Geometry and Uncertainty-Aware 3D Point Cloud Class-Incremental Semantic Segmentation

Yuwei Yang[1] Munawar Hayat[2] Zhao Jin[1] Chao Ren[1] Yinjie Lei[1,✉]
[1]Sichuan University [2]Monash University
yuwei@stu.scu.edu.cn munawar.hayat@monash.edu jinzhao@stu.scu.edu.cn
chaoren@scu.edu.cn yinjie@scu.edu.cn

Abstract

Despite the significant recent progress made on 3D point cloud semantic segmentation, the current methods require training data for all classes at once, and are not suitable for real-life scenarios where new categories are being continuously discovered. Substantial memory storage and expensive re-training is required to update the model to sequentially arriving data for new concepts. In this paper, to continually learn new categories using previous knowledge, we introduce class-incremental semantic segmentation of 3D point cloud. Unlike 2D images, 3D point clouds are disordered and unstructured, making it difficult to store and transfer knowledge especially when the previous data is not available. We further face the challenge of semantic shift, where previous/future classes are indiscriminately collapsed and treated as the background in the current step, causing a dramatic performance drop on past classes. We exploit the structure of point cloud and propose two strategies to address these challenges. First, we design a geometry-aware distillation module that transfers point-wise feature associations in terms of their geometric characteristics. To counter forgetting caused by the semantic shift, we further develop an uncertainty-aware pseudo-labelling scheme that eliminates noise in uncertain pseudo-labels by label propagation within a local neighborhood. Our extensive experiments on S3DIS and ScanNet in a class-incremental setting show impressive results comparable to the joint training strategy (upper bound). Code is available at: https://github.com/leolyj/3DPC-CISS

1. Introduction

The semantic segmentation of 3D point cloud plays a crucial role in applications such as virtual reality, robotics and autonomous vehicles. In recent years, a number of point cloud segmentation methods [16, 28, 29, 38, 39, 45] have achieved remarkable performance in the traditional setting where all classes are learned at once. Nevertheless, new categories are gradually discovered in real-life scenarios, and updating the model to cater for these new categories requires large memory storage and expensive re-training. In such situations, as illustrated in Fig. 1, class-incremental learning provides a promising paradigm, since it enables progressively learning novel knowledge in an efficient manner while preserving the previous capabilities.

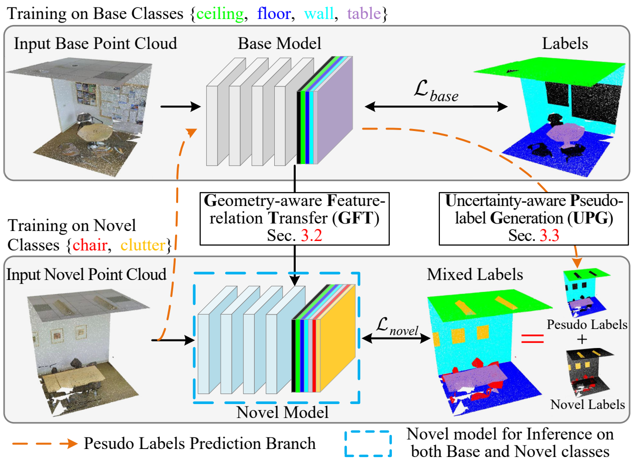

Training on Base Classes {ceiling, floor, wall, table}
Input Base Point Cloud
Base Model
Labels
$\mathcal{L}_{base}$
Geometry-aware Feature-relation Transfer (GFT) Sec. 3.2
Uncertainty-aware Pseudo-label Generation (UPG) Sec. 3.3
Training on Novel Classes {chair, clutter}
Input Novel Point Cloud
Mixed Labels
$\mathcal{L}_{novel}$
Pesudo Labels
+
=
Novel Labels
Novel Model
Pesudo Labels Prediction Branch
Novel model for Inference on both Base and Novel classes

Figure 1. To continually learn new categories without forgetting the previous ones, we leverage Geometry-aware Feature-relation Transfer (Sec. 3.2) to distill point-wise relationships from the base model and further employ Uncertainty-aware Pseudo-label Generation (Sec. 3.3) to synthesize pseudo labels of old classes with low uncertainty as guidance for novel model training.

The existing research on class-incremental learning is mostly on 2D image classification [17, 19, 21, 32] with some efforts extended to RGB semantic segmentation [3, 4, 10, 41]. These methods employ a strategy based upon *regularization* [11, 19, 44], *rehearsal/replay* [2, 17, 24, 32, 35] or *knowledge distillation* [8, 20, 21] to preserve previous knowledge. At present, only a few works have investigated

✉ Corresponding Author: Yinjie Lei (yinjie@scu.edu.cn)

3D point clouds based incremental learning for classification [5, 6, 9, 22, 43]. They focus on the classification of an individual object and extend 2D methods to 3D. Unlike classification which only considers a single object (not the scene with multiple objects), continually learning to segment 3D point cloud in complex scenes introduces multiple new challenges and has not been previously studied.

3D point clouds are disordered and unstructured, which makes it difficult to preserve previous knowledge and results in *catastrophic forgetting* [13,33,36]. This specifically becomes pronounced when old data is not available. We further observe that the 3D class-incremental segmentation faces the phenomenon of semantic shift, where the points belonging to old classes are indiscriminately collapsed into background during the current learning step. The semantic shift further suppresses the capability of the model to recognize old categories, thus exacerbating forgetting.

In this paper, we are the first to propose a class-incremental learning approach for 3D point cloud semantic segmentation. To prevent forgetting caused by unstructured point clouds, we design a Geometry-aware Feature-relation Transfer (GFT) strategy to transfer the structural relationships among point features. Moreover, to address the semantic shift issue, we assign uncertainty-aware pseudo-labels to the background points. Different from the conventional approaches, where pseudo-labels are directly obtained from the old model, we estimate uncertainties according to the distribution characteristics of points, and leverage the neighborhood information to propagate labels from low to high uncertainties. Our Uncertainty-aware Pseudo-label Generation (UPG), therefore, assists in eliminating the influence of noisy labels and helps tackle the semantic shift issue. Note that our approach does not involve any rehearsal or memory replay buffer to store old data or its annotations during the incremental process. We show promises of our approach through comprehensive evaluations on benchmarks defined on public datasets *i.e.,* S3DIS [1] and ScanNet [7]. Our key contributions are:

- A class-incremental learning framework for 3D point cloud semantic segmentation, to sequentially adapt to new classes from previous acquired knowledge.
- To transfer previous knowledge and prevent forgetting caused by unstructured nature of the point clouds, we propose a Geometry-aware Feature-relation Transfer (GFT) module that captures the point-wise feature relations based on the geometric information.
- To tackle the semantic shift issue where old classes are indiscriminately collapsed into the background, we design an Uncertainty-aware Pseudo-label Generation (UPG) strategy to enhance pseudo-labelling quality and thus provide effective guidance for old classes.
- Compared with several baselines on multiple benchmarks, our approach achieves promising results for 3D class-incremental semantic segmentation, closer to the joint training (upper bound) using all data at once.

