Semantic Complete Scene Forecasting from a 4D Dynamic Point Cloud Sequence

Zifan Wang^{*1,3}**, Zhuorui Ye**^{*1,3}**, Haoran Wu**^{*1,3}**, Junyu Chen**^{1,3}**, Li Yi**^{1,2,3}

[1]Tsinghua University, [2]Shanghai Artificial Intelligence Laboratory, [3]Shanghai Qi Zhi Institute
{wzf22,yezr21,wuhr20,junyu-ch21}@mails.tsinghua.edu.cn, ericyi@mail.tsinghua.edu.cn

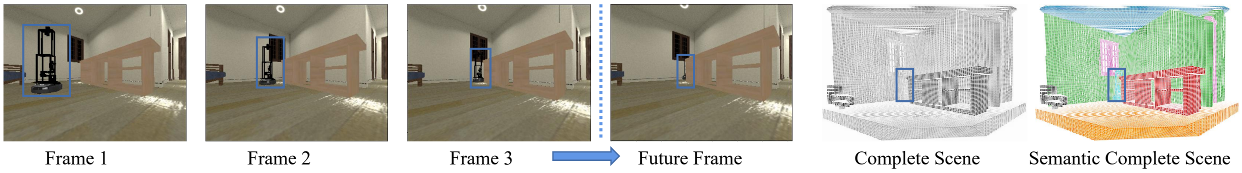

Figure 1: **Semantic Complete Scene Forecasting.** Combining past observations of an interaction scene, we can forecast the robot is about to walk behind a desk and will be occluded. Specifically, the SCSF task consumes a 4D point cloud sequence(here RGB is just to intuitively show the scene contents) and forecasts the complete scene with semantic labels in the whole space in the next frame. The SCSF task gives a comprehensive understanding of the future scene as shown in the last two figures.

Abstract

We study a new problem of semantic complete scene forecasting (SCSF) in this work. Given a 4D dynamic point cloud sequence, our goal is to forecast the complete scene corresponding to the future next frame along with its semantic labels. To tackle this challenging problem, we properly model the synergetic relationship between future forecasting and semantic scene completion through a novel network named SCSFNet. SCSFNet leverages a hybrid geometric representation for high-resolution complete scene forecasting. To leverage multi-frame observation as well as the understanding of scene dynamics to ease the completion task, SCSFNet introduces an attention-based skip connection scheme. To ease the need to model occlusion variations and to better focus on the occluded part, SCSFNet utilizes auxiliary visibility grids to guide the forecasting task. To evaluate the effectiveness of SCSFNet, we conduct experiments on various benchmarks including two large-scale indoor benchmarks we contributed and the outdoor SemanticKITTI benchmark. Extensive experiments show SCSFNet outperforms baseline methods on multiple metrics by a large margin, and also prove the synergy between future forecasting and semantic scene completion. The project page with code is available at scsfnet.github.io.

1 Introduction

Visual forecasting has aroused a wide spectrum of interests in the computer vision community, especially for RGB videos(Zhang et al. 2019a; Hu and Wang 2019; Luc et al. 2020; Gao et al. 2019; Hu et al. 2020). Recently point cloud sequence forecasting (Wen et al. 2022; Sun et al. 2020; Deng and Zakhor 2020a; Weng et al. 2021; Mersch et al. 2022; Wang and Tian 2022) has also obtained many attention. Forecasting the future geometric observation can help an intelligent agent plan its behavior accordingly, which in turn greatly benefits a wide range of downstream applications in autonomous driving, robotics, and augmented reality.

Existing works on point cloud forecasting (Wen et al. 2022; Sun et al. 2020; Deng and Zakhor 2020a; Weng et al. 2021; Mersch et al. 2022) mostly focus on forecasting raw future observation. The observation only provides a partial view of the dynamic scene, insufficient for a robot to perform low-level tasks like grasping and obstacle avoidance (Song et al. 2017) that require complete geometric understanding. Also, the raw future point cloud lacks semantic meaning, crucial for high-level tasks like object retrieval. Therefore, we propose a new task named **Semantic Complete Scene Forecasting (SCSF)** in this work. Given previous N frames of an egocentric 4D dynamic point cloud sequence, the task is to predict the complete scene along with its semantic labels corresponding to the next future frame.

The SCSF problem is very challenging since it requires both future forecasting and semantic scene completion (Song et al. 2017), or more intuitively, completing and segmenting something unknown. Previous works on point cloud forecasting (Ilg et al. 2017; Dosovitskiy et al. 2015) often formulate it as a flow prediction problem where future frames are treated as the deformation of earlier partial observations. However, they cannot provide a complete scene understanding which is crucial for occlusion inference or robot decision-making. On the other hand, works on semantic scene completion (Roldao, De Charette, and Verroust-Blondet 2022) consider only static scenes. Previous research treated forecasting and completion as two separate problems and no works exploited their connections deeply.

Our key observation is that **future forecasting and semantic scene completion are synergetic to each other** and can lead to huge gains mutually if modeled properly. On one hand, future forecasting models the underlying scene dynamics and improves motion understanding. Such understanding allows aggregating multiple observations across time for a more complete geometric understanding. On the other hand, a complete scene understanding allows future forecasting to focus on the object movements without bothering with sampling noise or occlusion variations faced by traditional point cloud forecasting methods. Thus semantic scene completion can greatly improve future forecasting and provide a comprehensive understanding of the scene.

Based on the observation, we jointly solve the forecasting and the semantic completion in 4D. We struggle to answer three questions: How can we forecast the complete future scene with high resolution? How can we leverage the multi-frame partial observations to solve the completion task? How can we leverage the prior of empty and occluded space, which is usually ignored in point cloud forecasting problems, to tackle the forecasting task? We address the above issues through a novel network named **SCSFNet**.

Specifically, to address the first issue, unlike traditional semantic scene completion methods that use low-resolution dense voxel representation, SCSFNet leverages a hybrid representation combining the benefits of sparse voxels, dense voxels, and implicit fields for high-resolution predictions. It employs a 4D sparse voxel encoder for fine-grained input information, uses dense voxels for flexible low-resolution structure generation, and predicts an implicit field from the low-resolution dense voxel grid for high-resolution predictions. To address the second issue, we utilize 4D sparse convolutions to aggregate multi-frame observations in the past and introduce attention-based skip connections in SCSFNet. This innovative design improves information propagation and occlusion inference for more precise future scene forecasting.

To address the third issue, we introduce an auxiliary task of forecasting a future visibility grid from past ones. This is a much easier task than predicting an entire high-resolution scene. The visibility grids' ambient information effectively aids in scene understanding.

To confirm the effectiveness of our method for intelligent agents, we construct two large-scale SCSF benchmarks, **IGPLAY** focusing on robot interaction with indoor objects, and **IGNAV** focusing on robot social navigation with other dynamic agents in scenes. Besides the two indoor datasets, we also use SemanticKITTI (Behley et al. 2019) to verify our method for outdoor scenarios. Extensive experiments demonstrate that by jointly modeling geometry forecasting and semantic completion, SCSFNet can outperform baseline methods by a large margin (11.6% CD relative improvements on the point cloud forecasting task and 10.8% mIoU improvement on the SCSF task, on the IGPLAY dataset).

To summarize, our main contributions are fourfold: i) We propose a new task of semantic complete scene forecasting from a 4D dynamic point cloud sequence; ii) We present SCSFNet to exploit the synergy between forecasting and semantic scene completion to effectively solve such a new task; iii) To evaluate our method, we introduce IGPLAY and IGNAV, two large-scale 4D egocentric vision datasets with complete geometry and semantic annotations covering abundant indoor scenes and rich dynamics; iv) Our experiments verify the effectiveness of our method and prove the synergy between forecasting and semantic scene completion.

2 Related Work

4D Sequential Point cloud Forecasting. Point cloud forecasting is crucial for understanding scene geometry and motion dynamics(Fan, Yang, and Kankanhalli 2021; Wen et al. 2022). Various methods have been employed in different scenarios. TLFPAD (Deng and Zakhor 2020a) proposed using scene stream embedding to model past point cloud frames' temporal relationships for future frame prediction. Mersch *et al.*(2022) proposed to use 3D convolution to jointly learn the spatial-temporal features of the input point cloud sequence. These approaches primarily emphasize visible surfaces rather than the entire scene. Occlusion4d (Van Hoorick et al. 2022) introduced a framework to estimate 4D visual representations from monocular RGB-D video, which encodes point clouds into a continuous high-resolution representation. Unlike Khurana *et al.*(2023), whose work attained SOTA point cloud forecasting using sensor extrinsics, our approach doesn't rely on camera poses or perfect odometry for scene alignment. Instead, we aim to jointly solve forecasting and semantic completion in the egocentric view across both indoor and outdoor environments.

Semantic Scene Completion. Semantic Scene Completion (Song et al. 2017) has gained significant momentum in the research community due to the unresolved challenges it faces (Li et al. 2019; Wang et al. 2019; Liu et al. 2020b; Chen et al. 2020; Cheng et al. 2021; Rist et al. 2021). **This task's output is a 3D voxel grid with a semantic label for each voxel.** SISNet (Cai et al. 2021) aims to deduce detailed shape information and nearby objects of similar categories. Without extra instance labels, SCSFNet can also obtain detailed geometric data using the implicit field based on the coarse voxel grid. Most works in this field only aim to process 3D static scenes like NYU and SUNCG (Silberman et al. 2012; Song et al. 2017; Firman et al. 2016; Xie et al. 2020; Chang et al. 2017) without any temporal information. We contribute two 4D dynamic indoor datasets IGPLAY and IGNAV. As for outdoor scenarios, SemanticKITTI (Behley et al. 2019) is a dataset suitable for Point Cloud Forecasting and 3D Semantic Scene Completion, allowing direct use of the former's input and the latter's ground truth for SCSF.

