), the model is not able to learn about far-away geometry. Therefore, the performance is weaker especially when evaluating larger scenes (25.6m and 51.2m). This effect can also be observed qualitatively in Fig. 4.

4.5. Synthesizing Segmentation Maps

Besides experiments for SSC, we also analyse the effectiveness of our training with pseudo-ground truth labels from an off-the-shelf segmentation network. To this end, we train our model with different levels of pseudo supervision, by providing pseudo segmentation maps for the input frame (one image), all forward-facing views (four images), and all forward and sideways-facing views (eight images) re-

Training Configuration	KITTI GT	+5	+10	+15
Only Input Frame	86.50%	82.64%	77.74%	73.65%
Only Front	86.76%	83.09%	77.73%	73.15%
Full	87.81%	84.88%	81.07%	77.19%

Table 3. **Accuracy of synthesized segmentation maps.** Segmentation maps from **S4C** produce accurate results when compared to the ground truth. Predicting segmentation maps for other frames (5, 10, and 15 timesteps in the future) shows the geometric accuracy of **S4C**.

spectively.

We evaluate the segmentation performance for the input image against the real ground-truth segmentation provided by KITTI-360. Furthermore, we project segmentation maps 5, 10 and 15 time steps into the future using our predicted geometry. This tests the model’s ability to reason about segmentation in 3D. We report results in Tab. 3.

Providing segmentation masks for more frames during training improves the segmentation performance in all settings. Given pseudo ground truth for all frames, our model is even able to improve over the pseudo segmentation ground truth it was trained with, which achieves an accuracy of 87.7% against the KITTI-360 ground truth. This trend manifests when synthesizing novel segmentation views a number of timesteps away, where the discrepancy between the best and worst model configuration rises from 1.3% to over 3.4%.

We hypothesize that this is due to two reasons: First, the pseudo-segmentation ground truth is imperfect and contains artefacts. By forcing the model to satisfy segmentations from different views, the model automatically learns cleaner segmentation masks with fewer artefacts. Second, by having more training views, we improve the 3D geometry, as shown in Sec. 4.4, and are able to learn better semantic labels in occluded regions. The further away we synthesize views, the more important such 3D semantic understanding is for accurate predictions.

5. Conclusion

This work presents **S4C**, a novel, image-based approach to semantic scene completion. It allows reconstructing both the 3D geometric structure and the semantic information of a scene from a single image. Although not trained on 3D ground truth, it achieves close to state-of-the-art performance on the KITTI-360 dataset within the SSCBench benchmark suite. As the first work on self-supervised semantic scene completion, **S4C** opens up the path towards scalable and cheap holistic 3D understanding.

Acknowledgements. This work was supported by the ERC Advanced Grant SIMULACRON, by the Munich Center for Machine Learning and by the EPSRC Programme Grant VisualAI EP/T028572/1. C. R. is supported by VisualAI EP/T028572/1 and ERC-UNION-CoG-101001212.

