On-the-fly Category Discovery for LiDAR Semantic Segmentation

Hyeonseong Kim⋆, Sung-Hoon Yoon⋆, Minseok Kim, and Kuk-Jin Yoon

Visual Intelligence Lab., KAIST, Korea
{brian617,yoon307,alstjrx1x1,kjyoon}@kaist.ac.kr

Abstract. LiDAR semantic segmentation is important for understanding the surrounding environment in autonomous driving. Existing methods assume closed-set situations with the same training and testing label space. However, in the real world, unknown classes not encountered during training may appear during testing, making it difficult to apply existing methodologies. In this paper, we propose a novel *on-the-fly category discovery* method for LiDAR semantic segmentation, aiming to classify and segment both unknown and known classes instantaneously during test time, achieved solely by learning with known classes in training. To embed instant segmentation capability in an inductive setting, we adopt a hash coding-based model with an expandable prediction space as a baseline. Based on this, *dual prototypical learning* is proposed to enhance the recognition of the known classes by reducing the sensitivity to intra-class variance. Additionally, we propose a novel *mixing-based category learning* framework based on representation mixing to improve the discovery capability of unknown classes. The proposed mixing-based framework effectively models out-of-distribution representations and learns to semantically group them during training while distinguishing them from in-distribution representations. Extensive experiments on SemanticKITTI and SemanticPOSS datasets demonstrate the superiority of the proposed method compared to the baselines. The code is available at `https://github.com/hskim617/OCDSS`.

Keywords: Autonomous driving · LiDAR semantic segmentation · On-the-fly category discovery

1 Introduction

LiDAR semantic segmentation is crucial for recognizing the surrounding environment in autonomous driving and has recently made significant progress [2, 11, 12, 26, 30, 38, 39, 50, 59, 65]. Although these models demonstrate promising results across various datasets, their effectiveness is constrained to closed-set settings, as they are assumed and trained within such scenarios. In real-world scenarios, however, unknown classes are bound to appear during testing, making it incapable of predicting unknown classes for closed-set models as in Fig. 1a.

⋆ Equal contribution.

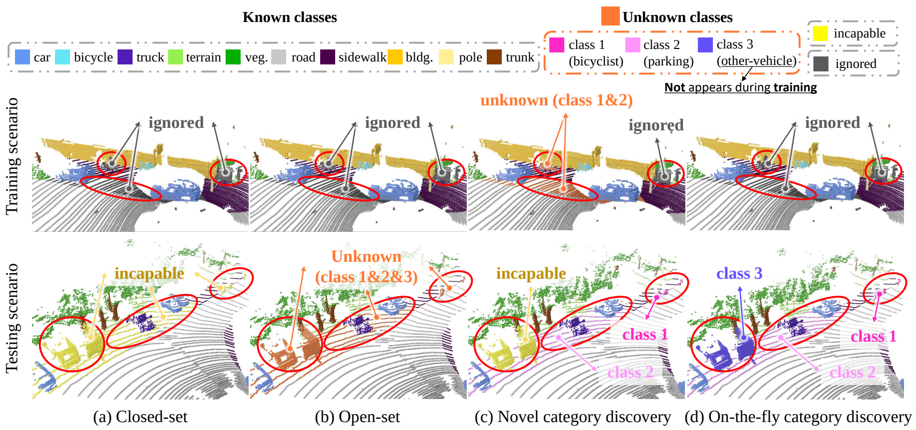


Fig. 1: Illustration of training and testing scenarios for on-the-fly category discovery in LiDAR semantic segmentation and relevant tasks. The goal of each task is (a) recognizing known classes, (b) further identifying and rejecting unknown classes, (c) learning clustering using unlabeled, unknown class data, and (d) cognizing unknown classes that are not identified during training while recognizing known classes.

To address this challenge, open-set recognition (OSR) [47], which aims to filter out unknown classes has been explored in diverse fields, including image classification [3,5,9,49,58], semantic segmentation [7,31,37,57], and 3D semantic segmentation [8,35]. Considering the nature of the human cognitive system that can cognize novel visual concepts based on understanding, simply rejecting unknown classes as in OSR (Fig. 1b) is unsatisfactory.

To narrow the gap between humans and machines by learning novel concepts, novel category discovery (NCD) [15,23,24,28,63] and on-the-fly category discovery (OCD) [14] are emerging as new research fields and attracting attention. Given labeled data of known classes and unlabeled data of unknown classes, the goal of NCD is to cluster the unlabeled data with the same semantics, aiming to cognize them as in Fig. 1c. Therefore, NCD is formulated as *transductive* learning in which data of unknown classes are available, making it incapable of predicting unknown classes not identified during training. On the other hand, OCD aims for a more realistic scenario by learning solely from data of known classes and *inductively* cognizing unknown classes at test time. Furthermore, during testing, the model should independently classify each test sample *instantaneously*, regardless of the distribution of the test data.