Hybrid Geometric Representations. Hybrid geometric representations (Ali et al. 2018; Zhang et al. 2020; Song, Song, and Huang 2020) has aroused great interest (Dourado, Guth, and de Campos 2022; Xu et al. 2019; Peng et al. 2020) for various downstream tasks. Convolutional Occupancy Networks (Peng et al. 2020) combines convolutional encoders with implicit occupancy decoders for 3D reconstruction. Our SCSFNet further utilizes implicit fields for 4D complete forecasting. GRNet (Xie et al. 2020) and SCSFNet both leverage point cloud input and intermediate 3D voxel grids, but SCSFNet decodes infinite implicit fields unlike GRNet converting the voxels back to point clouds explicitly with a limited resolution. While Liu *et al.*(2020a) leverages self-pruning and sparse voxels while overfitting a scene, we

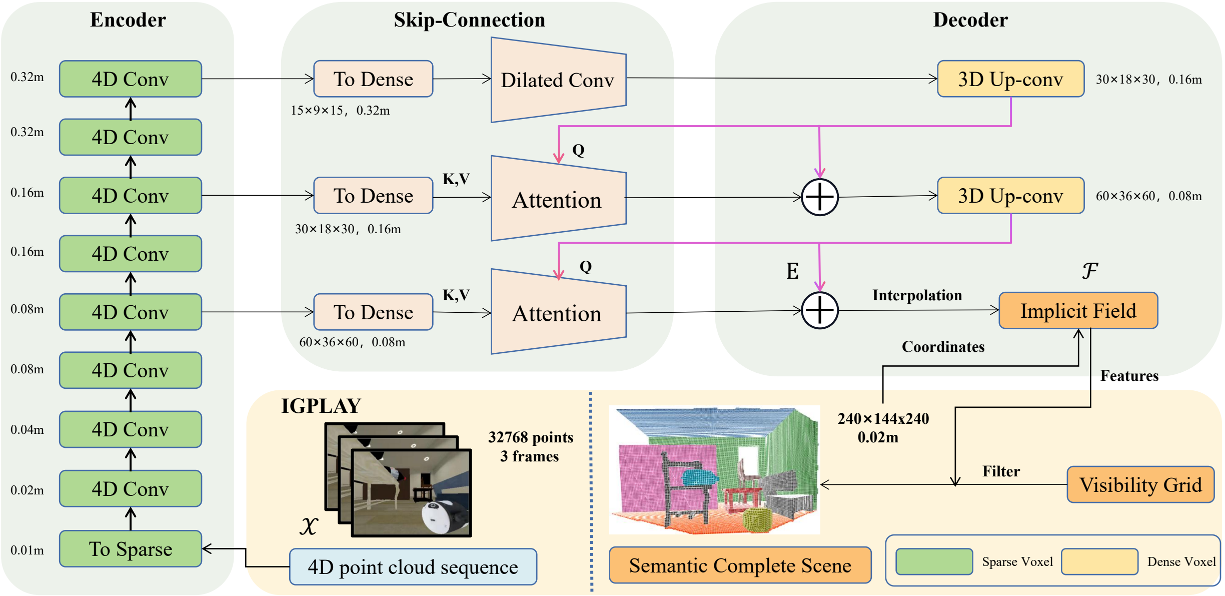

Figure 2: **The overview of SCSFNet.** First, a 4D point cloud sequence $\mathcal{X}$ will be transformed into 0.01m sparse voxels. The 4D sparse convolution encoder then generates spatial-temporal features at different scales (0.08m, 0.16m, and 0.32m). They are filled into 3D dense voxel grids and processed with attention-based skip connections at each scale. Finally, the model produces an implicit field $\mathcal{F}$ based on a low-resolution voxel grid E (0.08m), which is then used to generate a high-resolution voxel output (0.02m) filtered by the visibility grid for evaluation.

instead use dense voxels enriched with skipped point cloud features to dynamically determine scene geometry.

3 Method

We first define the **Semantic Complete Scene Forecasting (SCSF)** task formally. Let $P_t \in \mathbb{R}^{N\times 3}$ and $F_t \in \mathbb{R}^{N\times C}$ denote the coordinates and features of the t-th frame in a point cloud sequence, where N and C denote the number of points and feature channels. Given a point cloud sequence $\mathcal{X} = ([P_1, F_1], \ldots, [P_L, F_L])$ as input, the SCSFNet initially produces a low-resolution voxel feature grid $E \in \mathbb{R}^{X\times Y\times Z\times D}$, where D represents the feature dimensions, and X, Y, Z denote the dimensions of the low-level grid. Subsequently, Based on E, the SCSFNet then conducts interpolations and yields an implicit field $\mathcal{F} : \mathbb{R}^3 \to \mathbb{Z}$, describing the semantic label at an arbitrary point in the next frame. To generate a specific high-resolution voxel grid, we can obtain semantic occupancy of each voxel by querying their central coordinates in $\mathcal{F}$.

Our method, SCSFNet, uses an egocentric 4D point cloud sequence with N frames to predict future scenes with semantic information. It employs an encoder-decoder structure and hybrid geometric representations for high-resolution forecasting. Like U-Net (Ronneberger, Fischer, and Brox 2015), SCSFNet uses skip connections between the encoder and decoder to retain high-resolution input information. However, challenges arise in its design. Traditional scene completion methods use dense voxel grids in the decoder, resulting in a resolution limit. It is also complex to allow skip connections between a partial input and a complete prediction at different resolutions. Further, previous point cloud forecasting methods solely operate on a point cloud representation, making it difficult to utilize ambient information such as empty or occlusion space distribution.

To address the three challenges mentioned above, we utilize a hybrid geometric representation for high-resolution generation, design attention-based skip connections for features from partial to complete, and also introduce visibility grids to leverage ambient information. In the following, we will explain how we leverage a hybrid geometric representation to design SCSFNet in Section 3.1. Then we introduce our attention-based skip connections and visibility grids in Section 3.2 and Section 3.3 respectively.

3.1 SCSFNet with a Hybrid Geometric Representation

Semantic complete scene forecasting involves predicting future geometry and completing the scene. Traditional point cloud forecasting methods focus on point cloud representation, making it difficult to recover complete scene geometry. In semantic scene completion, the dense voxel grid is commonly used due to its ease in generating new structures, but its resolution constraint limits its use for tasks requiring detailed geometric understanding in large scenes. Hence, we utilize a hybrid geometric representation that combines the advantages of point clouds, sparse and dense voxels, and implicit fields for high-resolution predictions.

We design SCSFNet, an encoder-decoder structure that encodes the high-resolution point cloud sequence input and decodes an implicit field with infinite resolution based on a low-resolution voxel grid, as shown in Figure 2.

SCSFNet effectively processes input with minimal resolution loss by filling points into small voxels and utilizing Minkowski Engine (Choy, Gwak, and Savarese 2019) for 4D sparse convolutions. By incrementally increasing the convolution stride, we enlarge voxel sizes and obtain multi-scale features. Compared with 4D point cloud encoders like P4Transformer (Fan, Yang, and Kankanhalli 2021), the sparse voxel-based encoder aligns better with a dense voxel decoder, which is commonly used for semantic scene completion and structure generation.

After our 4D sparse convolution encoder extracts spatial-temporal features at different scales, we fill the aggregated features of sparse voxels into a 3D dense voxel grid with the same resolution at each scale. This information is fed to the decoder through skip connections, similar to U-Net (Ronneberger, Fischer, and Brox 2015). More details about our attention-based skip-connections are in Section 3.2.

The dense voxel exploited by our decoder makes it very easy to generate new geometry through up-convolutions. We leverage several 3D up-convolution layers to lift the resolution of the voxel grid to a higher scale. However, due to the restriction of memory, we cannot reach high resolution with such a dense voxel representation. Inspired by Convolutional Occupancy Networks (Peng et al. 2020), we further convert such a dense voxel feature grid into a semantic occupancy field to enable infinite resolution in the output. In particular, for an arbitrary point $p \in \mathbb{R}^3$, we obtain its feature by conducting a trilinear interpolation using the features of nearby voxels. Then the point feature goes through a Multi-layer Perceptron (MLP) to reach a semantic occupancy prediction where the label is either empty or a semantic class.

Combining a sparse voxel encoder, a dense voxel decoder, and an implicit field, we can encode the high-resolution input with little information loss while at the same time forecasting a complete scene with infinite resolution.

During the training phase, we use a loss based on the high-resolution voxel ground truth. This ground truth has a higher resolution compared with the dense voxel grid on which the implicit field is based. We randomly sample 10^5 points from all central coordinates of ground truth voxels on the fly during training and evaluate the semantic occupancy values of these points. We use a cross-entropy loss $\mathcal{L}_{\text{high}}$ to supervise. During inference, we can forecast a high-resolution dense voxel grid by simply querying the semantic occupancy of the central coordinate of each voxel. It is noted that the SCSFNet is trained and evaluated using points from the ground truth voxel centroids, rather than exact implicit functions. This approach is chosen due to the absence of ground truth meshes in existing datasets with high-resolution information, rendering it unfeasible to acquire precise implicit functions for both training and evaluation purposes. Moreover, this methodology aligns with common practices in baseline methods, which often perform experiments using point clouds, ensuring a fair and meaningful comparison.

3.2 Attention-Based Skip Connection

The naive encoder-decoder structure struggles to transfer detailed information in the bottleneck layer, making dense prediction tasks tough. Researchers have suggested using skip connections in a U-Net structure (Ronneberger, Fischer, and Brox 2015) to shortcut high-resolution features, which has improved tasks like semantic segmentation. However, the modality gap between the point cloud input and the dense voxel grid output presents a more significant challenge. This can be partially addressed by voxelizing the input and using a sparse voxel encoder, but transitioning from partial observations to complete predictions remains difficult.

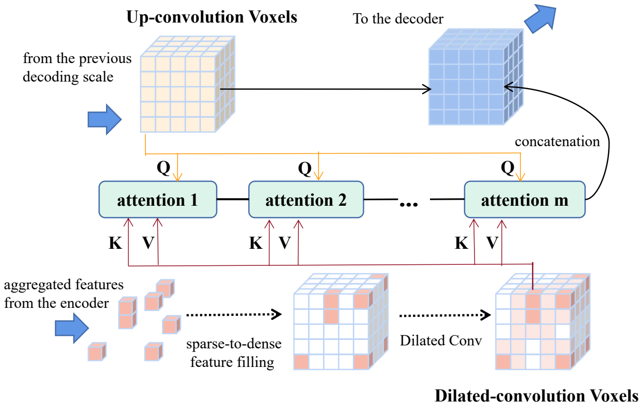
Up-convolution Voxels
To the decoder
from the previous decoding scale
concatenation
Q
Q
Q
attention 1
attention 2
...
attention m
K
V
K
V
K
V
aggregated features from the encoder
sparse-to-dense feature filling
Dilated Conv
Dilated-convolution Voxels

Figure 3: **Illustration of attention-based skip connections.** Dilated-convolution voxels from the encoder form the key and value. Up-convolution voxels from the previous scale form the query. Finally, we concatenate voxel features from multi-head cross attention and feed them to the decoder.

