

# Forgetting-Free Incremental Panoptic Lifting by Maximum-Visibility Viewpoint Selection

Akira Kohjin<sup>1,2</sup> Motoharu Sonogashira<sup>2</sup> Masaaki Iiyama<sup>1</sup> Yasutomo Kawanishi<sup>2</sup>  
<sup>1</sup>Shiga University, <sup>2</sup>RIKEN

s7025104@st.shiga-u.ac.jp, iiyama@iiyama-lab.org,  
 {motoharu.sonogashira, yasutomo.kawanishi}@riken.jp

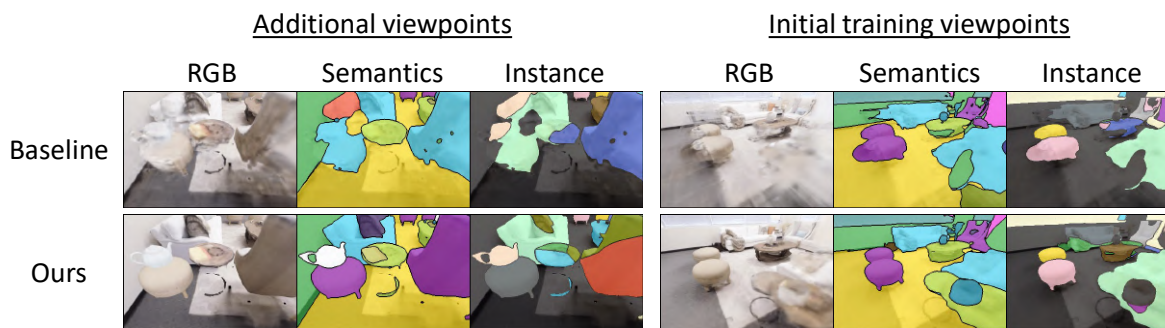


Figure 1. Comparison between the baseline and our proposed method. Our incremental approach enables adaptation to additional viewpoints, which are acquired sequentially, while effectively preventing catastrophic forgetting of the initial training viewpoints, which have already been obtained, with low computational cost.

## Abstract

We propose an incremental learning method for Panoptic Lifting, a novel view synthesis technique that leverages both RGB images and panoptic segmentation masks to represent a 3D scene. While Panoptic Lifting is valuable in applications such as VR, autonomous vehicles, and robotics, it faces a challenge when observations are limited (e.g., due to occlusions), as retraining from scratch becomes computationally expensive. Our approach incrementally updates the model to achieve high accuracy with reduced computational cost. To prevent catastrophic forgetting, a phenomenon where previous learned knowledge is lost during model updates, we introduce a viewpoint selection algorithm that solves a maximum coverage problem to identify a subset of viewpoints that maximizes the visibility of the scene. Experimental results demonstrate a 12% reduction in computation time compared to the naïve approach, while effectively preventing catastrophic forgetting.

## 1. Introduction

Understanding the surrounding environment through comprehensive 3D modeling based on sensor-acquired image

data remains a fundamental challenge in fields such as VR, autonomous vehicles, and robotics. Various applications benefit from novel view synthesis, and more recently, Neural Radiance Fields (NeRF) [19] have emerged as a powerful method, generating photorealistic results by modeling volumetric scene representations.

Beyond RGB-based representation, Panoptic Lifting [27] extends novel view synthesis to panoptic segmentation [14], providing not only RGB images but also semantic and instance masks. However, a major limitation of Panoptic Lifting, as well as other NeRF-based methods, is that their performance depends on the quantity and quality of observation data for model training. When environmental conditions, such as occlusion and limitations in the observation process, result in missing viewpoints, the accuracy of a model trained solely on the initial set of observations degrades, particularly in areas where observations are insufficient.

A naïve approach to address this issue is to acquire additional viewpoint data and rebuild the model from scratch with the expanded dataset. However, this results in high computational costs and wastes the previously trained model, making it impractical for recognizing across the entire scene. Incremental learning offers a more efficient al-

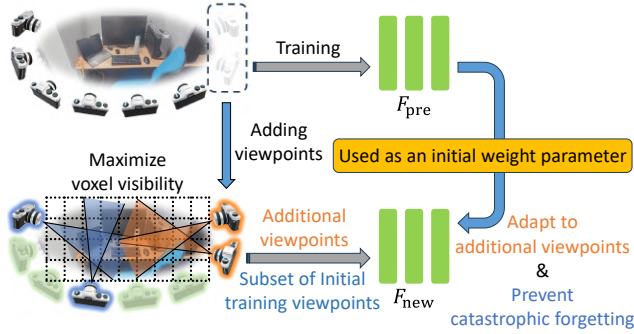


Figure 2. Our forgetting-free incremental Panoptic Lifting updates the model using only initial viewpoints (left side of the desk) to recognize the entire scene, including additional viewpoints (right side), with sufficient accuracy at low computational cost. Learning the voxel observations represented by  $\triangleleft$  contributes adaptation to the additional viewpoints, while learning those represented by  $\triangleleft$ , selected from the initial viewpoints maximizing voxel visibility of the scene, prevents catastrophic forgetting.

ternative by updating the model without discarding previous knowledge. However, this approach introduces the phenomenon of catastrophic forgetting, where the knowledge implicitly retained in the previously trained model is lost as the model trains on the additional viewpoint data (Figure 3).