In this paper, we broaden the scope of OCD to LiDAR point clouds and introduce a new problem, on-the-fly category discovery for LiDAR semantic segmentation (OCDSS). As shown in Fig. 1d, our goal is to classify and segment both known and unknown classes *instantaneously* in an *inductive* manner during test time, achieved solely by learning with known classes in training. To achieve the goal, one straightforward solution is adopting the previous approach in image classification [14] and applying it to the LiDAR semantic segmentation.

Therefore, we implemented the hash coding-based model proposed by [14] as a baseline for LiDAR point clouds. This model utilizes hash codes as class descriptors, generated by binarizing each dimension of the learned representation based on its sign. However, extending it to the LiDAR domain entails two limitations.

Firstly, due to LiDAR's sparse nature and lack of texture information, hash coding in learned representations is more sensitive to intra-class variance. In practice, we observed that the method for reducing sensitivity to intra-class variance by disentangling sign and magnitude [14] had shown unsatisfactory results in the LiDAR domain (refer Fig. 4a). Secondly, prior work relies on representation learning from known classes to discover unknown classes. Without a dedicated mechanism for discovering unknown classes, only a tentative expectation that the learned representations will form semantically meaningful groups for unknown classes is possible. This reliance, combined with the aforementioned limitation, complicates OCDSS.

Therefore, we propose a novel learning method for OCDSS that learns representations robust to intra-class variance in hash coding and suitable for unknown class discovery. First, we propose *dual prototypical learning* (DPL) that constrain the representation space by introducing extra hash prototypes, effectively reducing the sensitivity of hash codes to intra-class variance. Specifically, the hash prototypes enforce additional constraints to ensure that representations within the same intra-class share the same hash code, dramatically enhancing recognition capability for known classes. Second, we propose a novel *mixing-based category learning* (MCL) framework to improve the ability to cognize unknown classes. The proposed framework consists of three modules: a mixing module, a discrimination module, and a grouping module. The mixing module models out-of-distribution (OOD) representation through representation mixing and obtains mixed representation to mimic unknown classes during training. The discrimination module then enhances the model's OOD discrimination ability by learning to distinguish the representation of known classes from mixed representations. In addition, the grouping module develops the model's clustering capability for OOD representations by learning to cluster mixed representations within the same semantics. Meanwhile, we integrate the hash prototypes of DPL during MCL to fully exploit the discovering capability for unknown classes. Finally, we introduce an evaluation protocol for OCDSS and evaluate our approach on SemanticKITTI [4] and SemanticPOSS [42] datasets. Extensive experiments demonstrate the superiority of our method compared to baselines, while ablation studies show the effectiveness of each component.

Our main contributions are four-fold: (I) We introduce a new problem of on-the-fly category discovery for LiDAR semantic segmentation (OCDSS). (II) We propose dual prototypical learning that effectively learns representations that are less sensitive to intra-class variation in hash coding. (III) We present a novel mixing-based category learning framework, enabling the model to discover unknown classes. (IV) We introduce a new evaluation protocol for OCDSS and demonstrate the superiority of our method through extensive experiments.

2 Related Work

LiDAR Semantic Segmentation. LiDAR semantic segmentation is the task of inferring semantic labels to individual 3D points within the LiDAR point cloud. These approaches can be broadly categorized into point-based, projection-based, and voxel-based according to the way they represent and process point clouds. Point-based methods [26,36,44,51] directly operate on points in line with the foundational work PointNet [43] by using multi-layer perceptrons. Despite their advantages, such as minimal information loss and few parameters, applying these methods to large-scale LiDAR point clouds demands heavy computational resources and memory. Projection-based methods have been proposed to address efficiency and leverage advanced 2D convolutional neural network architectures. These methods project LiDAR points onto a range image [1,2,10,13,19,30,39,55,56] using spherical projection or a bird-eye-view [59]. However, the projection of 3D points into 2D representation might lose some geometric information. Voxel-based methods represent point clouds as 3D voxels and process them through 3D convolution [66] or sparse convolution [11, 12, 20, 38, 50, 65]. In particular, sparse convolution achieves high performance with less computation, so we adopt MinkowskiNet [12] as our backbone architecture.