2. Related Works

Incremental learning [30] enables the model to learn new knowledge *e.g*. new classes (class-incremental), new tasks (task-incremental) or new domains (domain-incremental), while retaining the previously learned information. Since task-incremental and domain-incremental can be regarded as a simplified version of the more challenging class incremental setup, we mainly focus on class-incremental methods. Below, we touch upon RGB image based class-incremental learning methods for 2D classification, and semantic segmentation, followed by discussion on incremental learning approaches for 3D point cloud data.

Class-Incremental learning. We can broadly categorize the existing approaches on image classification into the following families: (a) *regularization-based methods* [11, 19, 44] that avoid catastrophic forgetting by imposing constraints on the previously learned parameters while updating on new classes. (b) *distillation-based methods* [8, 20, 21] that transfer the knowledge acquired from previous models to new tasks by knowledge distillation from the previous temporal checkpoint. (c) *rehearsal/replay-based methods* [2, 17, 24, 32, 35] that define a memory budget to store data instances/features from the previous classes as exemplars, or employ a generative model to synthesize the information related to the previous tasks, and then use these exemplars to rehearse the previously acquired knowledge. (d) *Parameter-isolation-based methods* [12, 23, 31, 34, 42] that define different sub-networks or paths to prevent the interference of previously learned knowledge on new tasks by separating the parameters between different tasks.

Class-Incremental Segmentation on 2D Images. Recent efforts [3, 4, 10, 25, 26, 41] extend the traditional 2D image segmentation to class-incremental setting. Cermelli *et al.* [4] propose a knowledge distillation loss and a modified cross-entropy loss of background pixels to reduce forgetting on previous knowledge. [10] adopts a multi-scale pooling for distillation and entropy-based pseudo-labeling to transfer knowledge of the old classes. Recently, [41] proposes an uncertainty-aware contrastive distillation approach with impressive results. Class-incremental semantic segmentation is also studied in weakly supervised setting to alleviate the expensive and time-consuming pixel-level labeling in [3].

Class-Incremental Learning on 3D Point Cloud. Different from 2D images, class-incremental learning on 3D point cloud has only recently been tackled, with main focus on classification [5, 6, 9, 22, 43]. [9] proposes a geometric attention mechanism to prevent forgetting caused by redundant geometric information. For distillation, shared point-knowledge memory attention is explored in [22] while

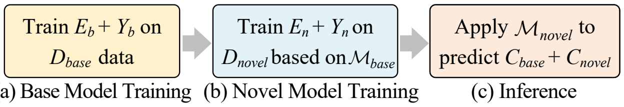
[6] develops a 3D distillation method. Memory replay using exemplars is explored in [43]. [5] study few-shot class-incremental learning for 3D classification Further, a teacher-student architecture [46] is developed to overcome co-occurrence challenge in 3D-incremental object detection. Despite the considerable efforts, fine-grained 3D class-incremental segmentation remains an unexplored area beyond the traditional offline 3D segmentation [16, 28, 29, 38, 39, 45]. In this paper, we propose multiple novel design choices to tackle the challenges associated with point clouds including the forgetting on unstructured 3D data and unique point semantic shift in class-incremental segmentation.

3. Methodology

3.1. Problem Formulation

Lets define a dataset D containing a set of $(\mathbf{P}, \mathbf{L})$ pairs, where $\mathbf{P} \in R^{3+F}$ represents the input 3D point cloud with xyz coordinates and F-dimensional features (*e.g. rgb color* etc.), and $\mathbf{L}$ contains corresponding point-wise labels. In the 3D class-incremental semantic segmentation task, C_{base} and C_{novel} are two disjoint class sets (*i.e.* $C_{base} \cap C_{novel} = \varnothing$), with corresponding D_{base} and D_{novel} data available respectively. Our goal is to train an incremental model using only the D_{novel} data, based on a pre-trained model on D_{base}, which recognizes both the base and novel classes at inference time. The structure of segmentation model is defined as the combination of feature extractor $E(\cdot)$ and classifier $Y(\cdot)$. As shown in Fig. 2, the basic steps for class-incremental semantic segmentation on 3D point cloud can be summarized as: **(a)** Train the base model (feature extractor E_b combined with the classifier Y_b, denoted as base/old model $\mathcal{M}_{base}$) on D_{base}. **(b)** Use the pre-trained base model to initialize a new model and randomly initialize the last layer of new classifier Y_n (denoted as novel model $\mathcal{M}_{novel}$), and train on D_{novel} data. **(c)** Apply novel model $\mathcal{M}_{novel}$ to segment point clouds of all $C_{base}+C_{novel}$ classes in the evaluation phase.

Fig. 3 illustrates the overall schematics of our framework, which completes the 3D class-incremental segmentation without requiring any previous data or annotations. In base $\mathcal{M}_{base}$ training, we adopt the DGCNN [38] as our backbone network, which constructs a dynamic graph to obtain and fuse both global semantic and local EdgeConv features. In the novel model $\mathcal{M}_{novel}$ training, we apply the Geometry-aware Feature-relation Transfer (GFT) module to perform base-to-novel knowledge distillation, so as to maximize the retention of previous information. Moreover, we employ our proposed Uncertainty-aware Pseudo-label Generation (UPG) strategy to synthesize pseudo labels of old categories, which will be combined with the current new class labels to assist $\mathcal{M}_{novel}$ training. In the following sections, we will detail the components of the network.


Train $E_b + Y_b$ on D_{base} data
Train $E_n + Y_n$ on D_{novel} based on $\mathcal{M}_{base}$
Apply $\mathcal{M}_{novel}$ to predict $C_{base} + C_{novel}$

(a) Base Model Training (b) Novel Model Training (c) Inference

Figure 2. The basic steps of class-incremental segmentation on 3D point cloud. (a) E_b and Y_b denotes the feature extractor and classifier of the base model $\mathcal{M}_{base}$ respectively. (b) E_n and Y_n represents the novel feature extractor initialized by E_b and classifier respectively. (c) After training $\mathcal{M}_{novel}$ on D_{novel} data, the model can predict both base and novel classes in the inference.

3.2. Geometry-aware Feature-relation Transfer

For our approach, we do not have access to previous data, or exemplar point clouds from previously encountered categories for replay. In order to tackle catastrophic forgetting that can cause significant performance degradation on old classes, we model the point-wise relative relationships within the geometric neighbors for transfer to retain knowledge learned on previous classes. We argue that the geometry-aware feature relation is discriminative for various semantic categories, and can be exploited to migrate information while learning continually.