To address the above challenge, we introduce the attention-based skip connections as shown in Figure 3. Given a set of sparse voxels at a certain scale of our encoder and a dense voxel grid at the corresponding scale of the decoder, our goal is to feed the sparse voxel features to the dense voxels in a similar fashion to traditional skip connections. An intuitive approach would be to simply add sparse voxels from the encoder to the dense voxels of the decoder. However, this method provides no information to the majority of the voxels, which are crucial for accurately forecasting a complete scene. Our attention-based skip connections are designed to alleviate this problem.

Specifically, we obtain 3D sparse aggregated features in the encoder, which has consolidated spatial-temporal details at different scales. These features are then filled in dense voxel grids, but only a limited subset of voxels contain information. To diffuse the encoded details efficiently, we utilize dilated convolutions. The outcomes of this step are referred to as **"dilated-convolution voxels"** as shown in Figure 3. And we call the voxels produced through 3D up-convolutions in the decoder as **"up-convolution voxels"**.

At the most coarse-grained scale, dilated-convolution voxels are directly processed with up-convolutions to enter the next scale. At other scales, we combine dilated-convolution voxel features with positional embeddings, generating key-value pairs (K_j, V_j) for key and value matrices K and V respectively. Positional embeddings are computed using the Fourier basis of voxel centroid positions through an MLP. For the i-th up-convolution voxel at the previous scale, a query Q_i is derived from its voxel feature and positional embedding. This allows us to compute

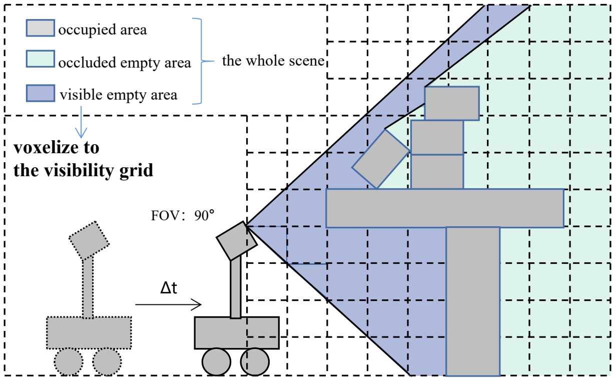

Figure 4: **Illustration of the visibility grid.** An iGibson robot is nearing a table. Forecasting visible empty areas is much easier than the whole empty areas. We specifically forecast the future visibility grid to assist the SCSF task in predicting the entire scene.

the skipped feature for the i-th up-convolution voxel using $\text{softmax}(\frac{Q_i K^T}{\sqrt{d_k}})V$, where d_k is the dimension of Q_i. In practice, we utilize multi-head cross-attentions and concatenate them to the final skipped features. Following U-Net's common practice, we combine skipped features with original up-convolution voxel features, which are then forwarded to the decoder.

Such design allows feeding high-resolution features directly from the encoder to every voxel in the decoder, facilitating implicit geometry completion and scene forecasting.

3.3 Visibility Grid

There is a strong ambient space prior in a point cloud from which we can extract the visible empty area: if we shoot a ray from the depth camera to a depth point in the scene, the space between the camera and the point should be empty; when we further extend this ray, it would reach the occluded space. This information is critical for a complete scene forecasting task since our main job is to inpaint the occupied area but not the visible empty space, as shown in Figure 4. This information is rarely used in traditional point cloud forecasting tasks since with a point cloud representation, such ambient information is hard to encode. Thanks to the hybrid representation, we can emphasize such important information for the SCSF task.

In particular, we introduce a dense visibility grid to model the information. By voxelizing each frame of the input point cloud sequence and projecting rays to identify visible voxels, we extract valuable visibility data. The size of the visibility grid matches the high-resolution ground truth voxel dimensions. This visibility grid is intended to complement our predicted results, but the challenge is that we know nothing about the scene in the next frame.

To address this issue, we introduce an auxiliary task of forecasting a future visibility voxel grid from past visibility grids. This task is much easier than recovering a complete scene with semantics in infinite resolution, so we can train a relegated version of SCSFNet to obtain a high-resolution visibility voxel grid with binary labels. Then we use the grid as an additional mask to filter out visible empty areas for the main SCSFNet output, which can ease the need to model occlusion variations when solving the challenging SCSF task.

4 Datasets

In order to train and evaluate our model, and to demonstrate the ability of our model for 4D completion and forecasting, we require sequential point clouds with both 3D semantic scene completion annotations and underlying dynamics that can be used for forecasting. For this purpose, we contribute two high-quality synthetic datasets IGPLAY and IGNAV using the Interactive Gibson Simulator (iGibson) (Xia et al. 2020) that runs on top of the pyBullet (Coumans and Bai 2016). To generate the simulated scenes suitable for our task, we specify the desired configuration of the environment in the logic language BDDL. With the BDDL description and a list of scene names, the logic states implemented in iGibson 2.0 provide a mechanism that facilitates the generation of simulated scenes with various objects. In our datasets, we provide 2D pictures (RGB, normal and semantics), 3D point clouds, 3D meshes, 3D visibility grids and 3D ground truth voxel grids. The train/test split is $80\%/20\%$. The time unit in iGibson is "timestep" instead of "second", so we just specify how far the robot and objects move in each timestep by iGibson interfaces. We provide more statistics in the supplementary. Besides the two indoor datasets we provide, the outdoor SemanticKITTI dataset is also suitable for our task. Given the limited options for real-world datasets meeting these criteria, we find SemanticKITTI to be the only suitable choice, which consists of large-scale 4D point cloud sequences with completion and semantic annotations. We hope to extend the evaluation of our approach to suitable real-world datasets released in the future.

4.1 IGPLAY & IGNAV

We present the IGPLAY dataset by capturing 1,000 scenes lasting 10 timesteps each, where a viewer interacts with various toys among the furniture. Each scene in IGPLAY contains 10 semantic classes in a great variety. When the viewer (an iGibson robot) moves and interacts with objects, partial and complete occlusions happen between the furniture and objects. This dataset supports effective learning for 4D dynamic scene completion and forecasting.

While IGPLAY already exhibits many features suitable for 4D forecasting, the whole dynamics of the scene only come from the viewer's movement and interaction with the objects. To provide a new perspective of the underlying dynamics, we provide a new dataset called IGNAV with several active robots navigating in the scene. Each scene in IGNAV contains 9 semantic classes. We captured 600 scenes lasting 10 timesteps each, where several robots move around the scene in random speeds, with another robot watching them.

4.2 SemanticKITTI

SemanticKITTI (Behley et al. 2019) is a very challenging and well-known large-scale outdoor dataset collected by autonomous cars. Various occlusions by trees or other cars make the dataset suitable for evaluating our SCSF task.

	IGPLAY		IGNAV		SemanticKITTI	
Method	IoU	mIoU	IoU	mIoU	IoU	mIoU
Occlusion4d (Van Hoorick et al. 2022) (CSF)	39.7	-	45.0	-	16.4	-
Occlusion4d (Van Hoorick et al. 2022) (SCSF)	40.8	23.8	46.1	24.6	17.1	6.4
TLFPAD (Deng and Zakhor 2020b) (CSF)	48.2	-	60.2	-	32.3	-
TLFPAD (Deng and Zakhor 2020b) (SCSF)	48.6	25.8	60.7	25.9	32.6	9.5
ST3DCNN (Mersch et al. 2022) (CSF)	49.1	-	61.4	-	28.8	-
ST3DCNN (Mersch et al. 2022) (SCSF)	49.4	25.5	62.9	26.1	29.5	9.6
SCSFNet (CSF)	53.9	-	63.6	-	33.7	-
SCSFNet (SCSF)	**56.5**	**36.3**	**69.5**	**39.9**	**34.5**	**16.1**

Table 1: **SCSF Results on IGNAV, IGPLAY, and SemanticKITTI**. We compare SCSFNet with baselines on both the CSF and the SCSF task using IoU and mIoU (in percentages). Our method significantly outperforms baselines, especially in mIoU.

	IGPLAY		IGNAV		SemanticKITTI	
Method	IoU(M)	IoU(S)	IoU(M)	IoU(S)	IoU(M)	IoU(S)
Occlusion4d (Van Hoorick et al. 2022) (SCSF)	6.3	28.2	4.4	27.1	3.4	7.8
TLFPAD (Deng and Zakhor 2020b) (SCSF)	11.9	29.3	2.1	28.8	4.9	11.7
ST3DCNN (Mersch et al. 2022) (SCSF)	9.3	29.6	0.0	29.3	4.8	11.9
SCSFNet (SCSF)	**28.2**	**38.4**	**18.3**	**42.6**	**12.4**	**17.8**

Table 2: **SCSF Results on dynamics of IGNAV, IGPLAY, and SemanticKITTI**. In the same setting as Table 1, we compare Movable IoU and Static IoU (in percentages) for the SCSF task. Our method outperforms baselines, especially in Movable IoU.

5 Experiments

We evaluate our proposed methods on two indoor synthetic datasets (IGPLAY and IGNAV) and one outdoor real-world dataset (SemanticKITTI).

For IGPLAY and IGNAV, the dimensions of the 3D space are 4.8m horizontally, 2.88m vertically, and 4.8m in depth. We use 3 RGB frames (interval: 1 iGibison timestep) as the input and a $240 \times 144 \times 240$ volume with grid size 0.02m as the ground truth, which is similar to (Song et al. 2017).

For SemanticKITTI, the dimensions of the 3D space are 51.2m ahead of the car, 25.6m to every side of the vehicle, and 6.4m in height. We input three frames of point clouds (0.2s interval) and use a $256 \times 256 \times 32$ volume with a 0.2m grid size as the ground truth, provided by SemanticKITTI for the semantic scene completion benchmark.

The three datasets have distinct differences. IGPLAY and IGNAV use a 480×640 image from a depth camera, convertible to a point cloud, while SemanticKITTI uses a raw LiDAR point cloud with varying points. IGPLAY and IGNAV contain more dynamic objects, whereas SemanticKITTI features larger spaces and complex real-world details. Well managing these datasets shows the versatility of SCSFNet.