Open-set Recognition. Open-set recognition (OSR) was first introduced in [47], extending the closed-set assumption to realistic scenarios where unknown classes appear during testing. As OSR aims to detect unknown classes, previous studies estimate the unknown probability based on extreme value theory [5] or estimate uncertainty through maximum softmax probability [25], maximum logit [3], or Bayesian inference approximation [16,33]. Other approaches include training a redundancy classifier for unknown classes [54,64] or using metric learning [9, 22, 48]. Meanwhile, generative model-based approaches employ generative adversarial networks to synthesize unknown class samples [18, 40] or use image reconstruction error for the criterion of determining unknown classes [41,49,58]. Unlike these methods for object recognition, studies for semantic segmentation [7, 31, 37, 57], panoptic segmentation [17, 27] in open-set situations have also been proposed. Recently, open-set semantic segmentation for LiDAR point clouds has emerged [8,35], extending an open-set problem to the LiDAR domain. While OSR methods are aware of unknown classes, they focus on identification, whereas we aim to go beyond achieving semantically cognizing them.

Novel Category Discovery. Novel category discovery (NCD), first formalized in [24], is another relevant task that aims to cognize unknown classes in unlabeled data by transferring the knowledge from known classes in labeled data. After DTC [24] proposed deep transfer clustering for NCD, several studies were suggested, achieving significant progress [15, 21, 23, 28, 60, 62, 63]. These approaches first learn representations using supervised learning from labeled known classes, and then discover unknown classes by transferring knowledge through pseudo-labeling on unknown classes based on pair-wise similarity [23, 28, 60, 62, 63] or cluster assignment [15, 21]. Additionally, EUMS [61] and NOPS [45] extend the NCD problem to semantic segmentation in images and LiDAR point clouds, respectively. Nevertheless, NCD usually assumes the number of unknown classes as prior knowledge [15, 23, 28, 60, 62, 63], and presuppose all unlabeled data as unknown classes. Hence, generalized category discovery (GCD) was introduced where unlabeled data contain both known and unknown classes [6,52]. However, as pointed out in [14], two limitations in prior settings still exist: 1) transductive learning and 2) offline evaluation. In prior settings, category discovery is formulated as transductive learning in which data of unknown classes are available during training. Also, offline evaluation constrains the results depending on the entire test data distribution, making instant feedback on each test sample impossible. Therefore, [14] proposed a new task named on-the-fly category discovery (OCD), which aims to recognize known classes and cognize unknown classes instantaneously in an inductive manner. In this study, we tackle the challenging OCD for LiDAR semantic segmentation, which instantaneously segments LiDAR point clouds from both known and unknown classes while learning exclusively from known classes.

3 Method

In Sec. 3.1, we first elaborate on our problem formulation of on-the-fly category discovery for LiDAR semantic segmentation (OCDSS). In Sec. 3.2, we introduce a hash coding-based baseline model for OCDSS that embeds expandable prediction space and instant inference capability within an inductive setting. Then, in Sec. 3.3, we describe our dual prototypical learning for reducing the sensitivity of representation in hash coding, improving the closed-set performance. In Sec. 3.4, we propose a mixing-based category learning framework to enable the model to possess category discovery capabilities for both known and unknown classes. Fig. 2 shows the overview of the proposed method.

3.1 Problem Formulation

In this section, we formulate the problem of OCDSS, which aims to recognize known classes and cognize unknown classes during testing in an inductive manner. Let $D_S = \{P_S\}$ be a support set used for training and $D_Q = \{P_Q\}$ be a query set for testing. Then, we have 3D point clouds $P_S = \{(x, y)\} \in \mathbb{R}^3 \times Y_S$ and $P_Q = \{(x, y)\} \in \mathbb{R}^3 \times Y_Q$, representing a single LiDAR scan from D_S and D_Q, respectively. Here, x denotes each point, and y is the corresponding label. Y_S and Y_Q are the label spaces of support and query sets. Following the setting of OCD [14], Y_S defines the known classes observed during training, while Y_Q contains both known and unknown classes that appear at testing, *i.e.* $Y_S \subset Y_Q$. We denote the number of known and unknown classes as $C_k = |Y_S|$ and $C_u = |Y_Q \backslash Y_S|$, where we assume C_u is unknown. This is different from existing NCD approaches [15,23,28,45,60,62,63] that assume C_u as a known prior. The goal is to train a model using only D_S so that the model can inductively classify and segment point clouds in D_Q instantaneously.