References

- [1] Shubhra Aich, Jean Marie Uwabeza Vianney, Md Amirul Islam, and Mannat Kaur Bingbing Liu. Bidirectional attention network for monocular depth estimation. In *IEEE International Conference on Robotics and Automation*, 2021. 2
- [2] Yara Ali Alnaggar, Mohamed Afifi, Karim Amer, and Mohamed ElHelw. Multi projection fusion for real-time semantic segmentation of 3d lidar point clouds. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2021. 3
- [3] Iro Armeni, Zhi-Yang He, JunYoung Gwak, Amir R Zamir, Martin Fischer, Jitendra Malik, and Silvio Savarese. 3d scene graph: A structure for unified semantics, 3d space, and camera. In *CVPR*, 2019. 2
- [4] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9297–9307, 2019. 3, 6
- [5] Jorge Beltrán, Carlos Guindel, Francisco Miguel Moreno, Daniel Cruzado, Fernando Garcia, and Arturo De La Escalera. Birdnet: a 3d object detection framework from lidar information. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018. 3
- [6] 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 multi-modal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 3
- [7] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 324–333, 2021. 3
- [8] Anh-Quan Cao and Raoul de Charette. Monoscene: Monocular 3d semantic scene completion. In *CVPR*, 2022. 1, 2, 3, 6, 7
- [9] Anh-Quan Cao and Raoul de Charette. Scenerf: Self-supervised monocular 3d scene reconstruction with radiance fields. In *ICCV*, 2023. 2
- [10] Xiaokang Chen, Kwan-Yee Lin, Chen Qian, Gang Zeng, and Hongsheng Li. 3d sketch-aware semantic scene completion via semi-supervised structure prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4193–4202, 2020. 3
- [11] Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12475–12485, 2020. 6
- [12] Ran Cheng, Christopher Agia, Yuan Ren, Xinhai Li, and Liu Bingbing. S3cnet: A sparse semantic scene completion network for lidar point clouds. In *Conference on Robot Learning*, pages 2148–2161. PMLR, 2021. 3
- [13] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *European conference on computer vision*, pages 628–644. Springer, 2016. 2
- [14] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 6
- [15] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *NeurIPS*, 27, 2014. 2
- [16] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 605–613, 2017. 2
- [17] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *CVPR*, 2018. 2
- [18] Xiao Fu, Shangzhan Zhang, Tianrun Chen, Yichong Lu, Lanyun Zhu, Xiaowei Zhou, Andreas Geiger, and Yiyi Liao. Panoptic nerf: 3d-to-2d label transfer for panoptic urban scene segmentation. In *International Conference on 3D Vision (3DV)*, 2022. 2
- [19] Clément Godard, Oisín Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *ICCV*, pages 270–279, 2017. 2
- [20] Clément Godard, Oisín Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *ICCV*, pages 3828–3838, 2019. 2, 5
- [21] Saurabh Gupta, Pablo Arbelaez, and Jitendra Malik. Perceptual organization and recognition of indoor scenes from rgb-d images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 564–571, 2013. 3
- [22] Xian-Feng Han, Hamid Laga, and Mohammed Bannamoun. Image-based 3d object reconstruction: State-of-the-art and trends in the deep learning era. *IEEE TPAMI*, page 1578–1604, 2021. 2
- [23] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 2
- [24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5
- [25] Tong He, Haibin Huang, Li Yi, Yuqian Zhou, Chihao Wu, Jue Wang, and Stefano Soatto. Geonet: Deep geodesic networks for point cloud analysis. In *CVPR*, 2019. 3
- [26] Alexander Hermans, Georgios Floros, and Bastian Leibe. Dense 3d semantic mapping of indoor scenes from rgb-d images. In *IEEE International Conference on Robotics and Automation*, 2014. 2
- [27] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d