To overcome these difficulties, we propose a forgetting-free incremental learning framework for Panoptic Lifting that effectively prevents catastrophic forgetting while reducing computational cost required for model training. Our approach introduces a viewpoint selection algorithm that prioritizes selecting a subset of initial training viewpoints to maximize the coverage of the visible regions of the scene. This algorithm is formulated as a maximum coverage problem targeting the visibility of voxel grids in a 3D scene and is solved using a greedy algorithm. By leveraging this selected subset along with the additional viewpoints for training, the model is updated efficiently, maintaining its ability to recognize the entire scene. To summarize, our main contributions are as follows:

- We propose an incremental learning method for Panoptic Lifting, i.e., novel view synthesis for RGB images and panoptic segmentation masks.
- We introduce an efficient viewpoint selection algorithm that prevents catastrophic forgetting by maximizing the coverage of visible regions in the 3D scene.

## 2. Related Work

### 2.1. Panoptic Lifting

Panoptic Lifting is a novel view synthesis method that generates not only RGB images from novel viewpoints but also



Figure 3. While a model trained only on the initial viewpoints accurately synthesizes segmentation masks for those viewpoints, incremental learning with additional viewpoints causes catastrophic forgetting, overwriting previous knowledge, introducing noise and reducing accuracy.

panoptic segmentation masks, using RGB images captured from various viewpoints of the scene, their corresponding camera pose and machine-generated panoptic segmentation [14] masks as input. Here, panoptic segmentation is a combination of semantic segmentation and instance segmentation, and it is defined as a comprehensive framework for image segmentation tasks. Semantic segmentation is a task of classifying each pixel in an image according to its object class, providing an overall labeling of the scene’s content. In contrast, instance segmentation assigns unique identifiers to each individual object within the image, distinguishing between different instances of the same class. As the output of panoptic segmentation, masks are obtained for objects in the scene, referred to as “things”, as well as for background elements, referred to as “stuff”.

Panoptic Lifting models a 3D scene using a framework that takes spatial positions and view directions as input, similar to NeRF. In addition to predicting view-dependent color and volumetric density at arbitrary 3D positions, it also estimates the probability distributions of semantic classes and instance identifiers. For representing color and density, Panoptic Lifting employs TensorRF [1], representing a 3D scene in a tensor format and learning the parameters of its low-rank decomposition to reduce memory consumption and computational cost compared to conventional MLP-based methods [3, 19, 23] while simultaneously enabling high-quality novel view synthesis. Additionally, separate MLPs are used for predicting semantic classes and instance identifiers, and the model is trained using pseudo labels generated by the Mask2Former [5] model, which has been pre-trained on the Microsoft COCO dataset [28].

Among the novel view synthesis methods of RGB images and segmentation masks based on NeRF [4, 15, 27], Panoptic Lifting currently represents the state-of-the-art in accuracy. Furthermore, Panoptic Lifting can generate

3D-consistent panoptic segmentation masks of the scene from independent 2D panoptic segmentation masks provided for each viewpoint as input. In cases where conventional panoptic segmentation frameworks are applied independently to 2D images captured from different viewpoints of a 3D scene, regions corresponding to the same “thing” or “stuff” may be assigned different semantic classes or instance identifiers, resulting in inconsistencies in recognition across viewpoints of the scene. In contrast, in Panoptic Lifting, the segmentation masks utilized as input from multiple viewpoints are integrated, allowing the inference of semantic classes and instance identifiers based on 3D positions and view directions. As a result, the rendered segmentation masks are more likely to assign consistently the same semantic class and instance identifier to the same object across different viewpoints, making it a more effective approach for consistent 3D scene recognition from observations across multiple viewpoints. Due to the advantages in novel view synthesis methods of RGB images and segmentation masks, we integrate our proposed approach with Panoptic Lifting.

The overall procedure for recognizing the surrounding environment as a 3D scene using Panoptic Lifting is outlined as follows.

1. Capture images from various viewpoints across the scene and estimating their camera poses.
2. Estimate panoptic segmentation masks for each image using a pre-trained Mask2Former model.
3. Train the model using the set of images, panoptic segmentation masks, and their camera poses.
4. Perform novel view synthesis with the model.

The process of acquiring the image to be used as the model input differs depending on the application. For instance, in a home monitoring system, an autonomous robot must understand the indoor environment as a unified 3D scene. Achieving this comprehensive understanding requires capturing a wide range of images to train a model that represents the entire space. However, a single traversal of the environment does not guarantee a complete set of observations, leaving some areas unobserved. Training the model on an image set lacking specific viewpoints of the scene is expected to degrade its inference accuracy in those viewpoints. In a naïve approach, additional observation data for lacking viewpoints need to be captured, and the model must be retrained from scratch using the entire set of images. This process discards the initially trained model and leads to an increase in computational cost as the training dataset grows. To overcome these challenges, we introduce an incremental learning approach into Panoptic Lifting.

## 2.2. Incremental Learning

Incremental learning is an approach that updates the model with sequentially acquired additional observation data, enabling it to adapt to the additional data while preserving previously learned knowledge. The model is updated by initializing it with pre-trained weight parameters learned from previous training data, followed by further training on the additional data. This approach allows for building a sufficiently accurate model with a lower computational cost compared to training a model from scratch using both previously learned data and additional data. However, a critical challenge in incremental learning is the phenomenon known as catastrophic forgetting. While the knowledge of previously learned data is implicitly retained in the weight parameters, intensive training on additional data overwrites it, which means that the model “forgets” the previously learned knowledge. As a result, inference accuracy on previously learned data is degraded.