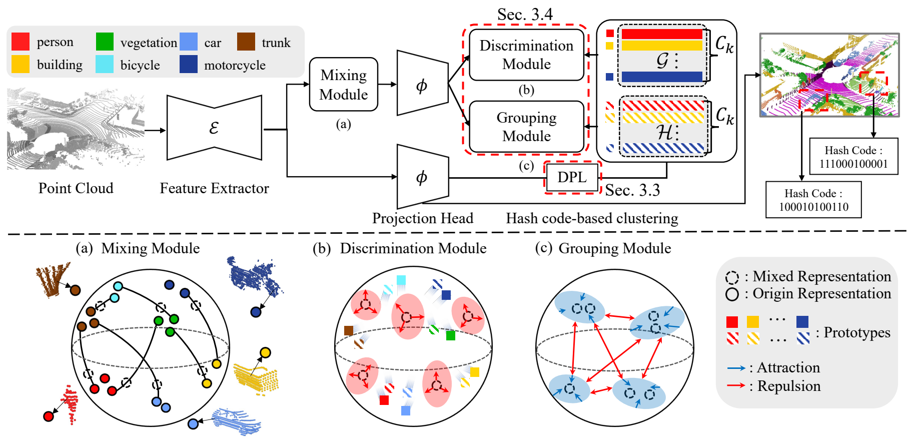


Fig. 2: Overview of the proposed framework. Given the LiDAR point cloud, the model extracts the features through feature extractor $\mathcal{E}$, which are then mapped into the prediction space to obtain the representations by the projection head ϕ. Through dual prototypical learning (DPL), the model learns representations suitable for hash code-based clustering, enhancing its recognition ability for known classes. Besides, mixing-based category learning (MCL) improves the model's cognition capability for unknown classes through three modules: mixing module, discrimination module, and grouping module. The final prediction is obtained by the hash code-based clustering.

3.2 Hash Coding-based Baseline Model

We first introduce a hash coding-based baseline model for OCDSS. To embed the expandable prediction space and instant inference capability while learning the model inductively, we adopt the basic pipeline of hash coding-based model from [14]. Formally, our baseline model consists of a feature extractor $\mathcal{E}$ and a projection head ϕ. Given point cloud $P_S = \{(x, y)\}_{i=1}^{N}$, the feature extractor first extracts the feature representation f_i for each point x_i. The projection head maps f_i to the prediction space, obtaining the projected representation $p_i = \phi(f_i) \in \mathbb{R}^L$, which is used for the hash coding. Here, L is the dimension of the hash code. Then, the hash code for point x_i is obtained by $hash(\phi(\mathcal{E}(x_i)))$, where $hash$ is a hashing operation. For an arbitrary vector $\mathbf{v} = [v_1, \ldots, v_l, \ldots, v_L] \in \mathbb{R}^L$, the hashing operation is defined as,

$$hash(\mathbf{v}) = [v_1^*, \ldots, v_l^*, \ldots, v_L^*], v_l^* = \begin{cases} 1, & v_l \geq 0 \\ 0, & v_l < 0 \end{cases}. \tag{1}$$

To predict classes of points from P_Q during testing, we use the hash code as a class descriptor where points with the same code are considered to be in the same class. This hash coding-based inference allows the model to have an expanded label space to predict new classes beyond the known classes in an inductive setting. Moreover, it also has the advantage of being able to immediately predict the result of each test sample individually.

For representation learning of the baseline model, [14] used supervised contrastive learning [29]; however, there is a lack of clear design choices for representation learning in LiDAR point clouds. For example, in the case of contrastive learning where pair-wise computation between samples is required, it is not suitable due to its high demand for computation and memory. Instead, we use a standard cross-entropy loss on the known classes. To learn a more discriminative representation, we employ parametric prototypes $\mathcal{G} = \{g_1, g_2, \ldots, g_{C_k}\} \in \mathbb{R}^{C_k \times L}$ for known classes as in [7] and replace the classifier as cosine similarity between the prototype and the projected representation p_i. Specifically, we compute the distance based on the cosine similarity as,

$$d_{cos}(p_i, g_j) = 1 - \frac{p_i \cdot g_j}{\|p_i\| \|g_j\|}. \tag{2}$$

Then, let us define $\sigma_{\mathcal{G}}(p_i, y_i)$ that computes the softmax probability of p_i with respect to $\mathcal{G}$ as follows:

$$\sigma_{\mathcal{G}}(p_i, y_i) = \frac{\exp(-d_{cos}(p_i, g_{y_i})/\tau_{cos})}{\sum_{j=1}^{C_k} \exp(-d_{cos}(p_i, g_j)/\tau_{cos})}, \tag{3}$$

where g_{y_i} denotes the prototype corresponding to the label y_i and τ_{cos} is a temperature parameter. The loss function for the baseline model is defined as,