- semantic occupancy prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9223–9232, 2023. 3
- [28] Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Shuang Li, Ganesh Iyer, Soroush Saryazdi, Nikhil Keetha, Ayush Tewari, Joshua B. Tenenbaum, Celso Miguel de Melo, Madhava Krishna, Liam Paull, Florian Shkurti, and Antonio Torralba. Conceptfusion: Open-set multimodal 3d mapping. *arXiv*, 2023. 3
- [29] James T Kajiya and Brian P Von Herzen. Ray tracing volume densities. *ACM SIGGRAPH computer graphics*, 18(3):165–174, 1984. 4
- [30] Abhijit Kundu, Xiaoqi Yin, Alireza Fathi, David Ross, Brian Brewington, Thomas Funkhouser, and Caroline Pantofaru. Virtual multi-view fusion for 3d semantic segmentation. In *ECCV*, 2020. 2
- [31] Abhijit Kundu, Kyle Genova, Xiaoqi Yin, Alireza Fathi, Caroline Pantofaru, Leonidas Guibas, Andrea Tagliasacchi, Frank Dellaert, and Thomas Funkhouser. Panoptic Neural Fields: A Semantic Object-Aware Neural Scene Representation. In *CVPR*, 2022. 2
- [32] Yevhen Kuznetsov, Jorg Stuckler, and Bastian Leibe. Semi-supervised deep learning for monocular depth map prediction. In *CVPR*, 2017. 2
- [33] Sihaeng Lee, Janghyeon Lee, Byungju Kim, Eojindl Yi, and Junmo Kim. Patch-wise attention network for monocular depth estimation. In *AAAI*, 2021. 2
- [34] Jie Li, Yu Liu, Dong Gong, Qinfeng Shi, Xia Yuan, Chunxia Zhao, and Ian Reid. Rgbd based dimensional decomposition residual network for 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7693–7702, 2019. 3
- [35] 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 Robotics and Automation Letters*, 5(1):219–226, 2019.
- [36] Jie Li, Kai Han, Peng Wang, Yu Liu, and Xia Yuan. Anisotropic convolutional networks for 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3351–3359, 2020. 3
- [37] Pengfei Li, Yongliang Shi, Tianyu Liu, Hao Zhao, Guyue Zhou, and Ya-Qin Zhang. Semi-supervised implicit scene completion from sparse lidar, 2021. 3
- [38] Yiming Li, Sihang Li, Xinhao Liu, Moonjun Gong, Kennan Li, Nuo Chen, Zijun Wang, Zhiheng Li, Tao Jiang, Fisher Yu, Yue Wang, Hang Zhao, Zhiding Yu, and Chen Feng. Ssbench: A large-scale 3d semantic scene completion benchmark for autonomous driving. *arXiv preprint arXiv:2306.09001*, 2023. 1, 2, 3, 5, 6
- [39] 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*, 2023. 1, 2, 3
- [40] Zhenyu Li, Zehui Chen, Xianming Liu, and Junjun Jiang. Depthformer: Exploiting long-range correlation and local information for accurate monocular depth estimation. *arXiv preprint arXiv:2203.14211*, 2022. 2
- [41] Zhenyu Li, Xuyang Wang, Xianming Liu, and Junjun Jiang. Binsformer: Revisiting adaptive bins for monocular depth estimation. *arXiv preprint arXiv:2204.00987*, 2022. 2
- [42] 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, 3
- [43] Yiyi Liao, Jun Xie, and Andreas Geiger. Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 2, 3, 5, 6
- [44] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE TPAMI*, 2015. 2
- [45] Shice Liu, Yu Hu, Yiming Zeng, Qiankun Tang, Beibei Jin, Yinhe Han, and Xiaowei Li. See and think: Disentangling semantic scene completion. In *Advances in Neural Information Processing Systems*, 2018. 3
- [46] Xiaoyang Lyu, Liang Liu, Mengmeng Wang, Xin Kong, Lina Liu, Yong Liu, Xinxin Chen, and Yi Yuan. Hr-depth: High resolution self-supervised monocular depth estimation. In *AAAI*, pages 2294–2301, 2021. 2
- [47] Ruben Mascaro, Lucas Teixeira, and Margarita Chli. Dif-fuser: Multi-view 2d-to-3d label diffusion for semantic scene segmentation. In *IEEE International Conference on Robotics and Automation*, 2021. 2
- [48] Daniel Maturana and Sebastian Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2015. 3
- [49] Nelson Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995. 4
- [50] John McCormac, Ankur Handa, Andrew Davison, and Stefan Leutenegger. Semanticfusion: Dense 3d semantic mapping with convolutional neural networks. In *IEEE International Conference on Robotics and Automation*, 2017. 2
- [51] Ruihang Miao, Weizhou Liu, Mingrui Chen, Zheng Gong, Weixin Xu, Chen Hu, and Shuchang Zhou. Occdepth: A depth-aware method for 3d semantic scene completion. *arXiv preprint arXiv:2302.13540*, 2023. 1, 3
- [52] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 2, 3
- [53] Andres Milioto, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. Rangenet++: Fast and accurate lidar semantic segmentation. In *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2019. 3
- [54] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 5