Let $\mathbf{P}_n \in D_{novel}$ be the current input point cloud of novel classes with S points. We define features extracted by E (including E_b for base, E_n for novel model) of $\mathbf{P}_n$ as $\mathbf{F}_n = E(\mathbf{P}_n)$. E is implemented using DGCNN [38], which extracts feature $\mathbf{F}_n$ by aggregating both global semantic and local EdgeConv features. To model the geometry of the point cloud, we first apply the Farthest Point Sampling (FPS) [29] on the input point cloud to uniformly sample r-proportional points (the number is $Z = \lfloor r * S \rfloor$) as anchors $\{\mathbf{P}_n^a\}_{a=1}^Z$). We calculate the ℓ_2 distance between each point and anchors using *xyz* coordinates, and apply KNN to sample the nearest K points $\mathbf{P}_n^{a,k}, k \in \mathcal{N}(a)$ to form areas reflecting the local geometric structures. On this basis, we represent the point-wise relative relationships $\mathcal{R}^a$ within the geometric neighbors, which carry the crucial structured knowledge for migration and can be formulated as:

$$\mathcal{R}^a = \frac{1}{K} \sum_{k \in \mathcal{N}(a)} (p_n^{a,k} - p_n^a) \oplus (\mathbf{F}_n^{a,k} - \mathbf{F}_n^a), \qquad (1)$$

where p_n are the *xyz* coordinates of points in $\mathbf{P}_n$, $\oplus$ indicates the concatenation operation. The feature relation on $\mathbf{P}_n$ of the base and novel model can be expressed as $\mathcal{R}_{base}^a$ and $\mathcal{R}_{novel}^a$ respectively. We perform base-to-novel feature relation distillation via simple MSE loss:

$$\mathcal{L}_{trans} = \frac{1}{Z} \sum_{a=1}^{Z} ||\mathcal{R}_{novel}^a - \mathcal{R}_{base}^a||^2. \qquad (2)$$

Using this distillation scheme, the structural and semantic knowledge contained in the old model is transferred to the new one, and helps prevent catastrophic forgetting.

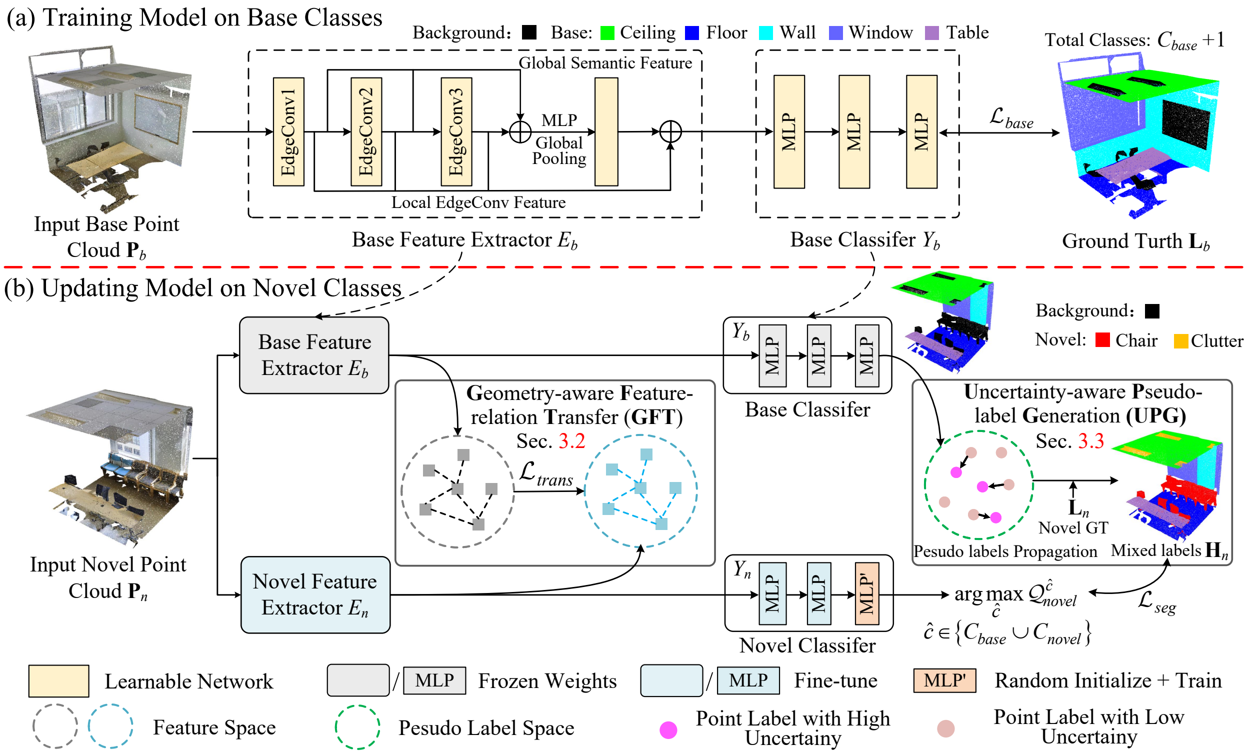

(a) Training Model on Base Classes
Background: ■ Base: 🟩 Ceiling 🟦 Floor 🟦 Wall 🟦 Window 🟪 Table
Total Classes: $C_{base}+1$
Global Semantic Feature
EdgeConv1
EdgeConv2
EdgeConv3
MLP
Global Pooling
Local EdgeConv Feature
MLP
MLP
MLP
$\mathcal{L}_{base}$
Input Base Point Cloud $\mathbf{P}_b$
Base Feature Extractor E_b
Base Classifer Y_b
Ground Turth $\mathbf{L}_b$
(b) Updating Model on Novel Classes
Base Feature Extractor E_b
Y_b
MLP
MLP
MLP
Base Classifer
Background: ■
Novel: 🟥 Chair 🟨 Clutter
Geometry-aware Feature-relation Transfer (GFT)
Sec. 3.2
$\mathcal{L}_{trans}$
Uncertainty-aware Pseudo-label Generation (UPG)
Sec. 3.3
$\mathbf{L}_n$
Novel GT
Pesudo labels Propagation
Mixed labels $\mathbf{H}_n$
Input Novel Point Cloud $\mathbf{P}_n$
Novel Feature Extractor E_n
Y_n
MLP
MLP
MLP'
Novel Classifer
$\arg\max_{\hat{c}} \mathcal{Q}_{novel}^{\hat{c}}$
$\hat{c} \in \{C_{base} \cup C_{novel}\}$
$\mathcal{L}_{seg}$
Learnable Network
MLP Frozen Weights
MLP Fine-tune
MLP' Random Initialize + Train
Feature Space
Pesudo Label Space
Point Label with High Uncertainy
Point Label with Low Uncertainy

Figure 3. The overall architecture of our proposed framework. (a) During base stage training, we train a backbone encoder E_b and classifier Y_b on the base classes. (b) While updating the base model to novel classes, we develop a geometry-aware feature-relation transfer module to distill previous knowledge of point-wise relationships and an uncertainty-aware pseudo-labelling strategy to tackle semantic shift issue.