To prevent catastrophic forgetting, several methods have been developed, including Experience Replay [2], which explicitly retrains a subset of previously learned data, Generative Replay [11], which explicitly learns data generated by the pre-trained model or a generative model, and approaches that incorporate knowledge distillation loss [10] from the pre-trained model during training. In particular, the Experience Replay enables noise-free incremental learning by directly utilizing previously learned observation data. For this reason, our proposed method adopts this strategy. The loss function  $L_{\text{INC}}(\theta)$  during incremental learning by Experience Replay, which uses both additional observation data and a subset of previously learned data, can be expressed as follows.

$$L_{\text{INC}}(\theta) = L(\theta, D_{\text{selected}}) + L(\theta, D_{\text{INC}}), \quad (1)$$

where  $\theta$  represents the weight parameters to be updated, initialized with the weight parameters of the previously trained model,  $L$  is the loss function used for learning,  $D_{\text{selected}}$  is a subset of the previously training data, and  $D_{\text{INC}}$  represents the additionally provided data. By explicitly relearning the knowledge from the previous training data, catastrophic forgetting can potentially be prevented.

Several methods [6, 12, 16, 18, 24] have been proposed for incremental learning in novel view synthesis, considering the prevention of catastrophic forgetting. Methods that prevent catastrophic forgetting through knowledge distillation [10], using a pre-trained model as a teacher, have been proposed [12, 18, 24], along with approaches that assume dynamic scenes and provide benchmark datasets [6]. Additionally, more practical frameworks that estimate the camera pose of newly acquired images [16] have also been introduced. These existing incremental learning approaches have demonstrated their effectiveness in novel view synthesis, enabling efficient learning of 3D scenes in practi-

cal scenarios. Building on this, our method extends incremental learning to include RGB image synthesis as well as panoptic segmentation mask synthesis, enabling more comprehensive scene understanding.

### 3. Forgetting-Free Incremental Panoptic Lifting

#### 3.1. Incremental Learning for Panoptic Lifting

Given a pre-trained Panoptic Lifting model that has been trained on incomplete observation data for certain viewpoints in the scene and newly acquired observation data from additional viewpoints, incremental learning can be performed using the weight parameters of the pre-trained model as the initialization. To achieve forgetting-free incremental learning for Panoptic Lifting by Experience Replay, it is crucial to select the most significant subset of the observation data from the initial training viewpoints as input, ensuring effective novel view synthesis of RGB images and panoptic segmentation masks. Through this approach, the model is expected to achieve comparable accuracy while requiring only a minimal amount of training data, thereby reducing computational costs compared to a naïve approach.

#### 3.2. Viewpoint Selection Algorithm for Maximizing Observed Voxels of the Scene

While various criteria can be considered for selecting the observation data of the initial training viewpoints, in this work, we introduce a viewpoint selection algorithm that maximizes the number of observed voxels in the scene for forgetting-free incremental Panoptic Lifting. This algorithm is based on the idea that selecting viewpoints covering a larger visible area of the target scene enables more efficient 3D scene reconstruction. To achieve this, we choose the subset of pre-training observation data from viewpoints that cover the largest number of voxels in the 3D space, and use them for Experience Replay in the incremental learning step. The 3D scene is modeled as a voxel grid, with the voxel coordinates corresponding to object surfaces. Among the candidate viewpoints from the initial training viewpoints, those that capture the largest number of voxels not observable from the additional or previously selected viewpoints are chosen. Since voxel coordinates are computed and stored only for voxels corresponding to object surfaces, rather than for the entire voxel grid of the scene, this approach helps reduce memory usage and allows for efficient parallel computation. This viewpoint selection is formulated as a maximum coverage problem, where the number of selectable viewpoints is constrained to maximize the observed voxels. This problem is addressed using the greedy algorithm, which yields an approximate solution for determining the set of observation data used for incremental learning.

---

#### Algorithm 1 Greedy Viewpoint Selection for Maximizing Voxel Visibility

---

**Require:** A set of initial training viewpoints identified by  $P = \{i \mid 1 \leq i \leq t, i \in \mathbb{Z}\}$ ,  
a set of additional viewpoints identified by  $I = \{i \mid t + 1 \leq i \leq n, i \in \mathbb{Z}\}$ ,  
a subset of selected viewpoints  $S(\subset P) \leftarrow \emptyset$ ,  
the number of viewpoints to be selected  $m$

- 1: Calculate the voxel coordinates of each pixel for  $P, I \triangleleft$  (Equation 2)
- 2: Define voxel subsets corresponding to each viewpoint  $V_1, V_2, \dots, V_n(\subset V = \{j \mid 1 \leq j \leq g, j \in \mathbb{Z}\}) \triangleleft$  (Equation 4)
- 3: **for**  $i = 1$  **to**  $m$  **do**
- 4:   Update voxels not covered by  $I, S \triangleleft$  (Equation 5)
- 5:   Select the largest voxel subset  $V_k \triangleleft$  (Equation 6)
- 6:   Transfer the viewpoint  $k$  from set  $P$  to set  $S$
- 7: **end for**
- 8: **return** Selected viewpoints  $S$

---

In Algorithm 1, the flow of our algorithm is shown.

#### 3.2.1. Calculation of voxel coordinates of object surfaces in images

The 3D scene is modeled as a voxel grid with  $s_x \times s_y \times s_z$  voxels. The  $\mathbf{s} = (s_x, s_y, s_z)$  is the number of voxels along each axis in the voxel coordinate system, and  $\mathbf{w}_{\min}$  and  $\mathbf{w}_{\max}$  are vectors whose elements represent the minimum and maximum values of each axis in the world coordinate system of the 3D scene.