- [55] Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *CVPR*, 2023. 3
- [56] Charles R Qi, Hao Su, Matthias Nießner, Angela Dai, Mengyuan Yan, and Leonidas J Guibas. Volumetric and multi-view cnns for object classification on 3d data. In *CVPR*, 2016. 3
- [57] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017. 3
- [58] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 2017. 3
- [59] Christoph B Rist, David Emmerichs, MarkusENZweiler, and Dariu M Gavrila. Semantic scene completion using local deep implicit functions on lidar data. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7205–7218, 2021. 3
- [60] Luis Roldao, Raoul de Charette, and Anne Verroust-Blondet. Lmscnet: Lightweight multiscale 3d semantic completion. In *2020 International Conference on 3D Vision (3DV)*, 2020. 1, 3, 6, 7
- [61] Luis Roldao, Raoul De Charette, and Anne Verroust-Blondet. 3d semantic scene completion: A survey. *IJCV*, 2022. 2, 3
- [62] Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulò, Norman Müller, Matthias Nießner, Angela Dai, and Peter Kotschieder. Panoptic lifting for 3d scene understanding with neural fields. In *CVPR*, 2023. 2, 4
- [63] Chonghao Sima, Wenwen Tong, Tai Wang, Li Chen, Silei Wu, Hanming Deng, Yi Gu, Lewei Lu, Ping Luo, Dahua Lin, and Hongyang Li. Scene as occupancy. *arXiv preprint arXiv:2306.02851*, 2023. 1, 3
- [64] 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. 1, 3, 6, 7
- [65] Jaime Spencer, Richard Bowden, and Simon Hadfield. Defeat-net: General monocular depth via simultaneous unsupervised representation learning. In *CVPR*, pages 14402–14413, 2020. 2
- [66] Pei Sun, Henrik Kretschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *CVPR*, 2020. 3
- [67] S. Thrun and B. Wegbreit. Shape from symmetry. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, pages 1824–1831 Vol. 2, 2005. 3
- [68] Shubham Tulsiani, Tinghui Zhou, Alexei A Efros, and Jitendra Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2626–2634, 2017. 2
- [69] Suhani Vora, Noha Radwan, Klaus Greff, Henning Meyer, Kyle Genova, Mehdi S. M. Sajjadi, Etienne Pot, Andrea Tagliasacchi, and Daniel Duckworth. Nesf: Neural semantic fields for generalizable semantic segmentation of 3d scenes, 2021. 2
- [70] Bing Wang, Lu Chen, and Bo Yang. Dm-nerf: 3d scene geometry decomposition and manipulation from 2d images. *arXiv preprint arXiv:2208.07227*, 2022. 2
- [71] Cheng Wang, Ming Cheng, Ferdous Sohel, Mohammed Benamoun, and Jonathan Li. Normalnet: A voxel-based cnn for 3d object classification and retrieval. *Neurocomputing*, 2019. 3
- [72] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (tog)*, 2019. 3
- [73] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 5
- [74] Felix Wimbauer, Nan Yang, Lukas Von Stumberg, Niclas Zeller, and Daniel Cremers. Monorec: Semi-supervised dense reconstruction in dynamic environments from a single moving camera. In *CVPR*, 2021. 2
- [75] Felix Wimbauer, Nan Yang, Christian Rupprecht, and Daniel Cremers. Behind the scenes: Density fields for single view reconstruction. In *CVPR*, 2023. 2, 3, 4, 5
- [76] Bichen Wu, Alvin Wan, Xiangyu Yue, and Kurt Keutzer. Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3d lidar point cloud. In *IEEE International Conference on Robotics and Automation*. IEEE, 2018. 3
- [77] Xinchun Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In *Advances in neural information processing systems*, 2016. 2
- [78] 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 *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3101–3109, 2021. 3
- [79] Nan Yang, Rui Wang, Jorg Stuckler, and Daniel Cremers. Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry. In *ECCV*, 2018. 2
- [80] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *CVPR*, pages 4578–4587, 2021. 2, 3
- [81] Maciej Zamorski, Maciej Zięba, Piotr Klukowski, Rafał Nowak, Karol Kurach, Wojciech Stokowiec, and Tomasz Trzcinski. Adversarial autoencoders for compact representations of 3d point clouds. *Computer Vision and Image Understanding*, 2020. 3
- [82] Huangying Zhan, Ravi Garg, Chamara Saroj Weerasekera, Kejie Li, Harsh Agarwal, and Ian Reid. Unsupervised learn-