3.3. Uncertainty-aware Pseudo-label Generation

Since labels for points of base categories C_{base} are not available during incremental learning process, they will be treated as an undifferentiated "background" class while updating the model on the D_{novel}. However, these points belonging to old classes have significantly varying semantic representations. If left unresolved, the novel model becomes biased towards the newly added classes and ignore old ones, thus giving rise to the semantic shift issue that exacerbates forgetting. A naive way to deal with semantic shift is to use the pseudo labels assigned by the old model. However, such pseudo labels are usually inaccurate and have high uncertainty, which leads to training instability. Therefore, we propose uncertainty-aware pseudo labeling (see Fig. 3) refines highly uncertain noisy labels by neighbors label propagation.

When input a novel point cloud $\mathbf{P}_n \in D_{novel}$, the predicted class probabilities of the base and novel model can be respectively expressed as $\mathcal{Q}_{base}^{V} = \mathcal{M}_{base}(\mathbf{P}_n)$ and $\mathcal{Q}_{novel}^{G} = \mathcal{M}_{novel}(\mathbf{P}_n)$. V and G denotes the number of $C_{base}+1$ and $C_{novel}+1$, *i.e.,* the total number of semantic classes plus a "background" class. Different from the traditional Monte Carlo Dropout (MC-Dropout) method [14], which performs multiple predictions to estimate uncertainty, we apply neighborhood spatial aggregation method combined with MC-dropout to complete the estimation of the point distribution uncertainty at once, which is based on space-dependent sampling and has been demonstrated to be effective in [27]. We first define the set $\{p_n^1, p_n^2, ..., p_n^i, ..., p_n^S\}$ as xyz positions of point cloud $\mathbf{P}_n$, and do K-Nearest Neighbors (KNN) search to get T neighbors of each point $p_n^i, i = 1, 2, ..., S$ as $p_n^{i,t}, t \in \mathcal{N}(i)$ based on xyz coordinates. We adopt Bayesian Active Learning by Disagreement (BALD) [15] as our spatial sampling uncertainty estimation function on point cloud, as follows:

$$\begin{aligned}\mathcal{U}_n^i = &\\ -\sum_c[\frac{1}{T}\sum_t q(y_n^i = c|\mathbf{P}_n^i, \hat{\omega}_t)]\log[\frac{1}{T}\sum_t q(y_n^i = c|\mathbf{P}_n^i,\ \hat{\omega}_t)]&\\ + \frac{1}{T}\sum_{c,t} q(y_n^i = c|\mathbf{P}_n^i, \hat{\omega}_t)\log q(y_n^i = c|\mathbf{P}_n^i, \hat{\omega}_t),&\end{aligned} \quad (3)$$

where y_n^i represents the predicted label for $\mathbf{P}_n^i$ of class c. $\hat{\omega}_t$ denotes the Bernoulli distribution over the weights of the neighboring points distribution, q represents the conditional probabilities. We calculate the normalized cosine similarity between coordinates $p_n^{i,t}, t \in \mathcal{N}(i)$ and p_n^i to implement $\hat{\omega}_t$, which can be denoted as σ_n^t. Finally, the uncertainty of each point of base model output on $\mathbf{P}_n^i$ is given as:

$$\begin{aligned}\mathcal{U}_n^i = -\sum_c [\frac{1}{T}\sum_t \mathcal{Q}_{base}^{i,c} * \sigma_n^t]\mathrm{log}[\frac{1}{T}\sum_t \mathcal{Q}_{base}^{i,c} * \sigma_n^t] \\ + \frac{1}{T}\sum_{c,t}(\mathcal{Q}_{base}^{i,c} * \sigma_n^t)\mathrm{log}(\mathcal{Q}_{base}^{i,c} * \sigma_n^t).\end{aligned} \tag{4}$$

After obtaining the uncertainty by Eq. (4), we set τ as the threshold to determine the points with the high or low uncertainty. We argue that the predicted labels of points are usually inaccurate leading to noise in generated pseudo labels. To provide better guidance to novel model, we aim to eliminate the uncertainty to refine the pseudo-labels quality. We observe that for a point on point cloud, the points in its local neighbors are more likely to have the same semantic representations. Therefore, for a point with high uncertainty, our strategy is to replace its prediction with the label of its nearest neighbor having the low uncertainty. Thus, the pseudo labels generated by the base model for input point cloud $\mathbf{P}_n^i$ will be combined with the current novel class labels $\mathbf{L}_n^i$ to form the mixed labels $\mathbf{H}_n^i$ for novel model training:

$$\mathbf{H}_n^i = \begin{cases} \underset{c}{\mathrm{argmax}}\ \mathcal{Q}_{base}^{i,c} & \mathbf{L}_n^i = c'_{bg}, \underset{c}{\mathrm{argmax}}\ \mathcal{Q}_{base}^{c} \neq c_{bg} \text{ and } \mathcal{U}_n^i \leq \tau, \\ \underset{c}{\mathrm{argmax}}\ \mathcal{Q}_{base}^{t,c} & \mathbf{L}_n^i = c'_{bg}, \underset{c}{\mathrm{argmax}}\ \mathcal{Q}_{base}^{c} = c_{bg} \text{ or } \mathcal{U}_n^i > \tau, \\ \mathbf{L}_n^i & \mathbf{L}_n^i \neq c'_{bg}, \\ \text{ignore} & \text{otherwise}, \end{cases} \tag{5}$$

where the c_{bg} and c'_{bg} denote a "background" class other than the C_{base} and C_{novel} semantic classes respectively. $c \in \{C_{base} \cup c_{bg}\}$, $\mathcal{Q}_{base}^{t,c}$ represent a nearest point t among T neighbor points for which $\mathcal{Q}_{base}^{t,c} \neq c_{bg}$ and $\mathcal{U}_n^t \leq \tau$. It is worth noting that we ignore part of the labels, for which the predicted probabilities of both the current point $\mathbf{P}_n^i$ and its neighbors $\mathbf{P}_n^t,\ t \in \mathcal{N}(i)$ have high uncertainty. Although these point labels are relatively few, they still have an averse impact on the results due to significant noise content. Finally, the cross-entropy segmentation loss is constructed between the novel model output and mixed label $\mathbf{H}_n^i$:

$$\mathcal{L}_{seg} = -\sum_i\sum_{\hat{c}}\mathbf{H}_n^i log(\mathcal{Q}_{novel}^{i,\hat{c}}), \tag{6}$$

where $i \in S$, $\hat{c} \in \{C_{base} \cup C_{novel}\}$. Base on above approach, the output of the base model will guide the novel model training to overcome the effects of semantic shift.