$$\mathcal{L}_{base} = \sum_i - \log \sigma_{\mathcal{G}}(p_i, y_i). \tag{4}$$

3.3 Dual Prototypical Learning

The baseline model is trained to generate similar representations for points belonging to the same class. However, during inference, points are classified into different classes if they do not share the same hash code, which can be sensitive to intra-class variance. Due to the sparse nature and lack of texture information in LiDAR, hash coding in learned representations becomes more sensitive to intra-class variance compared to images. To address this issue, we propose dual prototypical learning (DPL) to learn the representation robust to intra-class variance in hash coding. Specifically, DPL employs extra parametric prototypes to constrain the representation space, denoted as $\mathcal{H} = \{h_1, h_2, \ldots, h_{C_k}\} \in \mathbb{R}^{C_k \times L}$, referred to as hash prototypes. Similar to $\mathcal{G}$, we calculate the distance between the projected representation p_i and the hash prototypes. In this case, based on the fact that the sign of each dimension in representation determines the hash code, we utilize the L2 distance between signs instead of cosine similarity. In practice, for stable optimization, we use the hyperbolic tangent function, $tanh$, instead of the sign operation as,

$$d_{hash}(p_i, h_j) = \|tanh(p_i) - tanh(h_j)\|_2^2 / L. \tag{5}$$

Similar to $\sigma_{\mathcal{G}}$ in Eq. (3), we define softmax probability $\sigma_{\mathcal{H}}(p_i, y_i)$ as follows:

$$\sigma_{\mathcal{H}}(p_i, y_i) = \frac{\exp(-d_{hash}(p_i, h_{y_i})/\tau_{hash})}{\sum_{j=1}^{C_k} \exp(-d_{hash}(p_i, h_j)/\tau_{hash})}, \tag{6}$$

where h_{y_i} denotes the hash prototype corresponding to the label y_i and τ_{hash} is a temperature parameter. The prototypical learning loss based on hash prototypes is defined as,

$$\mathcal{L}_{hash} = \sum_i -\log \sigma_{\mathcal{H}}(p_i, y_i). \tag{7}$$

Along with $\mathcal{L}_{base}$, the final loss function of dual prototypical learning is as follows:

$$\mathcal{L}_{DPL} = \mathcal{L}_{base} + \mathcal{L}_{hash}. \tag{8}$$

With the proposed DPL, we impose additional constraints on representations to make hash codes less sensitive to intra-class variance, improving the model's recognition capability.

3.4 Mixing-based Category Learning

Along with the recognition, our goal is to train the model to cognize unknown classes through semantic understanding. Our basic idea to achieve this goal is to mimic unknown classes during training and learn the model to distinguish and group them from known classes for a comprehensive understanding of unknown semantics. To this end, we propose mixing-based category learning (MCL), which consists of a mixing module, a discrimination module, and a grouping module.

Mixing Module. We mix the representations of different known classes to mimic the unknown classes during training. Our intention in using representation mixing is that 1) it can effectively model OOD representations, as stated in [64], and 2) it can further mimic diverse unknown classes by modeling OOD representations that share common semantics depending on which classes are mixed. Specifically, we use manifold mixup technique [53] to mix two feature representations f_i and f_j of different classes y_i and y_j, where $y_i \neq y_j$. The mixed representation $\overline{f_{i,j}}$ is calculated as,

$$\overline{f_{i,j}} = r f_i + (1-r) f_j, \tag{9}$$

where r is a mixing ratio. We can also define the imaginary corresponding label $\overline{y_{i,j}}$ of $\overline{f_{i,j}}$, referred to as a mix label. Note that if $(i,j) \neq (k,l)$, then $\overline{y_{i,j}} \neq \overline{y_{k,l}}$. These mixed representations are considered as OOD representations in the following modules.

Discrimination Module. In the discrimination module (DM), we train the model to distinguish mixed representations from prototypes. In particular, we learn parametric prototypes $\mathcal{G}$ and $\mathcal{H}$ to be discriminative from the mixed representation. To this end, we first project $\overline{f_{i,j}}$ into projection space and obtain $\phi(\overline{f_{i,j}})$. Then, we enforce $\phi(\overline{f_{i,j}})$ not to be close to the prototype of any specific class in the form of entropy maximization as follows:

$$\begin{aligned}\mathcal{L}_{disc} = \sum_{i,j} \Big(&\sum_k \sigma_{\mathcal{G}}(\phi(\overline{f_{i,j}}), k) \log \sigma_{\mathcal{G}}(\phi(\overline{f_{i,j}}), k) \\ &+ \sum_k \sigma_{\mathcal{H}}(\phi(\overline{f_{i,j}}), k) \log \sigma_{\mathcal{H}}(\phi(\overline{f_{i,j}}), k) \Big).\end{aligned} \tag{10}$$