Let  $P = \{i \mid 1 \leq i \leq t, i \in \mathbb{Z}\}$  and  $I = \{i \mid t + 1 \leq i \leq n, i \in \mathbb{Z}\}$  be the set of initial training viewpoints and the set of additional training viewpoints, respectively. For each pixel in the images observed from  $P$  and  $I$ , the voxel that corresponds to the pixel is identified. Let  $\mathbf{w}_{h,w}$  be the world coordinate vector of the voxel corresponding to a pixel located at image coordinates  $(h, w)$ . The voxel coordinate vector  $\mathbf{v}_{h,w}$  is calculated by Equation 2.

$$\mathbf{v}_{h,w} = \lfloor \mathbf{s} \odot (\mathbf{w}_{h,w} - \mathbf{w}_{\min}) \oslash (\mathbf{w}_{\max} - \mathbf{w}_{\min}) \rfloor, \quad (2)$$

where,  $\lfloor \mathbf{x} \rfloor$  is an element-wise floor function to a vector  $\mathbf{x}$ , while  $\odot$  and  $\oslash$  are operators that perform element-wise multiplication and division, respectively, for vectors of the same size.

In this study, the camera parameters and depth used to compute  $\mathbf{w}_{h,w}$  are assumed to be pre-acquired as part of the dataset used in the experiments. However, for a more practical use case, it would be advisable to use depth maps estimated using COLMAP [26] or those obtained from the output of a Panoptic Lifting model pre-trained during the initial observation. In the experiments, the ratio of the number of voxels  $\mathbf{s}$  along each axis in the voxel coordinate sys-

tem is set to match the ratio of the range of each axis in the world coordinate system,  $\mathbf{w}_{\max} - \mathbf{w}_{\min}$ , with the number of voxels along the shortest axis fixed at 100. This enables the identification of voxels in  $V = \{j \mid 1 \leq j \leq g, j \in \mathbb{Z}\}$  observed by viewpoints in  $P \cup I$ , where  $g$  is the total number of voxels observed from  $P \cup I$ .

### 3.2.2. Maximization of the number of observed voxels by maximum coverage problem

Selecting the initial training viewpoints that observes the largest number of voxels in the space can be formulated as a maximum coverage problem. Maximum coverage problem involves selecting at most  $m$  subsets from a collection of multiple subsets to cover as many elements as possible, which can be expressed as:

$$\max_X \left| \bigcup_{k \in X} V_k \right| \text{ subject to } \begin{cases} X \subseteq \{i \mid 1 \leq i \leq t, i \in \mathbb{Z}\}, \\ |X| \leq m. \end{cases} \quad (3)$$

In our proposed method, when there are  $t$  candidate initial training viewpoints in  $P$ ,  $V_k$  represents the subset of voxels observed from the  $k$ -th viewpoint, and  $m$  corresponds to the number of viewpoints to be selected. Maximum coverage problem is known to be NP-hard [21], and approximate solutions can be obtained using methods such as greedy algorithm.

### 3.2.3. Selection of initial training viewpoints using a greedy algorithm

In our proposed method, the selection of the initial training viewpoints is performed by solving maximum coverage problem, formulated in the previous section, using the greedy algorithm. Based on each voxel observed by the initial training and additional viewpoints, identified using Eq. 2, a binary matrix  $\mathbf{M}$  is initialized using Eq. 4. In this matrix, the rows correspond to viewpoint identifiers, and the columns correspond to voxel identifiers.

$$\mathbf{M}_{i,j}^{(0)} = \begin{cases} 1 & \text{if } i \in (P \cup I) \text{ observes } j \in V \\ 0 & \text{otherwise} \end{cases}. \quad (4)$$

The size of the matrix is  $n \times g$ , where the row components correspond to the subset of the voxels  $V_i$  observed from each viewpoint.

The initial training viewpoints used for incremental learning as input are determined through an iterative selection process performed  $m$  times from the subsets of voxels defined in matrix  $\mathbf{M}$  for each viewpoint, where this iterative selection process follows a greedy algorithm.. First, the voxels that are not covered by the observations from the additional viewpoint set  $I$  and the selected viewpoint set  $S$  are updated using Eq. 5.

$$\mathbf{M}_{i,j}^{(a)} = \begin{cases} 0 & \text{if } j \in \bigcup_{i \in I \cup S} V_i \\ \mathbf{M}_{i,j}^{(a-1)} & \text{otherwise} \end{cases}, \quad (5)$$

where  $\mathbf{M}_{i,j}^{(a)}$  is the value of  $\mathbf{M}_{i,j}$  at the  $a$ -th iteration ( $1 \leq a \leq m, a \in \mathbb{Z}$ ). For the union of the voxel subsets observed by the viewpoints in  $I$  and  $S$ , the columns corresponding to the identifiers of these voxels are set to zero. This ensures that the voxels observed by the viewpoints already selected for incremental learning are excluded from consideration in the subsequent viewpoint selection process. In the first iteration of the selection process,  $S$  is an empty set, meaning that no viewpoints have been selected yet.