3.4. Network Training and Inference

We first train the base model E_b and Y_b using the cross-entropy on input base point cloud $\mathbf{P}_b$ and its labels $\mathbf{L}_b$:

$$\mathcal{L}_{base} = -\sum_i\sum_c\mathbf{L}_b^i log(Y_b^c(E_b(\mathbf{P}_b^i))), \tag{7}$$

where $i \in S$, $c \in \{C_{base} \cup c_{bg}\}$ denotes the semantic base classes plus a "background" class. To train the novel model, we optimize the last layer of Y_n and fine-tune the rest part by minimizing the feature relation distillation loss based on geometric structure and the segmentation loss based on mixed labels (novel ground-truths and pseudo-labels). The total loss is:

$$\mathcal{L}_{novel} = \mathcal{L}_{seg} + \mathcal{L}_{trans}. \tag{8}$$

At inference time, our model is able to predict both old and new categories *i.e.,* given a point cloud $\mathbf{P}$ as input, the novel model $\mathcal{M}_{novel}$ will predict the category $\hat{c}$ such that $\hat{c} \in \{C_{base} \cup C_{novel}\}$, *i.e.,* $\underset{\hat{c}}{\mathrm{argmax}}\ Y_n^{\hat{c}}(E_n(\mathbf{P}))$.

4. Experiments

4.1. Datasets and Setup

Datasets. We evaluate our method on two existing public benchmarks S3DIS [1] and ScanNet [7]. **(a)** *S3DIS* contains point clouds from 272 rooms in 6 indoor areas. Each point contains xyz coordinates and RGB information, and is labelled with one of 13 classes. We use the more challenging area 5 as validation and the other areas as training. **(b)** *ScanNet* is an RGB-D video dataset having 1,513 scans in 707 indoor scenes. Each point is labelled with one of 21 classes (20 semantic classes and unannotated place). The 1210 scans in the dataset are used for training and other 312 scans for validation. For the training and validation splits, we follow the standard setting of datasets [1, 7] for point cloud semantic segmentation [28, 29, 39].

Setup. We follow and use the common disjoint setting [4] in 2D class-incremental segmentation, where the incremental training includes the old and current classes of point cloud, but not the future classes. In order to construct the 3D datasets to satisfy the requirements of 3D class-incremental segmentation, we adopt two paradigms to develop C_{base} and C_{novel}. One (S^0) where classes are incrementally introduced as per their original class label order in the dataset, and the other (S^1) introduces classes in an alphabetical order. Details are provided in the supplementary materials. In the experiments, the number of novel classes C_{novel} is set to 5, 3 and 1 respectively.

For 3D point clouds, we apply a sliding window [29, 38] to divide the rooms of *S3DIS* and *ScanNet* into 7,547 and 36,350 $1m{\times}1m$ blocks respectively, and randomly sample 2048 points in each block as input. D_{base} consists of point cloud blocks containing at least 100 points for any classes in C_{base} but ignore the C_{novel} annotations. We follow a similar process to construct D_{novel}. It is worth noting that the same point cloud may appear in both D_{base} and D_{novel}, but they have different available annotations as per the training phase. We use the common mean intersection-over-union

Table 1. Experimental comparisons of 3D class-incremental segmentation methods on S3DIS dataset of S^0 and S^1 split. We apply the mIoU (%) as the evaluation metric. “BT”, “F&A”, “FT” in the table represents Base Training, Freeze and Add, Fine-Tuning respectively. “JT” denotes Joint Training on all base+novel classes at once. Asterisk (*) denotes traditional class-incremental methods EWC [19] and LwF [21] in our reproduction for 3D semantic segmentation. The joint training is treated as the upper bound, and the best results of incremental learning methods are in bold.

Methods	C_{novel}=5						C_{novel}=3						C_{novel}=1					
	S^0			S^1			S^0			S^1			S^0			S^1		
	0-7	8-12	*all*	0-7	8-12	*all*	0-9	10-12	*all*	0-9	10-12	*all*	0-11	12	*all*	0-11	12	*all*
BT	48.54	-	-	37.24	-	-	46.80	-	-	40.73	-	-	45.00	-	-	45.88	-	-
F&A	44.25	12.33	31.98	37.71	42.89	39.44	44.28	3.34	34.83	41.11	35.64	39.85	44.57	0.05	41.14	45.35	0.05	41.86
FT	34.96	30.25	33.15	10.99	50.67	26.53	28.87	31.56	29.49	17.83	54.69	26.34	29.44	29.52	29.45	23.80	5.74	22.41
EWC*	39.38	31.07	36.19	23.19	54.84	35.36	37.13	37.92	37.31	29.38	55.53	35.41	36.55	19.94	35.27	25.60	9.81	24.39
LwF*	44.55	35.01	40.88	32.83	55.19	41.43	43.07	38.34	41.98	37.69	54.73	41.62	39.94	35.50	39.60	32.16	18.26	31.09
Ours	**48.94**	**39.56**	**45.33**	**38.17**	**55.20**	**44.72**	**45.15**	**45.33**	**45.19**	**39.83**	**57.59**	**43.93**	**44.08**	**35.69**	**43.43**	**40.33**	**19.28**	**38.71**
JT	50.23	41.74	46.97	38.38	60.11	46.74	48.62	41.44	46.97	42.63	60.44	46.74	47.51	40.41	46.97	47.09	42.55	46.74

Table 2. Experimental comparisons of 3D class-incremental segmentation methods on ScanNet dataset of S^0 and S^1 split. We apply the mIoU (%) as the evaluation metric. “BT”, “F&A”, “FT” in the table represents Base Training, Freeze and Add, Fine-Tuning respectively. “JT” denotes Joint Training on all base+novel classes at once. Asterisk (*) denotes traditional class-incremental methods EWC [19] and LwF [21] in our reproduction for 3D semantic segmentation. The joint training is treated as the upper bound, and the best results of incremental learning methods are in bold.

Methods	C_{novel}=5						C_{novel}=3						C_{novel}=1					
	S^0			S^1			S^0			S^1			S^0			S^1		
	0-14	15-19	*all*	0-14	15-19	*all*	0-16	17-19	*all*	0-16	17-19	*all*	0-18	19	*all*	0-18	19	*all*
BT	37.73	-	-	29.30	-	-	34.03	-	-	30.84	-	-	31.57	-	-	30.78	-	-
F&A	36.06	1.77	27.48	25.25	18.72	23.62	32.58	0.86	27.82	26.95	7.37	24.02	30.99	0.95	29.49	30.41	0.01	28.89
FT	9.39	13.65	10.45	5.83	34.03	12.88	8.43	10.98	8.82	4.88	40.94	10.29	8.02	10.46	8.14	4.76	7.57	4.90
EWC*	17.75	13.22	16.62	14.93	33.30	19.52	15.70	11.74	15.11	8.78	31.74	12.22	15.66	6.76	15.21	12.24	8.84	12.07
LwF*	30.38	13.37	26.13	24.04	**37.88**	27.50	26.22	13.88	24.37	22.76	**42.34**	25.70	22.15	12.56	21.67	20.63	**13.88**	20.29
Ours	**34.16**	**13.43**	**28.98**	**26.04**	35.51	**28.41**	**28.38**	**14.31**	**26.27**	**28.79**	40.31	**30.52**	**25.74**	**12.62**	**25.08**	**24.16**	12.97	**23.60**
JT	38.13	16.63	32.76	30.81	38.79	32.81	35.46	17.44	32.76	31.65	39.38	32.81	33.53	18.08	32.76	32.91	30.76	32.81

(mIoU) as the evaluation metric. In our settings, we calculate the mIoU after the last incremental step for base classes C_{base}, novel classes C_{novel} and all classes C_{all} respectively.