- ing of monocular depth estimation and visual odometry with deep feature reconstruction. In *CVPR*, 2018. 2
- [83] Cheng Zhang, Zhi Liu, Guangwen Liu, and Dandan Huang. Large-scale 3d semantic mapping using monocular vision. In *2019 IEEE 4th International Conference on Image, Vision and Computing (ICIVC)*, 2019. 2
- [84] Jiahui Zhang, Hao Zhao, Anbang Yao, Yurong Chen, Li Zhang, and Hongen Liao. Efficient semantic scene completion network with spatial group convolution. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 733–749, 2018. 3
- [85] Pingping Zhang, Wei Liu, Yinjie Lei, Huchuan Lu, and Xiaoyun Yang. Cascaded context pyramid for full-resolution 3d semantic scene completion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7801–7810, 2019. 3
- [86] Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew J. Davison. In-place scene labelling and understanding with implicit scene representation. In *ICCV*, 2021. 2, 3, 4
- [87] Hang Zhou, David Greenwood, and Sarah Taylor. Self-supervised monocular depth estimation with internal feature fusion. In *BMVC*, 2021. 2
- [88] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, 2017. 2

S4C: Self-Supervised Semantic Scene Completion with Neural Fields

Supplementary Material

A. Overview

In this supplementary, we first in Appendix C explain how we evaluated our methods on SSCBench [38] and how we refined the invalid masks. In Appendix D, we discuss further implementation details to enable easy reproduction of our experimental results. Appendix E covers potential limitations of our method and Appendix F covers potential ethical concerns. Finally, we provide the readers with more qualitative results in Appendix G.

B. Video

For further results, please watch our **provided video**, which contains per-frame predictions and comparisons for an entire subsequence of the KITTI-360 test set.

Comparisons with MonoScene [8], SSCNet [64] and LMSCNet [60] happen at timestamps 00:16 and 02:21.

C. SSCBench

C.1. Example of Training Data

Fig. 5 shows the eight frames we use to sample rays from per training sample. The different frames observe different parts of the scene. This allows the network to learn geometry of the entire scene, even though areas are occluded in the input image.

C.2. Refined Invalid Masks

SSCBench processes annotated Lidar scans to produce labeled 3D voxel grids. If a voxel contains at least one Lidar measurement point, then it is considered occupied. The voxel label is obtained by majority voting over all contained Lidar measurement points. However, only a small fraction of voxels in the scene contain Lidar points. To differentiate between empty and unknown (= "invalid") voxels, the authors perform raytracing from the Lidar origins. Invalid voxels are ignored during evaluation.

During experiments, we observed that the unknown/empty voxel maps do not align perfectly with the voxel labels. For example, a slice of one to two voxels below the street surface is considered known and empty, even though it is clearly underground and could not have been observed.

This perturbs evaluation results and gives methods that were trained directly on this ground truth a significant advantage. They can learn to imitate this wrong behavior to achieve better accuracies, even though this does not reflect real-world accuracy. To ensure a more level playing field, we therefore employ a simple strategy to refine the provided invalid masks.

All ground truth voxel grids have shape $(256 \times 256 \times 32)$ describing width, height, and depth (from a birds-eye perspective). We consider every z column separately. If a voxel in a column has no valid occupied voxel below it (*i.e.* it is most likely below ground or there exists no valid ground truth for the entire column), we consider it invalid. As an additional check, we only consider voxels with $z \leq 7$. Through empirical study, we found that this corresponds to the z level of the ground. Note that **we never set occupied voxels to invalid**. Per scene, only about 2-3% percent of the voxels are affected by our invalid refinement strategy, as shown in Fig. 6. A mathematical precise formulation of this strategy is described in Algorithm 1.

As shown in Fig. 7, our strategy finds many voxels below ground, that were previously considered to be empty and known, and sets them to invalid.

For a fair comparison, we used the provided checkpoints and reran evaluation for all methods under the exact same setting as our method, using our invalid refinement strategy. We observe that all methods achieve

Algorithm 1 Refine invalid masks

Require: $\text{Inv} \in \{0, 1\}^{256 \times 256 \times 32}$

Require: $\text{Label} \in \{0, \dots, c\}^{256 \times 256 \times 32}$

\triangleright $\text{Label} = 0$ refers to known and empty.