Next, for the updated matrix  $\mathbf{M}^{(a)}$ , the viewpoint  $k$  that observes the largest number of voxels among the candidate initial training viewpoints is selected using Equation 6. The subset of voxels subset  $V_k$  is then selected, and the selected viewpoint  $k$  is transferred from the initial training viewpoint set  $P$  to the selected viewpoint set  $S$ .

$$k = \arg \max_i \left( \sum_j \mathbf{M}_{i,j} \right). \quad (6)$$

By  $m$  times iterating the viewpoint selection process defined by Equations 5 and 6,  $m$  viewpoints are selected from the initial training viewpoints. The viewpoint set  $I \cup S$  are then used for incremental learning as input.

## 4. Experiments

### 4.1. Dataset

We evaluated our proposed method using the Replica [13] indoor environment dataset. The Replica dataset is a high-quality synthetic dataset of various indoor scenes, and can generate data from arbitrary viewpoints by rendering RGB images, depth maps, semantic masks, and instance masks from 3D mesh data using the Habitat [25] simulator. In this experiment, a more complex indoor environment was reconstructed by adding 3D mesh data of various objects from the Google Scanned Objects datasets [9], which is publicly available on the Gazebo [20] platform, to the original Replica dataset.

Specifically, by placing additional objects such as chairs and desks into the indoor environment of ‘‘room 0’’ in the Replica dataset, we created an environment in which the initial observation data set had occlusions that made observation difficult, and we reproduced a situation in which the need for additional observations became clear. For the training observations, multiple camera sequences were randomly configured and rendered, simulating a scenario in which a person or robot capturing images while moving through the indoor environment. In contrast, for the evaluation observations, camera positions and directions were randomly set to enable evaluation from arbitrary viewpoints within the environment. We used 947 viewpoints for training and 200 viewpoints for evaluation.

## 4.2. Metrics

The evaluation metrics used in the experiments were PSNR, mIoU, and  $PQ^{\text{scene}}$ , which were also adopted in Panoptic Lifting [27]. PSNR is a metric for evaluating the quality of generated RGB images, while mIoU measures the accuracy of multi-class classification in semantic segmentation. Furthermore,  $PQ^{\text{scene}}$  is a modified version of the Panoptic Quality (PQ) [14] metric, designed to assess the 3D consistency of class and instance identifiers within a scene.

## 4.3. Experiment Settings

In our experiments, the dataset was separated into the initial and additional training viewpoints as stated in Section 4.1. To evaluate the effectiveness of our incremental learning method, we compared the following three types of models in the initial and additional training viewpoints.

- **PRE model** is a model trained from scratch using only the initial training viewpoints.
- **FULL model** is a naïve method model trained from scratch using both the initial training viewpoints and the additional viewpoints.
- **INC model** is our incremental learning model, initialized with the weight parameters of the *PRE* model and trained on the additional viewpoints along with a subset of the initial training viewpoints selected by our voxel-based method, which maximizes the number of observed voxels. Multiple models are constructed based on the size of the selected subset from the initial viewpoints. As variants, we also compare models trained with randomly selected subsets and those selected using farthest point sampling (FPS) [22].

As an implementation detail, to obtain the machine-generated 2D panoptic segmentation as input to the Panoptic Lifting model, we used a Mask2Former model pre-trained on COCO. For the backbone of Mask2Former, we employed a Swin-L [17] model pre-trained on ImageNet-21K [8]. Additionally, we mapped the COCO classes used by this model and the Replica classes to the 31 ScanNet [7] classes.

## 4.4. Results

### 4.4.1. Comparison on the initial training viewpoints

Table 1 presents quantitative comparison of the inference accuracy of each model on the test set of the initial training viewpoints, while Fig. 4 shows the comparison of the rendered RGB images, semantic segmentation masks, and instance segmentation masks output by each model. Both the *FULL* and *PRE* models were trained on all of the initial training viewpoints, achieving sufficient inference accuracy

Table 1. Quantitative comparison of the inference accuracy of each model on the test set of the initial training viewpoints. For the *INC* model, “Random”, “FPS”, and “Voxel” refer to the selection of initial viewpoints for input by random sampling, farthest point sampling, and our voxel-based method that maximizes the number of observed voxels, respectively.

PSNR							
<i>FULL</i>	32.8	-	-	-	-	-	-
<i>PRE</i>	32.6	-	-	-	-	-	-
<i>INC</i>	Number of selected viewpoints $m$						
	0	5	10	15	20	25	30
Random	14.6	21.2	24.2	25.1	26.2	27.0	28.4
FPS	-	15.1	19.2	22.3	25.7	27.3	27.8
<b>Voxel</b>	-	<b>26.7</b>	<b>27.4</b>	<b>28.2</b>	<b>28.5</b>	<b>29.2</b>	<b>29.6</b>
mIoU							
<i>FULL</i>	0.57	-	-	-	-	-	-
<i>PRE</i>	0.55	-	-	-	-	-	-
<i>INC</i>	Number of selected viewpoints $m$						
	0	5	10	15	20	25	30
Random	0.32	0.47	0.46	0.49	0.42	0.50	0.49
FPS	-	0.38	0.46	0.49	0.51	<b>0.56</b>	0.49
<b>Voxel</b>	-	<b>0.52</b>	<b>0.51</b>	<b>0.54</b>	<b>0.55</b>	0.55	<b>0.57</b>
$PQ^{\text{scene}}$							
<i>FULL</i>	0.32	-	-	-	-	-	-
<i>PRE</i>	0.29	-	-	-	-	-	-
<i>INC</i>	Number of selected viewpoints $m$						
	0	5	10	15	20	25	30
Random	0.18	0.21	0.26	0.20	0.28	0.28	0.22
FPS	-	0.19	<b>0.29</b>	0.23	0.27	<b>0.31</b>	0.26
<b>Voxel</b>	-	<b>0.30</b>	0.27	<b>0.31</b>	<b>0.32</b>	<b>0.31</b>	<b>0.27</b>