4.2. Implementation Details and Baselines

Implementation details. We use DGCNN [38] as the feature extractor and train the model both on D_{base} and D_{novel} using a batch size 32 and Adam optimizer [18] with intial 0.001 learning rate and 0.0001 weight decay for 100 epochs. After 50 epochs, the learning rate decays by 0.5. For incremental training on D_{novel}, the new classes weights in classifier $Y(\cdot)$ are randomly initialized while the rest are initialized with the pre-trained model's weights on the base classes. We empirically select the nearest neighbors K=12 and T=12, we further set uncertainty threshold τ to 0.0065 and 0.0045 for S3DIS and ScanNet via cross-validation respectively. The r is set to 0.25. The point clouds used in the training are augmented by gaussian jitter and random rotation around the z-axis.

Baselines. To compare our approach, we design 4 baselines in 2 directions: **1)** *Direct adaptation methods.* Inspired by [46], we use the “Freeze and Add” and “Fine-Tuning” methods. The former freezes the base model and adds a novel classifier output layer when training on the D_{novel}, while the latter randomly initializes the new classifier last layer and joins the base model for fine-tuning. **2)** *Forgetting-prevention methods.* We adapt Elastic Weight Consolidation (EWC) [19] and Learning without Forgetting (LwF) [21] from classical incremental learning models to 3D point cloud incremental segmentation setting. All the above models do not involve the rehearsal or replay of old samples. Additionally, the model jointly trained on all classes is treated as the performance upper bound.

4.3. Experimental Results

Comparsion with baselines. Tab. 1 $\sim$ Tab. 2 show the performance of our method versus baselines on S^0 and S^1 split of S3DIS and ScanNet datasets. Under different C_{novel} settings, we report the mIoU on the old and the new classes after incremental learning (*e.g.* the old 0-7 and the new 8-

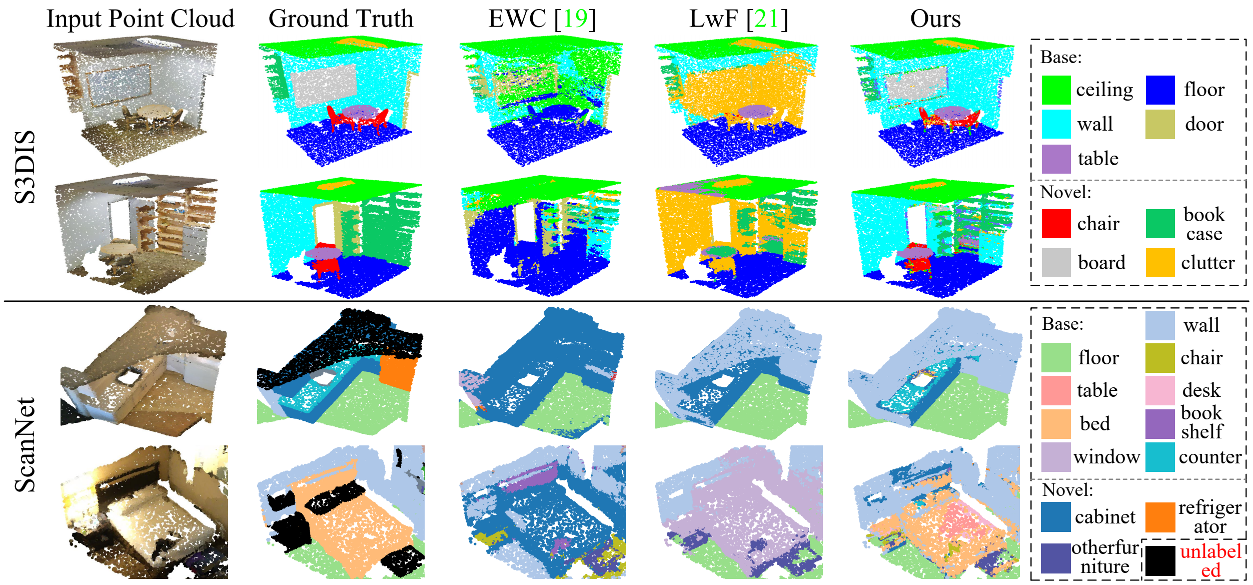


Figure 4. Qualitative comparison with the forgetting-prevention methods EWC [19] and LwF [21] on S3DIS and ScanNet datasets of $C_{novel} = 5$ in S^0 split. Only the base and novel classes included in current point cloud scenario are explained in the legend. Results in black on ScanNet dataset represent unlabeled and do not belong to either the base or novel classes.

12 with C_{novel}=5 on S3DIS dataset). From the tables, we observe that the results of the two direct adaptation methods are relatively unbalanced. For the freeze and add, while the performance on C_{base} is maintained due to the frozen base model, it is difficult to adapt to the new classes, resulting in poor performance on C_{novel}. The other fine-tuning method learns new classes by updating the parameters of both base feature extractor and the randomly initialized new classifier. Nevertheless, due to the lack of any measures to prevent forgetting, large degradation occurs in C_{base}. Since the adapted EWC [19] method introduces the weight regularization loss which restricts the modification of important weights for previous tasks, it can reduce the forgetting of the old tasks compared with fine-tuning. Meanwhile, LwF [21] leverages the output probability of previous model to guide the new classes learning, thus alleviating forgetting. Considering the overall mIoU, our method consistently achieves the best results on all the datasets. We note that our method is only 1.64% lower than joint training on S3DIS dataset and 3.78% on ScanNet dataset of S^0 split of C_{novel}=5. In addition, it is worth noting that our method is slightly lower than LwF on novel classes of ScanNet in S^1 split. It may be due to the fact that we introduce more constraints to preserve the old information. Fig. 4 shows the qualitative comparison of our method with EWC and LwF. Our approach strikes a balance between keeping the knowledge of base classes and learning the novel classes.

Comparison by changing the classes order. To study if the order of classes effects the results, we conduct several experiments on different splits (*i.e.*, S^0 and S^1). By comparing S^0 and S^1 results in Tab. 1 and Tab. 2, we can notice that, for the non-incremental joint training baseline, there is no difference in the performance since it is trained with all the labeled data at once. But for other incremental methods, the overall performance variations are observed, especially in C_{novel}=1 on S3DIS dataset. The overall performance of our model varies *e.g.* 43.43% (S^0) and 38.71% (S^1). These experiments suggest that the initial knowledge learned on the base classes has an impact on the subsequently introduced classes while learning incrementally. Nevertheless, amongst compared incremental learning approaches, our method still achieves the best mIoU under different orders of classes.