Notably, we stop the gradient of $\phi(\overline{f_{i,j}})$ and only optimize prototypes. In addition, we incorporate mixed representations that are mixed after being projected into the $\mathcal{L}_{disc}$ to enhance discrimination for more diverse OOD representations.
Grouping Module. We propose a grouping module (GM) to enhance the model's understanding of the underlying semantics. Since the mixed representations might share common semantics depending on which classes are mixed, we can group them using the mix labels. For mathematical brevity, we omit the summation over i and j here. Specifically, we employ contrastive loss [29] and propose contrastive learning on mixed representation based on mixed labels:

$$\mathcal{L}_{gp} = \frac{-1}{|P(i,j)|} \sum_{p \in P(i,j)} \log \sigma_{A(i,j)}(\frac{-d_{cos}(\phi(\overline{f_{i,j}}), \phi(\overline{f_p}))}{\tau_{cos}}), \tag{11}$$

where $A(i,j)$ denotes all mixed representations excluding $\phi(\overline{f_{i,j}})$ and $P(i,j)$ is all positive pairs of $\phi(\overline{f_{i,j}})$ that share the same mix label. $\sigma_{(A(i,j))}$ refers to a softmax operation with respect to $A(i,j)$. In practice, we perform sub-sampling on $A(i,j)$ to reduce the computation from heavy pair-wise operations. Additionally, we impose the following constraint so that mixed representations with the same mix label share the hash code as,

$$\mathcal{L}_{gp\text{–}hash} = \log \sum_{p \in P(i,j)} \exp(d_{hash}(\phi(\overline{f_{i,j}}), \phi(\overline{f_p}))/\tau_{hash}). \tag{12}$$

The loss function for the proposed MCL is defined as,

$$\mathcal{L}_{MCL} = \mathcal{L}_{disc} + \lambda(\mathcal{L}_{gp} + \mathcal{L}_{gp\text{–}hash}), \tag{13}$$

where, λ is the weight for $\mathcal{L}_{gp}$ and $\mathcal{L}_{gp\text{–}hash}$.

Finally, the total loss function of our method is,

$$\mathcal{L}_{total} = \mathcal{L}_{DPL} + \mathcal{L}_{MCL}. \tag{14}$$

4 Evaluation Protocol

4.1 Data Split

Datasets. We evaluate our method on two datasets, SemanticKITTI [4] and SemanticPOSS [42]. SemanticKITTI consists of 43,552 frames with point-wise annotations of 19 classes. We follow the official split and use sequences 00 to 07 and 09 to 10 with 19,130 frames for training, and sequence 08 with 4,071 frames for validation. SemanticPOSS contains 2,988 frames with point-wise annotations of 13 classes. We use sequence 03 for validation, while the remaining sequences are used for training, following the official split.
Scenarios. We designed three scenarios considering various real-world situations. More precisely, we defined two unknown class types that can exist in the real world: *observed-unknown* and *unobserved-unknown*. *Observed-unknown*

Table 1: Data splits of *scenario-A* on SemanticPOSS and SemanticKITTI datasets. The POSS_i and the KITTI_i denotes the i^{th} split.

Split	Unknown classes	Split	Unknown classes
POSS_0	*trashcan, traffic sign, bike, plants*	KITTI_0	*bicycle, other-vehicle, trunk, terrain, vegetation*
POSS_1	*pole, person, building*	KITTI_1	*bicyclist, traffic-sign, other-ground, fence, road*
POSS_2	*rider, fence, ground*	KITTI_2	*motorcyclist, motorcycle, pole, car, sidewalk*
POSS_3	*cone/stone, trunk, car*	KITTI_3	*person, truck, parking, building*

class refers to cases where data was captured during acquisition but remains unlabeled. An example could be cases where instances are not included in the pre-defined category set or when it is challenging to define a class at the time of annotation, resulting in their assignment as a void or an ignore label. On the other hand, *unobserved-unknown* class corresponds to cases where it first appears during testing without being captured during data acquisition. We then constructed three scenarios based on the presence of each unknown class type.

Scenario-A: In *scenario-A*, we assess the discovery capability for observed-unknown classes. Following the previous open-set 3D semantic segmentation [8, 35] literature that addresses the observed-unknown classes in testing, we set the labels of unknown classes to be void and ignore them during training. Depending on the choice of the unknown class, we made four splits each for SemanticKITTI and SemanticPOSS, as shown in Tab. 1. The splits are selected considering the diversity of unknown classes and the relationship between classes. We provide details on the selection process in the Supplementary Material.