across all evaluation metrics. In the *INC* model, when no selection was made from the initial training viewpoints, training with only the additional viewpoints caused catastrophic forgetting, leading to a decline in inference accuracy compared to the *PRE* model. This is also confirmed by the rendered images, where noise resulting from catastrophic forgetting can be seen. On the other hand, the models with initial training viewpoints selected by random sampling, farthest point sampling, or our proposed voxel-based method all prevented catastrophic forgetting through explicit training on these viewpoints, leading to improved inference accuracy from the *PRE* model. Furthermore, when comparing cases with the same number of selected viewpoints, our proposed method consistently achieved higher inference accuracy, demonstrating its effectiveness in preventing catastrophic forgetting.

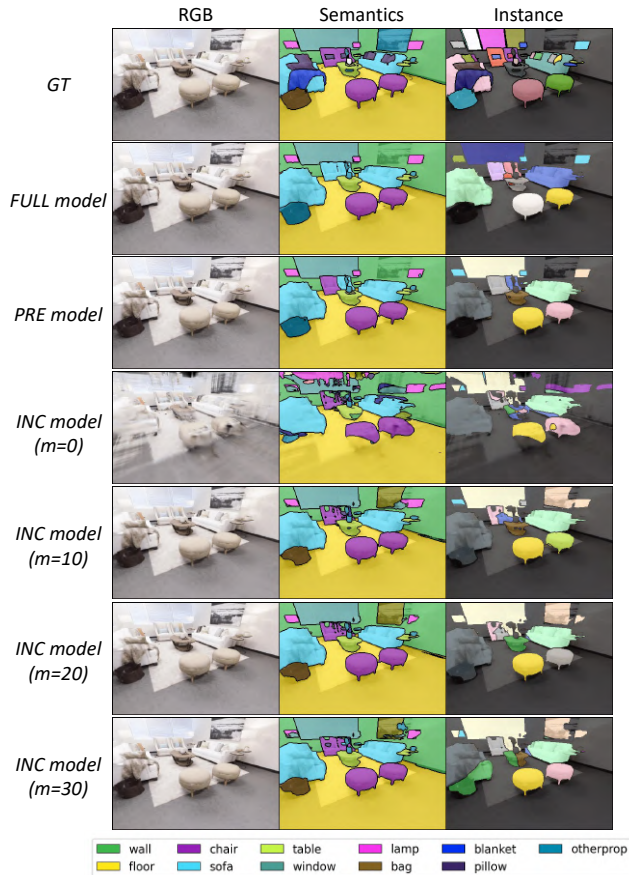


Figure 4. Comparison of the rendered RGB images, semantic segmentation masks, and instance segmentation masks output by each model of the initial training viewpoints. For the *INC* model, we present the outputs where the initial training viewpoints were selected using our proposed method, with  $m$  representing the number of selected viewpoints.

#### 4.4.2. Comparison on the additional viewpoints

Table 2 presents quantitative comparison of the inference accuracy of each model on the test set of the additional viewpoints, while Fig. 5 shows the rendered RGB images, semantic segmentation masks, and instance segmentation masks output by each model. Here, to eliminate the noise caused by the inherent inaccuracy of the Mask2Former outputs used as pseudo labels, specifically the semantic mask and instance mask, we present the results for the *INC* model using ground truth (GT) masks as input. In the *INC* model, incremental learning with the additional viewpoints improved inference accuracy compared to the *PRE* model, confirming the effectiveness of our incremental learning approach. The rendered images also show that objects unrecognized in the *PRE* model are accurately recognized. Furthermore, since the observation data from the additional viewpoints used as input to the *INC* model is unchanged re-

Table 2. Quantitative comparison of the inference accuracy of each model on the test set of the additional viewpoints. Since the *INC* model is trained on all additional viewpoints, its inference accuracy at these viewpoints does not depend on the selection method of the initial training viewpoints. Therefore, only the results obtained using our proposed method are presented in the table.

PSNR							
<i>FULL</i>	25.2	-	-	-	-	-	-
<i>PRE</i>	18.3	-	-	-	-	-	-
<i>INC</i>	Number of selected viewpoints $m$						
	0	5	10	15	20	25	30
	<b>25.1</b>	<b>25.3</b>	<b>25.3</b>	<b>25.4</b>	<b>25.2</b>	<b>25.4</b>	<b>25.2</b>
mIoU							
<i>FULL</i>	0.33	-	-	-	-	-	-
<i>PRE</i>	0.41	-	-	-	-	-	-
<i>INC</i>	Number of selected viewpoints $m$						
	0	5	10	15	20	25	30
	<b>0.81</b>	<b>0.87</b>	<b>0.86</b>	<b>0.87</b>	<b>0.87</b>	<b>0.87</b>	<b>0.87</b>
PQ <sup>scene</sup>							
<i>FULL</i>	0.30	-	-	-	-	-	-
<i>PRE</i>	0.29	-	-	-	-	-	-
<i>INC</i>	Number of selected viewpoints $m$						
	0	5	10	15	20	25	30
	<b>0.36</b>	<b>0.37</b>	<b>0.38</b>	<b>0.42</b>	<b>0.40</b>	<b>0.36</b>	<b>0.39</b>

gardless of how the initial training viewpoints are selected, no significant difference in inference accuracy was observed between selection methods.