Table 3. Individual contributions of modules. C_{novel}=5 on S3DIS (S^1) and ScanNet (S^0). FT, GFT and UPG represent Fine-Tuning, Geometry-aware Feature-relation Transfer and Uncertainty-aware Pseudo-label Generation. mIoU (%) used as the metric.

FT	GFT	UPG	S3DIS (S^1)			ScanNet (S^0)		
			0-7	8-12	*all*	0-14	15-19	*all*
✓	×	×	10.99	50.67	26.53	9.39	13.65	10.45
✓	✓	×	14.86	53.33	29.98	12.50	11.54	12.26
✓	×	✓	35.12	55.18	42.84	31.91	13.41	27.28
✓	✓	✓	**38.17**	**55.20**	**44.72**	**34.16**	**13.43**	**28.98**

4.4. Ablation Study

Individual contributions of different modules. Tab. 3 shows contributions of proposed modules on S3DIS (S^1) and ScanNet (S^0) for C_{novel}=5. From the results, we oberverse that adding both individual GFT and UPG modules

Table 4. Multi-step incremental segmentation of overlapped setting on S3DIS datasets in S^0 split. We use mIoU (%) as the evaluation metric. The first 8 classes are base, and the remaining 5 classes are novel. Instead of the incremental procedure in the disjoint setting of C_{novel}=5, We use multi-step increments, each step increments 1 class, total increments 5 times.

	0	1	2	3	4	5	6	7	8	9	10	11	12	*all*	*base*	*novel*
Base Training	88.74	96.58	73.30	0.00	6.76	40.60	17.61	64.70	-	-	-	-	-	-	48.54	-
Step 1	88.09	95.63	73.46	0.00	7.93	39.70	22.76	63.06	34.42	**-**	**-**	**-**	**-**	47.23	48.83	34.42
Step 2	85.46	95.66	71.57	0.00	0.90	32.01	19.74	50.07	13.69	3.73	**-**	**-**	**-**	37.28	44.43	8.71
Step 3	85.72	95.76	72.07	0.00	0.81	34.79	11.74	51.78	12.32	3.84	44.31	**-**	**-**	37.56	44.08	20.16
Step 4	85.91	95.37	65.17	0.00	0.00	31.28	6.82	44.29	0.21	5.04	40.10	8.02	**-**	31.85	41.11	13.34
Final Step 5	88.04	95.83	65.89	0.00	0.00	34.48	8.01	44.38	0.07	3.56	36.18	10.63	33.50	32.35	42.08	16.79

Table 5. Comparsion on incremental classification across datasets under different backbones. We follow [6] setting and metrics to conduct evaluation. The Acc_o^* denotes the base model's accuracy, Acc_o and Acc_n represents the accuracy on base and novel classes using the final incremental model. $\triangle = \frac{Acc_o^* - Acc_o}{Acc_o^*} \times 100\%$, the lower $\triangle$ represents less forgetting of the novel model. * denotes our reproduced baseline. The best results are in bold.

Backbone	Methods	ModelNet40 →ScanObjectNN				ModelNet40 →ModelNet10			
		Acc_o^*↑	Acc_o↑	Acc_n↑	△↓	Acc_o^*↑	Acc_o↑	Acc_n↑	△↓
DGCNN [38]	lwf-3D [6]*	92.91	73.34	79.41	21.06	91.71	87.14	93.32	4.98
	+GFT	92.91	76.31	81.19	17.87	91.71	88.95	93.32	3.01
	+GFT+UPG	**92.91**	**78.19**	**82.82**	**15.84**	**91.71**	**88.99**	**93.86**	**2.97**
PointNet [28]	lwf-3D [6]*	90.14	84.77	76.87	5.96	88.71	81.59	90.41	8.03
	+GFT	90.14	84.09	77.15	3.92	88.71	82.23	90.43	7.30
	+GFT+UPG	**90.14**	**86.84**	**79.12**	**3.66**	**88.71**	**83.18**	**91.27**	**6.20**
PointConv [39]	lwf-3D [6] *	92.69	87.19	79.33	5.93	91.26	83.59	92.13	8.41
	+GFT	92.69	88.32	79.75	4.71	91.26	83.80	92.03	8.18
	+GFT+UPG	**92.69**	**88.79**	**80.08**	**4.21**	**91.26**	**84.65**	**93.32**	**7.24**

results in performance gain. The UPG contributes the most to the improvement, suggesting that it can effectively solve the serious semantic shift of previous classes. Moreover, comparing the results of the first two rows or the last two rows, it can be found that the performance of the model on the base classes has been significantly increased. This phenomenon indicates that the GFT module can capture the feature relations according to the point structure to reduce catastrophic forgetting of old knowledge. The combination of two modules achieves the best results, where we observe gains over "FT" by 18.19% and 18.53% in terms of mIoU on S3DIS and ScanNet respectively.

Effects of multi-step increments. We further construct a more complicated *overlapped* setting [4] beyond *disjoint*, where the points at the current incremental episode may belong to the old, the current and the future classes. Tab. 4 shows the results of our method for challenging multi-step increments on S3DIS dataset of S^0 split. We notice that due to the adoption of more steps (5-step) in incremental learning, the model gradually forgets the previous knowledge, so the final overall mIoU is less compared to C_{novel}=5 in Tab. 1. Furthermore, we observe performance degradation in the incremental classes added earlier. We argue that this is because in the subsequent incremental classes learning, the model is forced to not only confront with the forgetting of old classes, but also to deal with the background semantic shift of both old and the unknown future classes, which confuses the model in recognition of previous categories.

Cross-dataset incremental classification. To further evaluate the effectiveness of our proposed modules, we perform additional cross-dataset incremental classification experiments following [6] setting. We can simply extend our method to incremental object recognition from dataset ModelNet40 [40]→ScanObjectNN [37] and ModelNet40 [40]→ModelNet10 [40]. In Tab. 5, we implement the lwf-3D [6] method without semantic embeddings in our reproduction as baseline, and we gradually add the designed GFT and UPG modules. It shows that introducing two modules significantly improve the performance across various backbone architectures, especially in terms of Acc_n and $\triangle$ metrics. This phenomenon fully demonstrates that the adopted modules are able to effectively reduce the forgetting of old classes information and continually learn new classes.