5.1 Evaluation Metrics

To evaluate forecasting results, we obtain high-resolution voxels or point clouds from implicit fields. We evaluate scene completion quality using voxel-level intersection over union (**IoU**) between predicted and ground truth labels. If labels contain semantics, we assess semantic scene completion quality using mean IoU across all classes (**mIoU**). For point clouds, we use point-level Chamfer Distance (**CD**).

5.2 Evaluating SCSFNet on the CSF Task and the SCSF Task

For the SCSF task defined above, we need to forecast the semantic scene completion. We can also delegate SCSF into the Complete Scene Forecasting (CSF) task, where we only forecast the scene completion as a binary classification.

Baselines. Since this paper is pioneering the SCSF task, there are no existing methods for comparison. We devise baselines from existing modern methods. The input is a 3-frame point cloud sequence and the output is the future complete scene.(1) *Occlusion4d* (Van Hoorick et al. 2022) is designed for estimating 4D visual representations and occlusions from RGB-D video. It can be trained end-to-end for the SCSF task. (2)*TLFPAD* (Deng and Zakhor 2020b), a FlowNet3D-based point cloud forecasting method is used for partial forecasting. Therefore we first forecast a partial point cloud of the next frame using TLFPAD, and then train a SCSFNet exclusively for (semantic) scene completion on the forecasted point cloud. (3) *ST3DCNN* (Mersch et al. 2022), a CNN-based point cloud forecasting method achieves SOTA (without sensor extrinsics) on the KITTI-Odometry dataset (Geiger, Lenz, and Urtasun 2012). We follow the same two-stage training approach as TLFPAD.

Comparison results between SCSFNet and baselines. SCSFNet outperforms baselines on three different datasets by a large margin as shown in Table 1. Thanks to the hybrid geometric representation and the implicit field, SCSFNet obtains high-resolution forecasting on both indoor and outdoor scenes. We observe Occlusion4d perform not well for whole-scene forecasting. Compared with SCSFNet on IGPLAY (SemanticKITTI), ST3DCNN has a $7.1\%(5\%)$ reduction in IoU but a $10.8\%(6.5\%)$ reduction in mIoU. Separating forecasting and semantic completion into two stages hampers completion and extraction of semantics, and since semantics prediction is harder and requires more information, the drop in mIoU is a bit larger. SCSFNet excels by jointly incorporating semantic scene completion and forecasting. The results are visualized in Figure 5.

Comparison results between the CSF and SCSF Task. Two Adjacent rows in Table 1 compare the performance of the same method across the CSF task and SCSF task. Notably, SCSFNet displays improvements: a 2.6% enhancement in IGPLAY, a 5.9% enhancement in IGNAV, and a 0.8% enhancement in SemanticKITTI. Baselines similarly

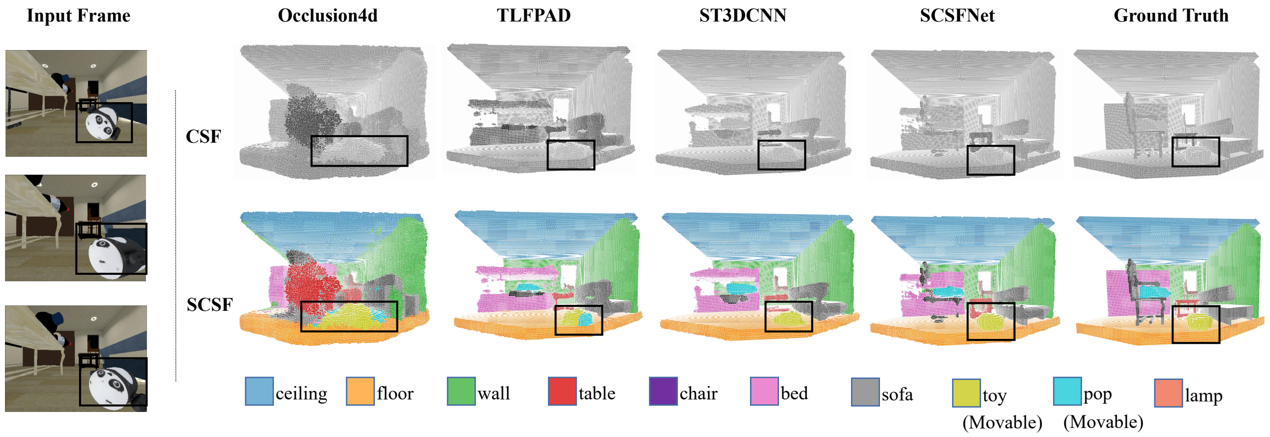

Figure 5: **Visualization results in IGPLAY.** In this sequence, an iGibson robot is approaching a panda toy on the ground and is going to hit it. Occlusion4d predicts the fuzzy shape of the toy. TLFPAD predicts half of the toy as the pop class. In comparison, SCSFNet can get relatively accurate geometry and semantics of the toy compared with the ground truth.

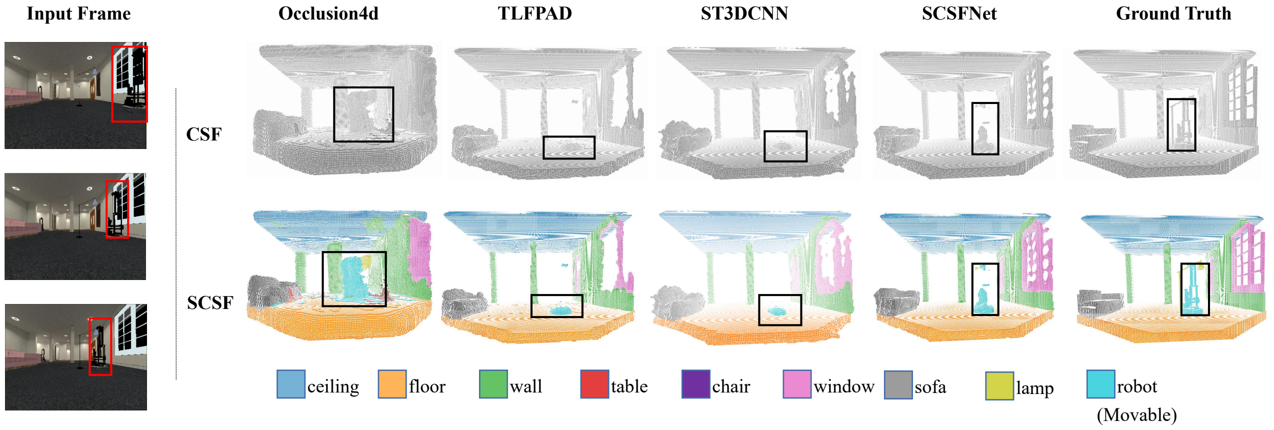

Figure 6: **Visualization results in IGNAV.** In this sequence, an iGibson robot is watching another robot beside the window going left. SCSFNet can gain more structures of this “slim” robot and other objects such as the window and the sofa.

exhibit improved IoU scores. This underscores the value of object semantic understanding for scene completion.

Comparison results between movable and static classes. We also report IoU when evaluating movable or static classes in Table 2. A class is labeled as movable if the corresponding objects can move, such as robots in IGNAV or cars in SemanticKITTI. This distinction allows us to emphasize a method’s ability to comprehend scene dynamics and forecast movable classes. In the SCSF task, we categorize predicted voxels as either movable or static based on their predicted semantic labels. This allows us to calculate IoU for each category and subsequently determine the Average Movable IoU and Average Static IoU. Compared to ST3DCNN on the SCSF task, our SCSFNet leads to 18.9%(8.8%), 18.3%(13.3%), 7.6%(5.9%) improvements in movable (static) IoU for IGPLAY, IGNAV, and SemanticKITTI respectively. Therefore, SCSFNet’s concurrent forecasting and completion harness the advantageous interaction. This integrated approach improves the handling of dynamic objects, a critical factor in accurate future forecasting. More details are provided in the supplementary.

Methods	IGPLAY	IGNAV
TLFPAD (Deng and Zakhor 2020b)	0.045	0.043
ST3DCNN (Mersch et al. 2022)	0.043	0.037
SCSFNet (SCSF)	**0.038**	**0.030**

Table 3: **Results on the point cloud forecasting of the next frame**. We report the Chamfer Distance of the point cloud of the next frame (lower is better, the unit is square meter).

5.3 Completion Helps Point Cloud Forecasting

We evaluate SCSFNet on point cloud forecasting task to verify semantic scene completion helps forecasting. Using iGibson’s intrinsic and extrinsic parameters, we project forecasted complete high-resolution voxels onto partial ones in IGPLAY and IGNAV, from which we can extract a fine-grained point cloud.

Configurations. TLFPAD (Deng and Zakhor 2020b) and

Methods	IoU	mIoU
SSC (from scratch)	60.1	39.5
SSC (finetuned by SCSF)	**62.3**	**40.5**

Table 4: **Results on the Semantic Scene Completion task**. The two experiments are on IGPLAY. The first model is trained in the SSC setting from scratch and the second model is pre-trained in the SCSF setting and fine-tuned in the SSC setting.

ST3DCNN (Mersch et al. 2022) are tailored to predict the point cloud of the future frame. We naturally train these two baselines end-to-end, using the next frame's point cloud as the ground truth. In comparison, SCSFNet is trained as usual (on SCSF task), and we use the above complete-to-partial projection to obtain a partial point cloud from the predicted complete scene.

Results. By jointly considering semantic completion and forecasting, our SCSFNet outperforms ST3DCNN, improving CD from 0.043 to 0.038 in IGPLAY and from 0.037 to 0.030 in IGNAV, as shown in Table 3.

5.4 Forecasting Helps Semantic Completion

We show that future forecasting benefits semantic scene completion in the Semantic Scene Completion task (Song et al. 2017). The Semantic Scene Completion task does not introduce any forecasting knowledge as described in Section 2.

Configurations. We conduct two IGPLAY experiments. (1)Train a SCSFNet from scratch with 3 past frames to predict the complete scene in the last frame. (2) Train a SCSFNet for the SCSF task (using 3 past frames to predict the scene in the future frame), and finetune it to predict the complete scene in the last given frame.