#### 4.4.3. Computational Time

Table 3 presents a comparison of the average computation time per epoch during training for each model. We conducted our experiments on a machine with an AMD EPYC 7352 CPU (24 cores, 2.3 GHz), an NVIDIA RTX A6000 GPU (48GB VRAM), and 384GB of RAM. By performing incremental learning, we reduced the amount of training data required to build a model capable of recognizing the entire scene, including both the initial viewpoints and the additional viewpoints, which led to a reduction in computational time during training the model. Notably, the computation time for the *INC* model that achieved accuracy close to the naïve approach was reduced to 11–14%.

## 5. Limitations

Our method confirmed the effectiveness of introducing incremental learning into Panoptic Lifting, but several limitations remain. As stated in Section 4.4.2, the pseudo labels

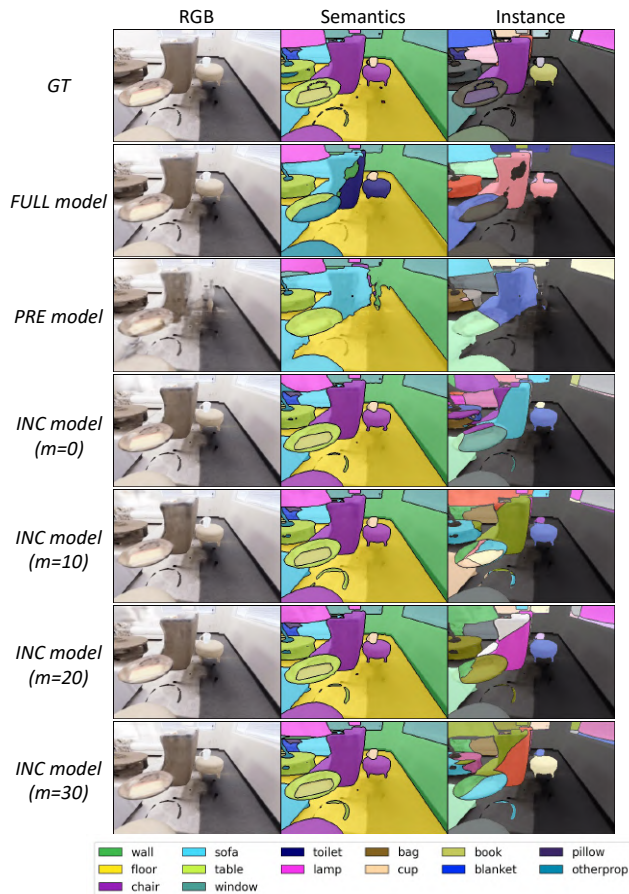


Figure 5. Comparison of the rendered RGB images, semantic segmentation masks, and instance segmentation masks output by each model of the additional viewpoints.

can sometimes be inaccurate (Figure 6). As a result, even with incremental learning from the *PRE* model, the *INC* model may produce inaccurate inferences, reducing accuracy compared to the *PRE* model. Similarly, at initial viewpoints with inaccurate pseudo labels, our method, which does not consider pseudo label accuracy, may select inaccurate viewpoints.

## 6. Conclusion

We have proposed the forgetting-free incremental learning framework for Panoptic Lifting that prevents catastrophic forgetting while reducing the computational cost required for training a model capable of recognizing the entire scene. Our approach utilizes a viewpoint selection algorithm that prioritizes selecting the subset of the initial training viewpoints to maximize the coverage of the visible regions of the scene by considering the number of observed voxels. In the experiments, the effectiveness of our approach was confirmed, and the computational time required to build a

Table 3. Comparison of the average computation time [s] per epoch during training for each model. To consider the difference in computation time caused by incorporating multiple types of loss functions used during training (in Section 3.3 of [27]) into the overall loss function, the table presents the average computation time per epoch when these loss functions are included. Furthermore, since the computation time of the *INC* model does not depend on the selection method of the initial training viewpoints, only the values obtained using our proposed method are shown.

	Appearance & semantic loss	Instance loss	Segment loss	
<i>FULL</i>	12,857	14,033	13,577	
<i>PRE</i>	6,484	9,244	9,576	
<i>INC</i>				
Number of selected viewpoints	0	1,172	1,358	1,603
	5	1,295	1,504	1,753
	10	1,339	1,525	1,778
	15	1,411	1,594	1,838
	20	1,380	1,569	1,834
	25	1,387	1,601	1,846
	30	1,396	1,597	1,860

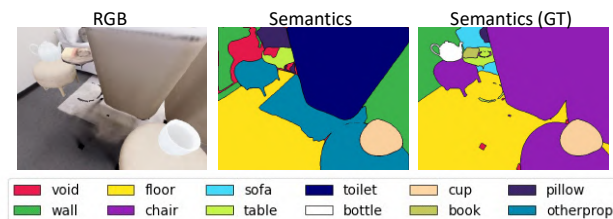


Figure 6. An example of inaccurate pseudo labels for input at the additional viewpoints. In cases where the observation camera is close to a low-textured surface of an object, such as the flat back of a chair, pseudo labels tend to be inaccurate. If the front of the chair is correctly recognized in the *PRE* model, the inference accuracy may decrease in the *INC* model.

model with sufficient accuracy was reduced compared to the naïve approach. As future work, we consider incorporating a viewpoint selection algorithm that accounts for the accuracy of pseudo labels. In addition, exploring its applicability beyond small indoor scenes, particularly to real or large-scale environments, and further extending it to dynamic scenes would be a valuable direction.