5. Conclusion

In this paper, we propose a class-incremental learning approach to semantic segmentation of 3D point cloud. Due to the disordered and unstructured nature of point cloud, continual migration and retention of learned knowledge becomes a significant challenge. We exploit point-wise associations based upon the geometric structure of point cloud to distill knowledge between models. Further, to tackle the semantic shift issue where labels for old class points get collapsed into background, we develop an uncertainty-aware pseudo-labelling strategy, that helps preserve learned knowledge on previous classes. Our extensive experiments show the promises of our approach to alleviate catastrophic forgetting, where our achieved overall performance is close to joint training (upper bound). Our model is currently only evaluated for indoor 3D point cloud data. We leave exploration to more challenging outdoor scenes as future work. In addition, we will further study the class-incremental segmentation of 3D point cloud under few-shot setting.

Acknowledgement: This work was supported by the National Natural Science Foundation of China (No.62276176).

References

[1] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1534–1543, 2016. 2, 5

[2] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 233–248, 2018. 1, 2

[3] Fabio Cermelli, Dario Fontanel, Antonio Tavera, Marco Ciccone, and Barbara Caputo. Incremental learning in semantic segmentation from image labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4371–4381, 2022. 1, 2

[4] Fabio Cermelli, Massimiliano Mancini, Samuel Rota Bulo, Elisa Ricci, and Barbara Caputo. Modeling the background for incremental learning in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9233–9242, 2020. 1, 2, 5, 8

[5] Townim Chowdhury, Ali Cheraghian, Sameera Ramasinghe, Sahar Ahmadi, Morteza Saberi, and Shafin Rahman. Few-shot class-incremental learning for 3d point cloud objects. *arXiv preprint arXiv:2205.15225*, 2022. 2, 3

[6] Townim Chowdhury, Mahira Jalisha, Ali Cheraghian, and Shafin Rahman. Learning without forgetting for 3d point cloud objects. In *International Work-Conference on Artificial Neural Networks (IWANN)*, pages 484–497. Springer, 2021. 2, 3, 8

[7] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5828–5839, 2017. 2, 5

[8] Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. Learning without memorizing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5138–5146, 2019. 1, 2

[9] Jiahua Dong, Yang Cong, Gan Sun, Bingtao Ma, and Lichen Wang. I3dol: Incremental 3d object learning without catastrophic forgetting. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 35, pages 6066–6074, 2021. 2

[10] Arthur Douillard, Yifu Chen, Arnaud Dapogny, and Matthieu Cord. Plop: Learning without forgetting for continual semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4040–4050, 2021. 1, 2

[11] Sayna Ebrahimi, Mohamed Elhoseiny, Trevor Darrell, and Marcus Rohrbach. Uncertainty-guided continual learning in bayesian neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 75–78, 2019. 1, 2

[12] Chrisantha Fernando, Dylan Banarse, Charles Blundell, Yori Zwols, David Ha, Andrei A Rusu, Alexander Pritzel, and Daan Wierstra. Pathnet: Evolution channels gradient descent in super neural networks. *arXiv preprint arXiv:1701.08734*, 2017. 2

[13] Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999. 2

[14] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning (ICML)*, pages 1050–1059. PMLR, 2016. 4

[15] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011. 4

[16] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11108–11117, 2020. 1, 3

[17] Ronald Kemker and Christopher Kanan. Fearnet: Brain-inspired model for incremental learning. *arXiv preprint arXiv:1711.10563*, 2017. 1, 2

[18] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[19] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences (NAS)*, 114(13):3521–3526, 2017. 1, 2, 6, 7

[20] Vinod K Kurmi, Badri N Patro, Venkatesh K Subramanian, and Vinay P Namboodiri. Do not forget to attend to uncertainty while mitigating catastrophic forgetting. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 736–745, 2021. 1, 2

[21] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 40(12):2935–2947, 2017. 1, 2, 6, 7

[22] Yuyang Liu, Yang Cong, Gan Sun, Tao Zhang, Jiahua Dong, and Hongsen Liu. L3doc: Lifelong 3d object classification. *IEEE Transactions on Image Processing (TIP)*, 30:7486–7498, 2021. 2

[23] Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7765–7773, 2018. 2

[24] Oleksiy Ostapenko, Mihai Puscas, Tassilo Klein, Patrick Jahnichen, and Moin Nabi. Learning to remember: A synaptic plasticity driven framework for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11321–11329, 2019. 1, 2

[25] Firat Ozdemir, Philipp Fuernstahl, and Orcun Goksel. Learn the new, keep the old: Extending pretrained models with new

anatomy and images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 361–369. Springer, 2018. 2

[26] Firat Ozdemir and Orcun Goksel. Extending pretrained segmentation networks with additional anatomical structures. *International Journal of Computer Assisted Radiology and Surgery ((IJCARS)*, 14(7):1187–1195, 2019. 2

[27] Chao Qi, Jianqin Yin, Huaping Liu, and Jun Liu. Neighborhood spatial aggregation based efficient uncertainty estimation for point cloud semantic segmentation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 14025–14031. IEEE, 2021. 4

[28] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 652–660, 2017. 1, 3, 5, 8

[29] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5099–5108, 2017. 1, 3, 5

[30] Haoxuan Qu, Hossein Rahmani, Li Xu, Bryan Williams, and Jun Liu. Recent advances of continual learning in computer vision: An overview. *arXiv preprint arXiv:2109.11369*, 2021. 2

[31] Jathushan Rajasegaran, Munawar Hayat, Salman Khan, Fahad Shahbaz Khan, and Ling Shao. Random path selection for incremental learning. *Advances in Neural Information Processing Systems*, 3, 2019. 2

[32] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2001–2010, 2017. 1, 2

[33] Anthony Robins. Catastrophic forgetting, rehearsal and pseudorehearsal. *Connection Science*, 7(2):123–146, 1995. 2

[34] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. 2

[35] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017. 1, 2

[36] Sebastian Thrun. Lifelong learning algorithms. In *Learning to learn*, pages 181–209. Springer, 1998. 2

[37] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1588–1597, 2019. 8

[38] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (TOG)*, 38(5):1–12, 2019. 1, 3, 5, 6, 8

[39] Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9621–9630, 2019. 1, 3, 5, 8

[40] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1912–1920, 2015. 8

[41] Guanglei Yang, Enrico Fini, Dan Xu, Paolo Rota, Mingli Ding, Moin Nabi, Xavier Alameda-Pineda, and Elisa Ricci. Uncertainty-aware contrastive distillation for incremental semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2022. 1, 2

[42] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. In *International Conference on Learning Representations (ICLR)*, 2018. 2

[43] Maciej Zamorski, Michał Stypułkowski, Konrad Karanowski, Tomasz Trzciński, and Maciej Zięba. Continual learning on 3d point clouds with random compressed rehearsal. *arXiv preprint arXiv:2205.08013*, 2022. 2, 3

[44] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning (ICML)*, pages 3987–3995. PMLR, 2017. 1, 2

[45] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16259–16268, 2021. 1, 3

[46] Na Zhao and Gim Hee Lee. Static-dynamic co-teaching for class-incremental 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 36, pages 3436–3445, 2022. 3, 6