Results. The SCSFNet finetuned from the SCSF task improves IoU by 2.1% and mIoU by 1.0%, as shown in Table 4. This demonstrates the critical role of additional forecasting knowledge in semantic scene completion. The model, trained within the context of the SCSF task, acquires the ability to forecast, thereby offering deeper insights into scene completion.

We establish the synergy between completion and forecasting by proving both directions in Section 5.3 and Section 5.4.

5.5 Ablations

Table 5 shows how different architectural choices affect our network's performance for the SCSF task in IGPLAY.

Implicit representations help high-resolution forecasting. An examination of our model without implicit representations involves using a low-resolution voxel-wise cross-entropy loss for supervision and excluding the implicit representation. When assessing this model against high-resolution ground truths, we up-sample the predicted low-resolution voxel grid to a high-resolution one through a simple copying mechanism. However, this up-sampling method results in a loss of the ability to capture fine-grained geometry, leading to a significant decrease of 9.5% in Intersection over Union (IoU) and 9.9% in mean Intersection over Union (mIoU). This underscores the crucial role of implicit representations as key components in our SCSFNet.

Methods	IoU	mIoU
No implicit fields	47.0	26.4
No attention-based skip connections	54.3	31.5
No visibility grids	55.7	34.3
P4T encoder (Fan, Yang, and Kankanhalli 2021)	50.9	29.5
Ours	**56.5**	**36.3**

Table 5: **Ablation study of SCSFNet on IGPLAY for the SCSF task.** P4T means P4transformer.

Attention-based skip connections benefit spatial-temporal features. When simply incorporating point filling into voxels as skip connections, as opposed to utilizing our well-designed cross-attention modules, SCSFNet exhibits a lower IoU and mIoU by 2.2% and 4.8%, respectively. This observation demonstrates that our attention-based skip connections play a pivotal role in enhancing spatial-temporal feature learning.

Pretrained visibility grid is necessary for scene understanding. This auxiliary task for the ambient space is easier to train than the main SCSF task. We can train a relegated version of SCSFNet to obtain a high-resolution visibility voxel grid with binary labels and can achieve more than 80% in IoU. When removing the visibility grid, we see a drop in mIoU from 36.3% to 34.3% as a result of losing this critical ambient space prior.

Voxel-based encoders are more suitable for completion than point-based encoders. Rather than employing a 4D sparse convolution encoder, an alternative approach involves using a 4D point cloud encoder, such as P4Transformer (Fan, Yang, and Kankanhalli 2021), to directly leverage the point clouds. This substitution results in a 5.6% decrease in IoU and a 6.8% decrease in mIoU. This outcome substantiates the assertion that voxel-based methods are more effective for large-scale completion..

6 Conclusion

We introduce a new task of semantic complete scene forecasting (the SCSF task) from a 4D point cloud sequence and propose a new backbone SCSFNet for 4D scene understanding, equipped with hybrid geometric representations, attention-based skip connections, and visibility grids. To evaluate our method, we introduce IGPLAY and IGNAV, two large-scale 4D egocentric vision datasets with complete geometry and semantic annotations covering abundant indoor scenes and rich dynamics. Extensive experiments on our two high-quality indoor datasets and the outdoor SemanticKITTI benchmark not only confirm the significance of jointly modeling geometry forecasting and semantic completion but also demonstrate the effectiveness of our method.

References

Ali, N.; Zafar, B.; Riaz, F.; Hanif Dar, S.; Iqbal Ratyal, N.; Bashir Bajwa, K.; Kashif Iqbal, M.; and Sajid, M. 2018. A hybrid geometric spatial image representation for scene classification. *PloS one*, 13(9): e0203339.

Behley, J.; Garbade, M.; Milioto, A.; Quenzel, J.; Behnke, S.; Stachniss, C.; and Gall, J. 2019. SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. In *Proc. of the IEEE/CVF International Conf. on Computer Vision (ICCV)*.

Cai, Y.; Chen, X.; Zhang, C.; Lin, K.-Y.; Wang, X.; and Li, H. 2021. Semantic scene completion via integrating instances and scene in-the-loop. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 324–333.

Chang, A.; Dai, A.; Funkhouser, T.; Halber, M.; Niessner, M.; Savva, M.; Song, S.; Zeng, A.; and Zhang, Y. 2017. Matterport3D: Learning from RGB-D Data in Indoor Environments. *International Conference on 3D Vision (3DV)*.

Chen, X.; Lin, K.-Y.; Qian, C.; Zeng, G.; and Li, H. 2020. 3d sketch-aware semantic scene completion via semi-supervised structure prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4193–4202.

Cheng, R.; Agia, C.; Ren, Y.; Li, X.; and Bingbing, L. 2021. S3cnet: A sparse semantic scene completion network for lidar point clouds. In *Conference on Robot Learning*, 2148–2161. PMLR.

Choy, C.; Gwak, J.; and Savarese, S. 2019. 4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3075–3084.

Coumans, E.; and Bai, Y. 2016. Pybullet, a python module for physics simulation for games, robotics and machine learning.

Deng, D.; and Zakhor, A. 2020a. Temporal lidar frame prediction for autonomous driving. In *2020 International Conference on 3D Vision (3DV)*, 829–837. IEEE.

Deng, D.; and Zakhor, A. 2020b. Temporal LiDAR Frame Prediction for Autonomous Driving. In *International Conference on 3D Vision (3DV)*.

Dosovitskiy, A.; Fischer, P.; Ilg, E.; Hausser, P.; Hazirbas, C.; Golkov, V.; Van Der Smagt, P.; Cremers, D.; and Brox, T. 2015. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, 2758–2766.

Dourado, A.; Guth, F.; and de Campos, T. 2022. Data Augmented 3D Semantic Scene Completion with 2D Segmentation Priors. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 3781–3790.

Fan, H.; Yang, Y.; and Kankanhalli, M. 2021. Point 4d transformer networks for spatio-temporal modeling in point cloud videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14204–14213.

Firman, M.; Mac Aodha, O.; Julier, S.; and Brostow, G. J. 2016. Structured prediction of unobserved voxels from a single depth image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5431–5440.

Gao, H.; Xu, H.; Cai, Q.-Z.; Wang, R.; Yu, F.; and Darrell, T. 2019. Disentangling propagation and generation for video prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9006–9015.

Garbade, M.; Chen, Y.-T.; Sawatzky, J.; and Gall, J. 2019. Two stream 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 0–0.

Geiger, A.; Lenz, P.; and Urtasun, R. 2012. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Hu, A.; Cotter, F.; Mohan, N.; Gurau, C.; and Kendall, A. 2020. Probabilistic future prediction for video scene understanding. In *European Conference on Computer Vision*, 767–785. Springer.

Hu, Z.; and Wang, J. 2019. A novel adversarial inference framework for video prediction with action control. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 0–0.

Ilg, E.; Mayer, N.; Saikia, T.; Keuper, M.; Dosovitskiy, A.; and Brox, T. 2017. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2462–2470.

Khurana, T.; Hu, P.; Held, D.; and Ramanan, D. 2023. Point Cloud Forecasting as a Proxy for 4D Occupancy Forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1116–1124.

Li, J.; Liu, Y.; Gong, D.; Shi, Q.; Yuan, X.; Zhao, C.; and Reid, I. 2019. Rgbd based dimensional decomposition residual network for 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7693–7702.

Liu, L.; Gu, J.; Zaw Lin, K.; Chua, T.-S.; and Theobalt, C. 2020a. Neural sparse voxel fields. *Advances in Neural Information Processing Systems*, 33: 15651–15663.

Liu, Y.; Li, J.; Yan, Q.; Yuan, X.; Zhao, C.; Reid, I.; and Cadena, C. 2020b. 3D gated recurrent fusion for semantic scene completion. *arXiv preprint arXiv:2002.07269*.

Luc, P.; Clark, A.; Dieleman, S.; Casas, D. d. L.; Doron, Y.; Cassirer, A.; and Simonyan, K. 2020. Transformation-based adversarial video prediction on large-scale data. *arXiv preprint arXiv:2003.04035*.

Mersch, B.; Chen, X.; Behley, J.; and Stachniss, C. 2022. Self-supervised point cloud prediction using 3D spatio-temporal convolutional networks. In *Conference on Robot Learning*, 1444–1454. PMLR.

Peng, S.; Niemeyer, M.; Mescheder, L.; Pollefeys, M.; and Geiger, A. 2020. Convolutional occupancy networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, 523–540. Springer.

Rist, C. B.; Emmerichs, D.; Enzweiler, M.; and Gavrila, D. M. 2021. Semantic scene completion using local deep implicit functions on lidar data. *IEEE transactions on pattern analysis and machine intelligence*, 44(10): 7205–7218.

Roldao, L.; De Charette, R.; and Verroust-Blondet, A. 2022. 3D semantic scene completion: a survey. *International Journal of Computer Vision*, 1–28.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, 234–241. Springer.

Silberman, N.; Hoiem, D.; Kohli, P.; and Fergus, R. 2012. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, 746–760. Springer.

Song, C.; Song, J.; and Huang, Q. 2020. Hybridpose: 6d object pose estimation under hybrid representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 431–440.

Song, S.; Yu, F.; Zeng, A.; Chang, A. X.; Savva, M.; and Funkhouser, T. 2017. Semantic scene completion from a single depth image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1746–1754.

Sun, X.; Wang, S.; Wang, M.; Wang, Z.; and Liu, M. 2020. A novel coding architecture for LiDAR point cloud sequence. *IEEE Robotics and Automation Letters*, 5(4): 5637–5644.

Van Hoorick, B.; Tendulkar, P.; Surís, D.; Park, D.; Stent, S.; and Vondrick, C. 2022. Revealing Occlusions with 4D Neural Fields.

Wang, H.; and Tian, Y. 2022. Sequential Point Clouds: A Survey. *arXiv preprint arXiv:2204.09337*.

Wang, Y.; Tan, D. J.; Navab, N.; and Tombari, F. 2019. Forknet: Multi-branch volumetric semantic completion from a single depth image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8608–8617.

Wen, H.; Liu, Y.; Huang, J.; Duan, B.; and Yi, L. 2022. Point Primitive Transformer for Long-Term 4D Point Cloud Video Understanding. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIX*, 19–35. Springer.