$\text{Inv}_{\text{new}} \leftarrow 0^{256 \times 256 \times 32}$

for $x \in \text{range}(256)$ **do**

for $y \in \text{range}(256)$ **do**

for $z \in \text{range}(7)$ **do**

$\triangleright z = 7$ is street height.

$b \leftarrow \bigwedge_{i=0}^z (\text{Inv}[x, y, i] \vee \text{Label}[x, y, i] == 0)$

$\text{Inv}_{\text{new}}[x, y, z] = b$

end for

end for

end for

return $\text{Inv}_{\text{new}} \vee \text{Inv}$

λ_{seg}	0.02	Encoder-Decoder	34885032
λ_{ph}	1	ϕ_D (Density MLP)	6721
λ_{cas}	0.001	ϕ_S (Segmentation MLP)	7891
η (learning rate)	$1e - 04$	Total weights	34899644
m (points per ray)	64	z_{near}	3
n (number of frames)	8	z_{far}	80

Table 4. Hyperparameters used during training.

slightly better results than reported in SSCBench [38]. This affirms that the strategy is fair and provides a more accurate view on the performances of the different methods.

Both code and data for the invalid refinement strategy will be released upon acceptance of the paper to promote their use in future research and to facilitate fair comparisons.

C.3. VoxFormer Reproduction

Besides MonoScene [8], another recently published work for supervised camera-based SSC is VoxFormer [39]. In the interest of a comprehensive evaluation, we wish to also provide benchmark results for this method. However, despite using the official code base of VoxFormer and using the official checkpoint for KITTI-360 provided by SSCBench [38], we were not able to reproduce the benchmark results reported in [38]. Both when using the original SSCBench settings and when using our refined invalids, we achieve results that are not competitive with the other methods.

The results in the original SSCBench work suggest that VoxFormer is approximately on par with MonoScene on SSCBench-KITTI-360. Therefore, in the interest of fairness, we choose to not report our VoxFormer results for now. We hope to work with the authors of VoxFormer and SSCBench to provide representative evaluation results for VoxFormer in the final version of this paper.

D. Implementation Details

We use the same loss parameters for all trainings except for specific ablation studies. A detailed overview of the exact hyperparameters is given in Tab. 4.

Training on a single A40 GPU takes approximately 5 days until convergence. We observed that our method is relatively robust when it comes

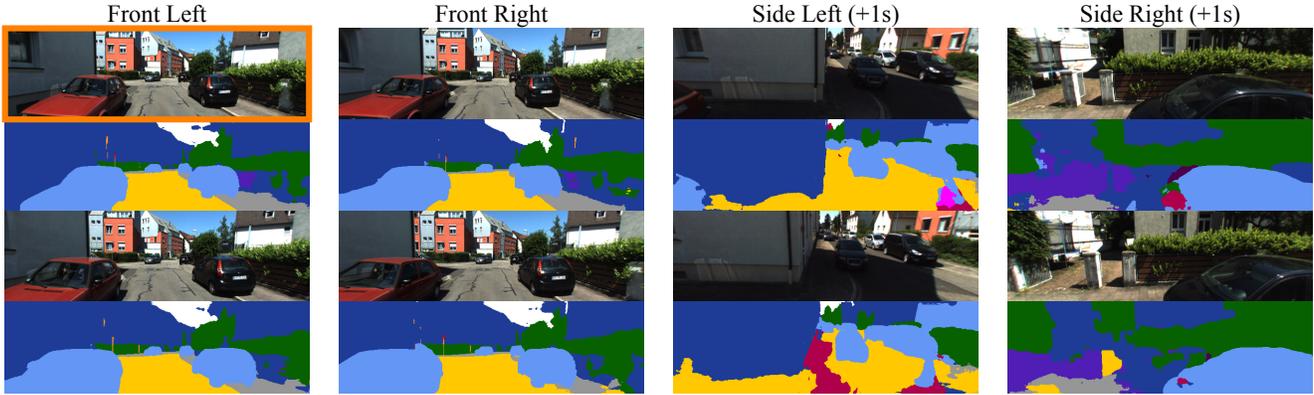


Figure 5. **Example of data sample.** Per training forward pass, we use eight frames (two timesteps, left / right, front / side) and corresponding segmentation maps which were predicted from an off-the-shelf segmentation network. The input image that is passed to the encoder-decoder is marked in orange. The different frames observe different areas of the scene, allowing us to learn geometry and semantics even in occluded regions.