Scenario-B: In *scenario-B*, we evaluate the capability to discover unobserved-unknown classes that **do not appear** in the training data. Specifically, we select relatively infrequent classes from SemanticKITTI, {*other-vehicle*, *bicyclist*}, as unknown classes to prevent a significant decrease in the amount of training data. Frames containing any points of these classes are then excluded from training data. As a result, 13,501 out of 19,130 frames are used for training. We did not use SemanticPOSS, as all classes appear in most frames. The evaluation is conducted for all classes, including unobserved-unknown classes, using the same validation data as the official benchmark.

Scenario-C: Both observed-unknown and unobserved-unknown classes are considered in *scenario-C*. In SemanticKITTI, we select {*other-vehicle*, *parking*} as observed-unknown classes, and {*bicyclist*} as an unobserved-unknown class. Frames containing any points of the unobserved-unknown class are excluded from training data, resulting in 17,954 frames for training, and observed-unknown classes are ignored during training. The evaluation proceeds as in *scenario-B*.

4.2 Baselines

Since OCDSS necessitates inductive learning with known classes while enabling instant inference during testing, many previous approaches are impractical under our setting. Clustering-based methods do not provide instant inference and demand a substantial amount of memory due to the large number of points in the LiDAR point cloud, making them more impractical. Previous NCD method [45] is also unable to learn under our setting where unknown classes are unavailable. Therefore, following the previous OCD approach [14], we implement MLDG [34] and SMILE [14], representative OCD methods in image classification, in the 3D LiDAR points domain as baselines.

Baseline. The baseline is the hash coding-based model introduced in Sec. 3.2.

MLDG. MLDG is a meta-learning method for domain generalization. Since OCDSS is more like generalizing to unknown classes from known classes, we simulate introducing unknown classes during training. At each training iteration, we randomly divide known classes into two groups, using one group for meta-training and the other for meta-testing, and then applying MLDG to Baseline.

SMILE. SMILE proposes sign-magnitude disentanglement to better capture the class-level semantics. Following [14], we modified the projection head ϕ by replacing the final single MLP layer with two separate MLP layers for sign and magnitude predictions, and subsequently applying SMILE.

Implementation Details. We use the same backbone network and optimization strategy for all baselines. Specifically, MinkUNet34 [12] is used as the feature extractor $\mathcal{E}$. For the projection head ϕ, we use multi-layer perceptrons. We use SGD optimizer [46] with learning rate 0.01, momentum 0.9, and weight decay $1e-4$ with the batch size of 4. We use fixed mixing ratio $r = 0.1$, $\tau_{cos} = 0.07$, $\tau_{hash} = 0.14$, and $L = 12$. More details on baselines and implementation are provided in Supplementary Material.

4.3 Evaluation Metrics

We adopt **Greedy-Hungarian** [15] and **Strict-Hungarian** [52] evaluation protocols by following [14]. During testing, the class descriptor from the hash code directly forms clusters. Then, we match these clusters with ground truth using the Hungarian matching algorithm [32]. Clusters that are not matched are regarded as misclassified. In Greedy-Hungarian, we divide the points into two groups, known and unknown classes, based on the ground truth and then perform Hungarian matching separately within each group. On the other hand, Strict-Hungarian conducts Hungarian matching for all clusters without distinguishing between known and unknown classes. The mean IoU (mIoU) is used for evaluating semantic segmentation performance.

5 Experimental Results

5.1 Comparison

Quantitative Results. We conduct comparative experiments with baselines on all proposed scenarios. We report the results on *scenario-A* in Tab. 2, while the results on *scenario-B* and *scenario-C* are reported in Tab. 3. As shown in Tabs. 2

Table 2: On-the-fly category discovery results on SemanticPOSS (left) and SemanticKITTI (right) within ***Scenario-A***. Best in **bold**.

		Strict-Hungarian (%)			Greedy-Hungarian (%)		
Split	Method	Unknown	Known	All	Unknown	Known	All
POSS0	Baseline	9.20	28.59	22.63	11.41	32.71	26.16
	MLDG	6.33	34.69	25.96	7.55	40.54	30.39
	SMILE	5.79	40.34	29.71	9.05	56.52	41.91
	Ours	**17.88**	**49.60**	**39.84**	**20.30**	**56.54**	**45.39**
POSS1	Baseline	13.21	23.16	20.87	20.01	30.11	27.78
	MLDG	15.72	28.97	25.91	23.66	35.65	32.89
	SMILE	15.20	32.47	28.48	18.22	46.40	39.89
	Ours	**26.68**	**44.11**	**40.09**	**39.28**	**50.47**	**47.89**
POSS2	Baseline	21.03	19.82	20.10	28.51	27.15	27.46
	MLDG	10.04	30.17	25.52	19.82	33.40	30.27
	SMILE	9.00	31.64	26.42	25.27	38.24	35.24
	Ours	**24.52**	**41.40**	**37.51**	**39.59**	**46.71**	**45.07**
POSS3	Baseline	4.47	28.92	23.28	18.12	28.63	26.21
	MLDG	2.24	33.12	26.00	15.43	33.96	29.68
	SMILE	1.89	36.60	28.59	24.04	37.38	34.30
	Ours	**17.09**	**51.39**	**43.48**	**49.19**	**52.54**	**51.77**
Avg	Baseline	11.98	25.12	21.72	19.51	29.65	26.90
	MLDG	8.58	31.74	25.85	16.62	35.89	30.81
	SMILE	7.97	35.26	28.30	19.15	44.64	37.84
	Ours	**21.54**	**46.63**	**40.23**	**37.09**	**51.57**	**47.53**