Weng, X.; Wang, J.; Levine, S.; Kitani, K.; and Rhinehart, N. 2021. Inverting the pose forecasting pipeline with SPF2: Sequential pointcloud forecasting for sequential pose forecasting. In *Conference on robot learning*, 11–20. PMLR.

Xia, F.; Shen, W. B.; Li, C.; Kasimbeg, P.; Tchapmi, M. E.; Toshev, A.; Martín-Martín, R.; and Savarese, S. 2020. Interactive Gibson Benchmark: A Benchmark for Interactive Navigation in Cluttered Environments. *IEEE Robotics and Automation Letters*, 5(2): 713–720.

Xie, H.; Yao, H.; Zhou, S.; Mao, J.; Zhang, S.; and Sun, W. 2020. Grnet: Gridding residual network for dense point cloud completion. In *European Conference on Computer Vision*, 365–381. Springer.

Xu, Q.; Wang, W.; Ceylan, D.; Mech, R.; and Neumann, U. 2019. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. *Advances in neural information processing systems*, 32.

Yan, X.; Gao, J.; Zheng, C.; Zheng, C.; Zhang, R.; Cui, S.; and Li, Z. 2022. 2dpass: 2d priors assisted semantic segmentation on lidar point clouds. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII*, 677–695. Springer.

Zhang, J.; Wang, Y.; Long, M.; Jianmin, W.; and Philip, S. Y. 2019a. Z-order recurrent neural networks for video prediction. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, 230–235. IEEE.

Zhang, P.; Liu, W.; Lei, Y.; Lu, H.; and Yang, X. 2019b. Cascaded context pyramid for full-resolution 3D semantic scene completion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7801–7810.

Zhang, Z.; Sun, B.; Yang, H.; and Huang, Q. 2020. H3dnet: 3d object detection using hybrid geometric primitives. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*, 311–329. Springer.

Semantic Complete Scene Forecasting from a 4D Dynamic Point Cloud Sequence Technical Appendix

This document provides a list of supplemental materials to support the main paper.

- **Dataset Description.** We detail the generation process and statistics of IGPLAY and IGNAV in Section A. Multi-type data formats allow us to conduct a variety of experiments related to the SCSF task.
- **Experiment Details.** We specify the computing infrastructure used for running experiments. In order to fit other relevant models to the new SCSF task compared with SCSFNet, we need to do some modifications. We will give details about this process in Section B.
- **Additional Ablation Studies.** We provide additional ablation studies in Section C. Specifically, we compare the performance of our 4D sparse voxel-based encoder with a point-based encoder and explore the impact of data augmentation on enhancing the performance of SCSFNet.
- **More results about Movable IoU.** We elaborate further on the calculation of Movable IoU for the SCSF task in Section D. The results in Section D confirm our claim that SCSFNet can better handle dynamic objects.
- **More results about Semantic Scene Completion.** Ignoring future forecasting, we can degenerate SCSFNet to tackle SSC (Semantic Scene Completion) problem for IGPLAY and NYUCAD in the form of 3D or 4D input in Section E and get great performance.
- **Additional Visualizations.** We provide more visualizations in Section F, trying to explore more dynamic scenes for the SCSF task and prove the effectiveness of our SCSFNet.
- **Limitations and Future work.** We analyze the limitations of our work based on the existing datasets experiments in Section G. We hope that future work will try to solve these problems for the SCSF task.

A Dataset Description

A.1 iGibson

iGibson (Xia et al. 2020) is a simulation environment based on Bullet (Coumans and Bai 2016) that provides fast visual rendering and physical simulation. iGibson is equipped with fifteen fully interactive high-quality scenes, and hundreds of large 3D scenes reconstructed from real homes and offices, providing 12000+ additional interactive scenes. Some of the features of iGibson include domain randomization, integration with motion planners, and easy-to-use tools to collect human demonstrations. Based on the kinematic and extended states, iGibson 2.0 implements a list of logic predicates that indicate whether a logical state is true or false by checking conditions. The logic states implemented and checked by iGibson 2.0 such as “inside of”, “on top of”, “next to”, and so on. This system makes it more convenient to generate datasets.

A.2 Dataset Generation

To generate the simulated scenes in IGPLAY, we first specify the desired configuration of the environment in the logic language BDDL. In a typical BDDL description that we use, there is 1 floor, 2 tables, 1 bed, 2 sofas, 2 chairs, 3 floor lamps, 2 table lamps, and 1 iGibson robot. We then specify the desired logical state to make sure that everything is on the floor except table lamps which are on the table. After iGibson succeeds in creating the ideal scenario described by BDDL, we randomly discard some objects in the scene and then add two types of interactive and movable objects at random locations in the scene: toys and pops. These objects will drop on the table, on the sofa, on the chair, or just on the ground. This process was repeated and finally, we captured 1000 scenarios, each lasting 10 timesteps, with various data captured from the robot which is interacting with toys among the furniture.

While IGPLAY already exhibits many features suitable for 4D forecasting, the whole dynamics of the scene come from the viewer’s movement and interaction with the objects. To provide a new perspective of the dynamics, we provide a new dataset called IGNAV with several robots navigating in the scene. The generation process of IGNAV is similar to that of IGPLAY. In order to make IGNAV different from IGPLAY, we made some adjustments in the semantic categories. For IGNAV, we similarly use the BDDL description to generate 1 floor, 2 tables, 1 bed, 2 chairs, 3 floor lamps, 2 table lamps, and 4 iGibson robots on average. Every robot will move at a certain velocity and direction. We captured 600 scenarios, each lasting 10 timesteps. Several robots move around the scene in their own way and interact with each other, with another robot watching them.

A.3 Dataset statistics

We provide multiple data formats including RGB, normal, semantic labels, the point cloud, and the voxel-based semantic complete scene, which are shown in Figure 7.

Each scene in IGPLAY contains 10 semantic classes: ceiling, floor, wall, table, chair, bed, sofa, lamp, toy, and pop, with partial and complete occlusions happening as the viewer moves and interacts with objects.

Each scene in IGNAV contains 9 semantic classes: ceiling, floor, wall, table, chair, sofa, lamp, window/door, and robot, with partial and complete occlusions happening as iGibson robots move and interact with each other or the furniture.

In each timestep of our IGPLAY and IGNAV, we provide the 2D input image (480×640) with RGB, normal and se-

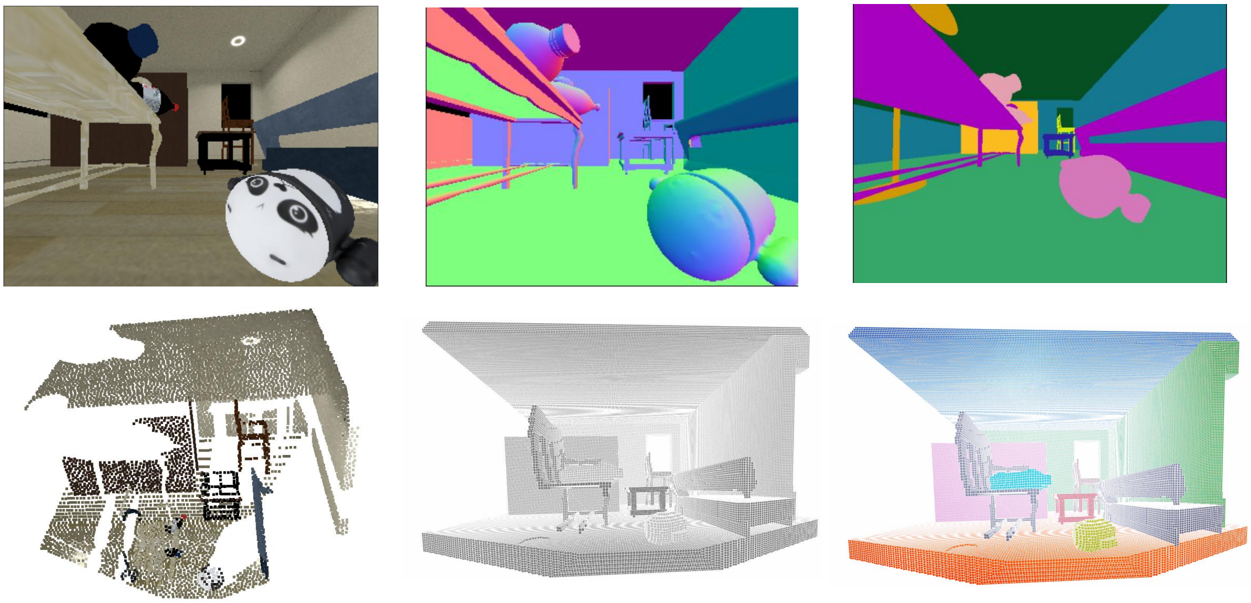

Figure 7: **Visualizations for IGPLAY.** We choose one frame in a sequence from IGPLAY. In the first row, we show the 2D RGB image, the 2D normal image, and the 2D semantic image. In the second row, we show the input 3D point cloud, the 3D voxel-based ground truth for the CSF task, and the 3D voxel-based ground truth for the SCSF task.

mantic labels, the 3D input point cloud (32768 points, Farthest Point Sampling from the point cloud generated by the 2D image), the 3D sparse voxel representation of the ground truth for the CSF task and SCSF task, and the visibility grid. The resolution of the ground truth is 240 × 144 × 240 with grid size 0.02m, To store data more efficiently, we just record the non-empty voxels. For IGPLAY and IGNAV, the dimensions of the 3D space is 4.8m horizontally, 2.88m vertically, and 4.8m in depth. We also provide the intrinsic parameters of the iGibson camera and the field of view of the agent. The train/test split is 80%/20%. We have shown typical data clips in the video we provide.

B Experiment Details

Our experimental results presented in the manuscript (like Table 1 and Table 2) use the same experimental configuration. Training a single SCSFNet model takes approximately 50 epochs in Ubuntu, with each epoch taking around 10 minutes. We utilize 2 Nvidia GeForce RTX 3090 GPUs, each with 24GB of VRAM. Our chosen optimizer is AdamW with a learning rate of 0.01, and we employ a batch size of 4. For streamlined management of our Python environment, we use Anaconda (version 1.9.0) and Vscode (version 1.75). And we use Python (version 3.8.16), Pytorch (1.8.1), MinkowskiEngine(0.5.4) and KNN-CUDA(0.2). Detailed instructions regarding our virtual Python environment will be provided once we release our code on GitHub in the near future.