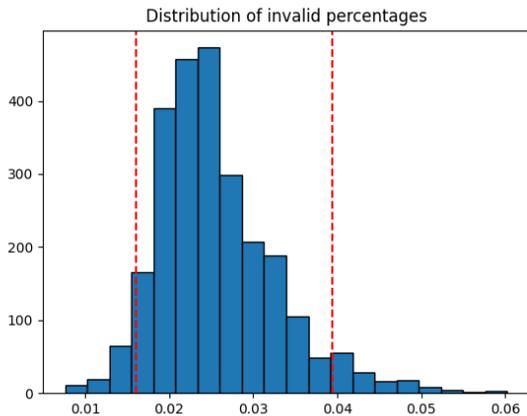


Figure 6. **Percentage of voxels affected by our strategy.** Per-scene, our invalid refinement strategy only affects approximately 2-3% of the all voxels. The histogram shows the percentage of affected voxels over the entire test set.

to changes to hyperparameters.

E. Limitations

While we believe that our proposed method achieves very strong results given that we use much weaker supervision compared to other approaches, it is not free of limitations.

As many other self-supervised methods for 3D reconstruction, we make the assumption that the world is made up of lambertian materials. This means, that the color of a surface point does not change when observed from a different angle. In our photometric loss, we therefore directly compare the pixel colors of different observations. However, this assumption does not work for some categories. For example, our method struggles to correctly reconstruct windows of a car, that are both shiny and see-through and thus appear differently when viewed from a different angle. Unlike methods that purely rely on photometric losses, however, our method is made more robust through the addition of the semantic loss, which does not suffer from this limitation.

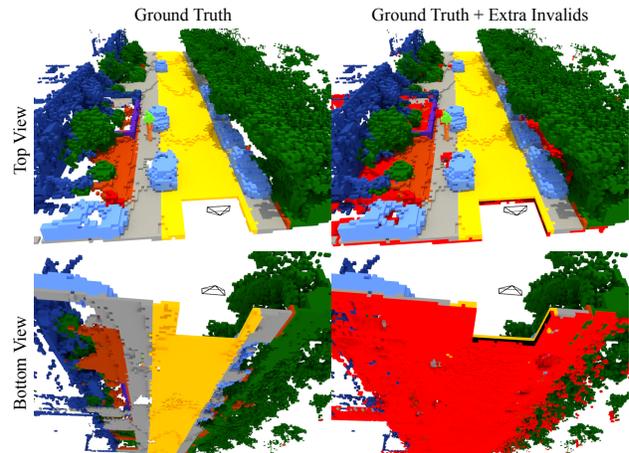


Figure 7. **Example of refined invalids found by our strategy.** We show the result of our invalid refinement strategy for an exemplary scene. The left column shows two views of the same scene (with all occupied voxels). In the right column, we add all voxels (marked in bright red) that were previously considered empty and known, but identified by our strategy as invalid.

Another limitation of our method is that our accuracy is dependent on the prediction quality an off-the-shelf image segmentation network. While forward facing views from a driving vehicle are quite common in training datasets for segmentation networks, sideways facing views are rather rare. Therefore, sometimes the segmentation predictions for the side views are not perfect. The performance of our method could potentially be improved when using a better image segmentation network.

F. Ethical and Safety Concerns

The datasets our method can be trained on potentially contain video footage of persons. To ensure privacy, faces and other identifying features should be anonymized in the dataset. Furthermore, it should be ensured that the dataset contains sufficient diversity.

Furthermore, training datasets often only contain video material from

a specific geographic location. As traffic environments in different regions of the world or even a single country look vastly different, our method is not guaranteed to generalize to unknown environments. Therefore, our method is not ready to be employed in safety-critical environments.

G. Additional Results

G.1. Rendered 2D Segmentation Maps

To extend Sec. 4.5 of the main paper, we show predicted segmentation masks for our model in Fig. 8. Furthermore, we show synthesized segmentation masks for novel views.