## Acknowledgements

This work was partly supported by Japan Society for the Promotion of Science (JSPS) KAKENHI JP24H00733 and Japan Science and Technology Agency (JST) ASPIRE JP-MJAP2305.

## References

- [1] Chen Anpei, Xu Zexiang, Geiger Andreas, Yu Jingyi, and Su Hao. TensoRF: Tensorial radiance fields. In *ECCV*, pages 333–350, 2022. 2
- [2] Chaudhry Arslan, Rohrbach Marcus, Elhoseiny Mohamed, Ajanthan Thalaiyasingam, K. Dokania Puneet, H. S. Torr Phillip, and Ranzato Marc’Aurelio. On tiny episodic memories in continual learning. *arXiv preprint arxiv:1902.10486*, pages 1–15, 2019. 3
- [3] Jonathan T. Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P. Srinivasan. Mip-NeRF: A multiscale representation for anti-aliasing neural radiance fields. In *ICCV*, pages 5855–5864, 2021. 2
- [4] Wang Bing, Lu Chen, and Bo Yang. DM-NeRF: 3D scene geometry decomposition and manipulation from 2D images. In *ICLR*, pages 1–25, 2023. 2
- [5] Cheng Bowen, Misra Ishan, G. Schwing Alexander, Kirillov Alexander, and Girdhar Rohit. Masked-attention mask transformer for universal image segmentation. In *ICCV*, pages 1290–1299, 2022. 2
- [6] Zhipeng Cai and Matthias Müller. CLNeRF: Continual learning meets NeRF. In *ICCV*, pages 23185–23194, 2023. 3
- [7] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In *CVPR*, pages 5828–5839, 2017. 6
- [8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *ICCV*, pages 248–255, 2009. 6
- [9] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B. McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *ICRA*, pages 2553–2560, 2022. 5
- [10] Hinton Geoffrey, Vinyals Oriol, and Dean Jeff. Distilling the knowledge in a neural network. *arXiv preprint arxiv:1503.02531*, pages 1–9, 2015. 3
- [11] Shin Hanul, Kwon Lee Jung, Kim Jaehong, and Kim Jiwon. Continual learning with deep generative replay. In *NIPS*, pages 2994–3003, 2017. 3
- [12] Chung Jaeyoung, Lee Kanggeon, Baik Sungyong, and Mu Lee Kyoung. MEIL-NeRF: Memory-efficient incremental learning of neural radiance fields. *arXiv preprint arxiv:2212.08328*, pages 1–18, 2022. 3
- [13] Straub Julian, Whelan Thomas, Ma Lingni, Chen Yufan, Wijmans Erik, Green Simon, J. Engel Jakob, Mur-Artal Raul, Ren Carl, Verma Shobhit, Clarkson Anton, Yan Mingfei, Budge Brian, Yan Yajie, Pan Xiaqing, Yon June, Zou Yuyang, Leon Kimberly, Carter Nigel, Briaies Jesus, Gillingham Tyler, Mueggler Elias, Pesqueira Luis, Savva Manolis, Batra Dhruv, M. Strasdat Hauke, De Nardi Renzo, Goesele Michael, Lovegrove Steven, and Newcombe Richard. The Replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, pages 1–10, 2019. 5
- [14] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollar. Panoptic segmentation. In *CVPR*, pages 9404–9413, 2019. 1, 2, 6
- [15] 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 *ICCV*, pages 12871–12881, 2022. 2
- [16] Zhang Letian, Li Ming, Chen Chen, and Xu Jie. IL-NeRF: Incremental learning for neural radiance fields with camera pose alignment. *arXiv preprint arxiv:2312.05748*, pages 1–23, 2023. 3
- [17] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, pages 10012–10022, 2021. 6
- [18] Guo Mengqi, Li Chen, Chen Hanlin, and Hee Lee Gim. Incremental neural implicit representation with uncertainty-filtered knowledge distillation. In *ECCV*, pages 237–254, 2024. 3
- [19] 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. *Communication of the ACM*, 65(1):99–106, 2021. 1, 2
- [20] Koenig Nathan and Howard Andrew. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *IROS*, pages 2149–2154, 2004. 5
- [21] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions–i. *Mathematical Programming*, 14(1):265–294, 1978. 5
- [22] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NIPS*, pages 5099–5108, 2017. 6
- [23] Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. KiloNeRF: Speeding up neural radiance fields wreplith thousands of tiny mlps. In *ICCV*, pages 14335–14345, 2021. 2
- [24] Po Ryan, Dong Zhengyang, W. Bergman Alexander, and Wetzstein Gordon. Instant continual learning of neural radiance fields. In *ECCV Workshop*, 2023. 3
- [25] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A platform for embodied AI research. In *ICCV*, pages 9339–9348, 2019. 5
- [26] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, pages 4104–4113, 2016. 4
- [27] 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*, pages 9043–9052, 2023. 1, 2, 6, 8

- [28] Lin Tsung-Yi, Maire Michael, Belongie Serge, Bourdev Lubomir, Girshick Ross, Hays James, Perona Pietro, Ramanan Deva, Zitnick C. Lawrence, and Dollár Piotr. Microsoft COCO: Common objects in context. In *ECCV*, pages 740–755, 2014. [2](#)