B.1 TLFPAD

This baseline (Deng and Zakhor 2020b) is tailored for processing a 4-frame sequence of point clouds to predict the next frame. Hence, we can adapt this baseline to perform the (surface) point cloud forecasting task with minimal modifications. However, due to its convolutional structure requiring 4-frame input, we replicate the first frame twice to create a 4-frame point cloud sequence. Each frame in the sequence comprises 8192 points.

As TLFPAD is not capable of managing occlusions and generating comprehensive future forecasts, our approach involves an initial step of forecasting a partial point cloud using TLFPAD. In this process, TLFPAD calculates motion vectors from the last input frame to generate the predicted frame, and we ensure that the point features such as color from the last frame are assigned to the predicted frame. Subsequently, employing the point cloud with features in the predicted frame, we proceed to perform semantic scene completion using the module from our SCSFNet.

We adhere to the default configuration of this baseline and conduct training for 100 epochs. Finally, we report the optimal outcome achieved on the validation set.

B.2 Occlusion4D

The baseline (Van Hoorick et al. 2022) is designed for future scene completion. The input is a contiguous sequence of point cloud frames from a single view used to obtain an implicit representation.

In the training procedure, it samples solid points close to the target point cloud as positive instances and air points far away from the target point cloud as negative instances. Then it concatenates positive and negative instances to obtain query points and trains the network to predict properties on these points including density, semantic label and so on based on the implicit representation of the scene.

In the evaluation procedure, it doesn't have the target point cloud, so it just randomly samples query points and uses the network to predict the properties of these points. To obtain the predicted point cloud, it filters out query points with densities larger than a given threshold.

We need to modify it for the CSF and SCSF task. For instance, We use contiguous sequences of 3 point cloud frames S_0, S_1, S_2 with color as the input for IGPLAY. we convert the ground truth voxels in the next frame to a point cloud V and use farthest point sampling to sample 6144 points as our target point cloud. Due to the limitation of memory for this model, each frame contains 6144 points. We train the network to predict density on query points. In the evaluation procedure, we use $240 \times 144 \times 240$ query points, each of which represents a voxel with a side length of 0.02m. For those voxels with predicted densities larger than the density threshold, we assume they are occupied. Then we can calculate IoU for the CSF task.

B.3 ST3DCNN

This baseline (Mersch et al. 2022) utilizes a novel 3D representation composed of concatenated range images as input. It simultaneously predicts future range images by an encoder-decoder structure and assigns per-point scores indicating validity or invalidity for multiple future time steps. We refer to this approach as ST3DCNN, which we consider as the SOTA approach (without sensor extrinsics) on the KITTIOdometry dataset, in line with the description from (Khurana et al. 2023).

For SemanticKITTI, we maintain the same field of view (FOV) as in KITTIOdometry. However, we only need to forecast the point cloud in front of the vehicle. To achieve this, we adjust the range of yaw angles from $[-\pi, \pi]$ to $[-\pi/2, \pi/2]$. Additionally, we resize the range map from 2048×96 to 1024×96. For IGPLAY and IGNAV, we modify the FOV to $1/3\pi$ and resize the range map to 256×256.

It is noteworthy that the performance of ST3DCNN does not surpass that of TLFPAD. This discrepancy can be attributed to the fact that ST3DCNN's pipeline solely predicts the point cloud without incorporating additional information like color or semantic logits. In contrast, TLFPAD benefits from information derived from the last frame due to its scene-flow approach. If the ST3DCNN pipeline could accommodate supplementary aspects such as 'color range map' or 'semantic range map', it might yield enhanced results.

We stick to the default configuration of this baseline and train the model for 50 epochs. Subsequently, we select 8192 valid points from the predicted range map to generate the forecasted point cloud for the next frame. For semantic scene completion, we train a SCSFNet following the same approach as TLFPAD.

C Additional Ablation Studies

We have discussed some components of SCSFNet including the implicit field (IF), attention-based skip connections (AS), visibility grids (VG), and the 4D sparse voxel-based encoder compared with point-based methods in the manuscript. Furthermore, we did more ablation studies on IGPLAY considering other factors, as shown in Table 6.

Methods	IoU	mIoU
No IF	47.0	26.4
No AS	54.3	31.5
No VG	55.7	34.3
P4T encoder (Fan, Yang, and Kankanhalli 2021)	50.9	29.5
Full Supervision	56.3	36.3
use DA	56.9	36.5
Ours	56.5	36.3

Table 6: **Ablation study of SCSFNet on IGPLAY for the SCSF task.** IF means implicit fields. AS means attention-based skip connections. VG means visibility grids. P4T means P4transformer. DA means data augmentation.

Supervising implicit functions through on-the-fly random samples is both effective and efficient. During the training phase, we employ random sampling of 10^5 points from all the central coordinates of ground truth voxels. Another choice is using the entire ground truth, which entails calculating the loss based on $240 \times 144 \times 240 = 8294400$ voxels for a single data point in IGPLAY. This method, however, proves time-consuming and yields slightly inferior results (IoU drops from 56.5% to 56.3%). This substantiates that employing a random subset of 10^5 voxels on the fly is significantly faster while maintaining effectiveness.

Data augmentation improves the performance of SCSFNet. Initially, we used a division scheme that selected $(0, 1, 2)$, $(3, 4, 5)$, and $(6, 7, 8)$ as input indices, with (3), (6), and (9) as output indices for a 4D point cloud sequence of 10 frames in IGPLAY. This approach ensured no input overlap. However, to maximize data utilization, we adopted a more effective strategy by selecting $(0, 1, 2), (1, 2, 3), \ldots, (6, 7, 8)$ as input indices. This augmentation technique results in training the network three additional times for a single sequence in IGPLAY. This approach yielded a slight improvement, boosting IoU from 56.5% to 56.9% and mIoU from 36.3% to 36.5% for our SCSFNet.

D More results about Movable IoU

In section 5.2 of the manuscript, we compared results between movable and static classes and reported them in Table 2 in the manuscript. A class is labeled as movable if the corresponding objects can move, such as robots in IGNAV or cars in SemanticKITTI. We present a comprehensive breakdown of IoU values for each category and offer an in-depth analysis of the dynamics within each dataset.

In the IGPLAY scenario, the agent's observations encompass the entire scene, including interactions with movable toys and objects popping up from the ground. Hence, accurate forecasting on IGPLAY necessitates special attention to these elements. As shown in Table 7, our SCSFNet significantly outperforms TLFPAD in forecasting the semantic complete scene for dynamic objects like "toy" and "pop". Specifically, our SCSFNet demonstrates a remark-

	Every Category										Avg.		
Methods	ceil.	floor	wall	table	chair	bed	sofa	toy	pop	lamp	IoU(M)	IoU(S)	mIoU
TLFPAD	64.4	55.7	31.2	8.6	3.1	42.1	25.1	10.3	13.4	4.5	11.9	29.3	25.8
ST3DCNN	65.4	62.9	32.9	9.5	4.3	39.0	19.2	6.7	11.9	3.2	9.3	29.6	25.5
SCSFNet	70.7	65.0	37.9	20.7	12.0	53.4	32.8	24.2	32.1	14.6	**28.2**	**38.4**	**36.3**

Table 7: **SCSF Results on dynamics of IGPLAY**. We list IoU of each specific category (10 classes). “toy” and “pop” in blue are movable classes. IoU(M) in Avg. means Average Movable IoU. IoU(S) in Avg. means Average Static IoU.

	Every Category									Avg.		
Methods	ceil.	floor	wall	table	chair	window	sofa	robot	lamp	IoU(M)	IoU(S)	mIoU
TLFPAD	78.1	65.0	42.6	1.7	1.5	19.1	16.9	2.1	5.8	2.1	28.8	25.9
ST3DCNN	74.2	70.1	51.6	8.4	7.1	12.5	7.1	0.0	3.5	0.0	29.3	26.1
SCSFNet	83.6	72.5	58.5	14.8	24.2	37.4	35.8	18.3	14.0	**18.3**	**42.6**	**39.9**

Table 8: **SCSF Results on dynamics of IGNAV**. We list IoU of each specific category (9 classes). “robot” in blue is the movable class.

	Every Category																			Avg.		
Methods	car	bicycle	motorcycle	truck	other-vehicle	person	bicyclist	motorcyclist	road	parking	sidewalk	other-ground	building	fence	vegetation	trunk	terrain	pole	traffic-sign	IoU(M)	IoU(S)	mIoU
TLFPAD	24.7	0.0	0.6	2.3	2.1	0.1	0.0	0.0	37.4	8.3	20.0	0.1	17.8	5.8	23.4	5.2	24.0	6.8	2.2	4.9	11.7	9.5
ST3DCNN	23.8	0.2	0.9	2.3	1.9	0.3	0.2	0.0	51.3	6.6	25.1	0.0	14.2	4.4	20.6	1.8	29.3	0.1	0.1	4.8	11.9	9.6
SCSFNet	29.6	4.8	11.2	27.7	13.9	2.8	0.1	0.0	41.3	18.9	23.9	1.0	22.2	10.5	27.5	15.3	27.3	18.1	10.0	**12.4**	**17.8**	**16.1**

Table 9: **SCSF Results on dynamics of SemanticKITTI**. We list IoU of each specific category (19 classes). “car”, “truck”, “other-vehicle”, “person”, “bicyclist”, and “motorcyclist” in blue are the movable classes.

able 16.3% enhancement in Movable IoU and a notable 9.1% increase in Static IoU compared to TLFPAD.

In the IGNAV scenario, the mobility of robots adds complexity to predicting their trajectories, interactions, and occlusions. As depicted in Table 8, TLFPAD faces challenges, resulting in a significant 16.2% reduction in Movable IoU and a noteworthy 13.8% decrease in Static IoU.

For SemanticKITTI, classes such as “car” that encompass moving objects with annotations like “moving-car” from SemanticKITTI are considered movable classes. The results in Table 9 highlight that SCSFNet yields notable enhancements, with a 7.5% boost in Movable IoU and a commendable 6.1% improvement in Static IoU.

Across all these outcomes, a clear pattern emerges: in contrast to static objects, SCSFNet excels in addressing dynamic objects, a capability that holds paramount significance for accurate future forecasting.