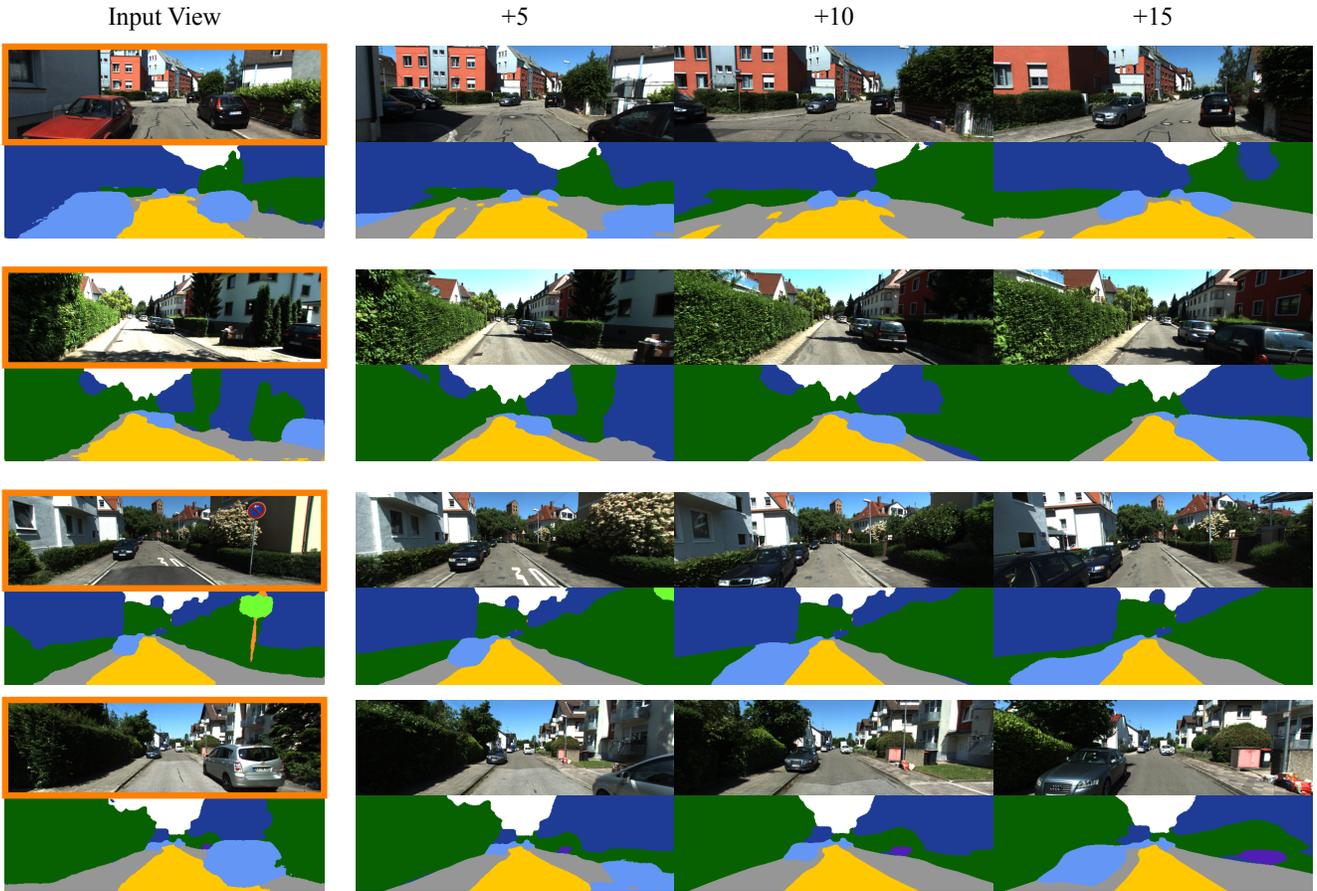


Figure 8. **Predicted segmentation masks by our model and semantic novel view synthesis.** For a given input image (marked in orange), we can predict a dense segmentation mask. Additionally, our formulation allows to synthesize segmentation masks for later frames (+5, +10, +15) of the video. Even when the new view points are far away from the input view, the synthesized segmentation masks are clean and capture the semantics correctly.