E Semantic Scene Completion tasks for SCSFNet

Our paper proposed the SCSF task, which mainly focuses on future forecasting. If we do not forecast and only focus on the current frame (the last given frame), degenerate cases for CSF (Complete Scene Forecasting) and SCSF (Semantic Complete Scene Forecasting) become SC (Scene Completion) and SSC (Semantic Scene Completion). Our SCSFNet can solve the SSC problem using the same pipeline for the SCSF task in different dimensions.

3D SSC. We treat every frame in a 4D point cloud sequence as a single point cloud input ignoring the connections between different frames. In this way, we only feed one frame into the model and supervise this single frame.

4D SSC. We consider that connections between different frames in a 4D point cloud sequence are important to gain more information in scene understanding and completion, and our SCSFNet is capable of processing multiple frames simultaneously. In this way, we feed 3 consecutive frames into the model, generate three high-resolution voxel grids to represent the completion results, and then supervise these 3 frames simultaneously. We need a little modification of the network architecture in this setting.

The results in IGPLAY are shown in Table 10. For the SSC problem, we mainly use IoU and mIoU to evaluate the completion with semantic labels. SCSFNet (4D) gives a 0.2% improvement in IoU and a 2.5% improvement in mIoU compared with SCSFNet (3D).

In particular, SCSFNet (4D) has an obvious forecasting advantage in the “toy” class. In IGPLAY, the agent interacts with these objects, thus they have strong motion continuity between successive frames. If multiple frames are taken together as the input, SCSFNet can model them jointly to gain more insights into the scene dynamics.

Similarly, we try our SCSFNet (3D) in NYU-

	Scene Completion (SC)			Semantic Scene Completion (SSC)										
Methods	prec.	recall	IoU	ceil.	floor	wall	table	chair	bed	sofa	toy	pop	lamp	avg.
SCSFNet(3D)	88.7	**66.3**	61.1	**68.3**	51.2	**64.5**	30.5	**21.2**	54.9	42.2	27.7	41.1	26.9	42.8
SCSFNet(4D)	**90.7**	65.4	**61.3**	64.9	**52.8**	63.4	**32.0**	**21.2**	**60.3**	**42.7**	**42.4**	**45.6**	**27.5**	**45.3**

Table 10: **SSC Results on IGPLAY dataset**. We show the results of SCSFNet in 3D and SCSFNet in 4D for SSC problem. For better comparison such as movable objects, we list IoU of each specific category.

	Scene Completion (SC)			Semantic Scene Completion (SSC)											
Methods	prec.	recall	IoU	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	avg.
SSCNet (Song et al. 2017)	75.4	**96.3**	73.2	32.5	92.6	40.2	8.9	33.9	57.0	59.5	28.3	8.1	44.8	25.1	40.0
DDRNet (Li et al. 2019)	88.7	88.5	79.4	54.1	91.5	56.4	14.9	37.0	55.7	51.0	28.8	9.2	44.1	27.8	42.8
TS3D (Garbade et al. 2019)	-	-	76.1	25.9	93.8	48.9	33.4	31.2	66.1	56.4	31.6	38.5	51.4	30.8	46.2
CCPNet (Zhang et al. 2019b)	91.3	92.6	82.4	56.2	**94.6**	58.7	**35.1**	44.8	68.6	65.3	37.6	35.5	53.1	35.2	53.2
Sketch (Chen et al. 2020)	90.6	92.2	**84.2**	59.7	94.3	64.3	32.6	51.7	72.0	**68.7**	**45.9**	19.0	**60.5**	**38.5**	55.2
Ours (SCSFNet)	**92.8**	89.1	83.3	**60.6**	94.3	**65.5**	32.7	**55.3**	**74.5**	66.7	43.3	**42.4**	58.5	**38.5**	**57.5**

Table 11: **SSC Results on NYUCAD dataset (Silberman et al. 2012)**. We compare SCSFNet with models using the same straightforward pipeline and ignore models using iterative pipeline.

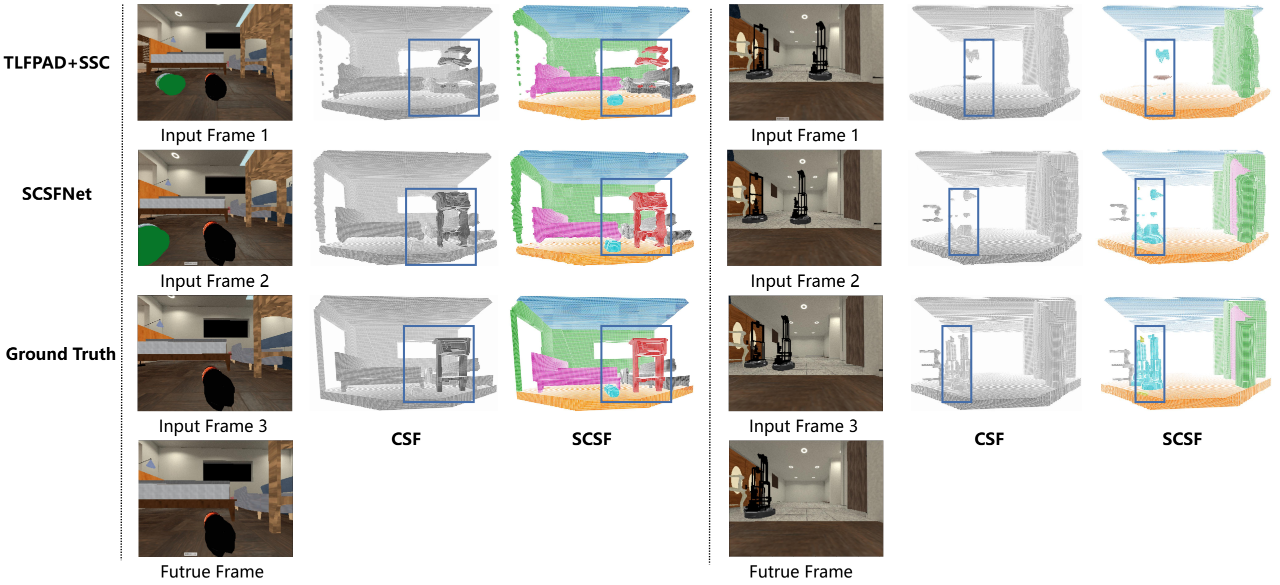

Figure 8: **Visualizations for our synthetic indoor datasets.** The sequence on the left is taken from IGPLAY. The agent saw a bottle of soda and a bottle of cola at first and approached the bottle of cola beside the table. The sequence on the right is taken from IGNAV. The agent is watching a robot going toward another stationary robot. SCSFNet can predict the shape of the bottle, the table, and the robots better under the change of perspective and dynamic objects.

CAD(Silberman et al. 2012), one of the most famous SSC datasets. We compare our model to other state-of-the-art SSC methods using the same iterative pipeline (without additional instance labels) in Table 11. We can find that our SCSFNet is a generalizable model not only for future forecasting but also for classic semantic completion.

F Additional Visualizations

We also offer enhanced visualizations of both the CSF task and the SCSF task across three distinct datasets. With IGPLAY and IGNAV shown in Figure 8, our aim is to present a more dynamic visualization and showcase the capability of SCSFNet for semantic complete forecasting. As for SemanticKITTI (Behley et al. 2019) shown in Figure 9, we strive to demonstrate the versatility of our SCSFNet, emphasizing its applicability in real-world scenarios.

Our datasets and detailed instructions for access will be made available to the public in the near future. And we will also provide detailed code for data visualization using the raw datasets IGPLAY and IGNAV.

G Limitations and Future work

In our work, we require dynamic scenes that can support completion and forecasting at the same time. Given the lim-

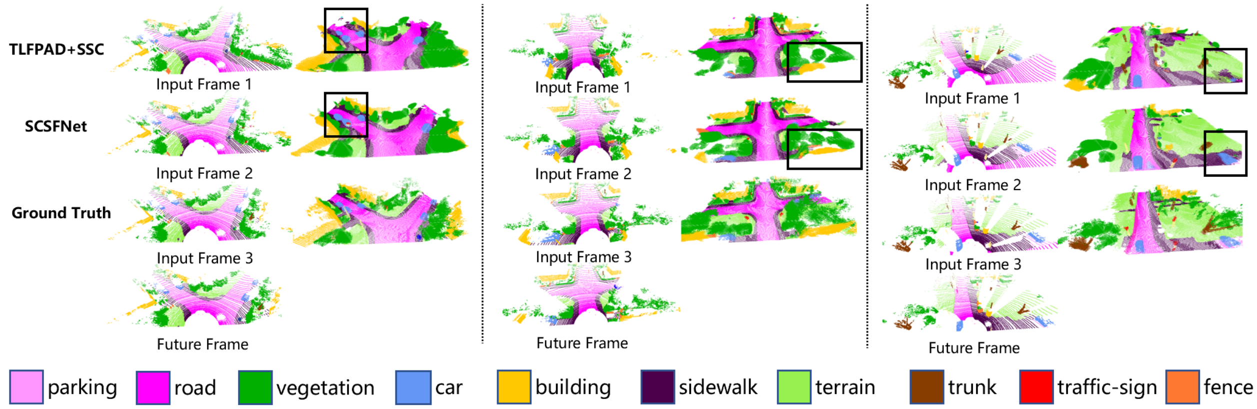

Figure 9: **Visualizations for SemanticKITTI.** We provide three groups of visualizations. Each group consists of two columns: the first column represents the three past frames of point clouds and the future frame, and the second column signifies SCSF predictions by different methods. We do not show the results for the CSF task because it's hard to distinguish objects without semantics. Although SemanticKITTI only provides raw point cloud data, we can use the semantic segmentation logits predicted by 2DPASS (Yan et al. 2022).

ited options for real-world datasets meeting these criteria, we find SemanticKITTI to be the only suitable choice, which consists of large-scale 4D point cloud sequences with completion and semantic annotations. To further demonstrate the effectiveness of our method in the indoor setting, we generate 2 large-scale indoor datasets. We hope to extend the evaluation of our approach to suitable real-world datasets released in the future.

Additionally, in our experiments, we only performed relatively short-term forecasting due to the inherent challenges of the SCSF task. This limitation hinders the model's ability to provide an accurate understanding of the entire scene, particularly focusing on dynamic objects, in the long term. In the future, we aspire to see more efficient approaches that can handle long-term SCSF tasks with minimal computational overhead.