		Strict-Hungarian (%)			Greedy-Hungarian (%)		
Split	Method	Unknown	Known	All	Unknown	Known	All
KITTI0	Baseline	10.47	25.34	21.43	13.22	30.28	25.79
	MLDG	12.65	23.17	20.40	20.59	27.34	25.56
	SMILE	9.94	28.09	23.31	21.66	38.96	34.41
	Ours	**22.55**	**45.91**	**39.77**	**32.93**	**55.12**	**49.28**
KITTI1	Baseline	13.51	23.72	21.04	20.46	25.13	23.90
	MLDG	8.99	28.87	23.64	12.71	29.76	25.27
	SMILE	6.59	36.61	28.71	22.63	**43.32**	37.88
	Ours	**22.23**	**38.70**	**34.36**	**36.26**	42.64	**40.96**
KITTI2	Baseline	10.03	26.75	22.35	12.54	28.45	24.27
	MLDG	9.50	30.15	24.72	13.58	31.33	26.66
	SMILE	11.12	31.09	25.84	**25.75**	39.79	36.10
	Ours	**14.98**	**38.39**	**32.23**	23.39	**42.90**	**37.77**
KITTI3	Baseline	12.93	26.12	23.34	21.69	27.27	26.10
	MLDG	7.80	25.44	21.72	16.00	27.51	25.09
	SMILE	**19.21**	30.97	28.49	**32.73**	36.60	35.79
	Ours	17.74	**45.15**	**39.38**	27.94	**47.89**	**43.69**
Avg	Baseline	11.74	25.48	22.04	16.98	27.78	25.02
	MLDG	9.74	26.91	22.62	15.72	28.99	25.65
	SMILE	11.72	31.69	26.59	25.69	39.67	36.05
	Ours	**19.38**	**42.04**	**36.44**	**30.13**	**47.14**	**42.93**

Table 3: On-the-fly category discovery results on SemanticKITTI within ***Scenario-B*** (left) and ***Scenario-C*** (right). Best in **bold**.

	Strict-Hungarian (%)			Greedy-Hungarian (%)		
Method	Unknown	Known	All	Unknown	Known	All
Baseline	14.63	24.89	23.81	33.64	25.98	26.79
MLDG	**25.80**	25.68	25.70	40.75	25.49	27.10
SMILE	13.10	31.61	29.67	**47.67**	32.55	34.14
Ours	21.73	**47.83**	**45.08**	42.12	**48.96**	**48.24**

	Strict-Hungarian (%)			Greedy-Hungarian (%)		
Method	Unknown	Known	All	Unknown	Known	All
Baseline	11.77	30.61	27.63	19.14	31.07	29.19
MLDG	11.80	33.77	30.30	29.47	34.70	33.87
SMILE	11.08	41.74	36.90	**51.86**	43.89	45.15
Ours	**16.66**	**47.29**	**42.45**	37.19	**47.76**	**46.09**

and 3, in all scenarios, the baseline model exhibits diminished recognition ability for known classes due to its sensitive representation in hash coding, while the MLDG and SMILE demonstrate improved recognition capabilities compared to the baseline. Compared to all baselines, the proposed method achieves superior performance for known classes, demonstrating the effectiveness of the proposed representation learning for OCDSS, including DPL and MCL. Furthermore, the proposed method's comparative performance in the known classes compared to the oracle (All: 46.16% in SemanticPOSS, All: 49.23% in SemanticKITTI) indicates its superior recognition ability. MLDG and SMILE exhibit lower results for unknown classes in Strict-Hungarian except for MLDG in *scenario-B*. This implies that these models tend to become overly focused on the semantics of known classes, making them particularly confused when faced with unknown classes. MLDG performing well for unknown classes in *scenario-B* is speculated to be the simulation of unknown classes during training, which enhances gen-

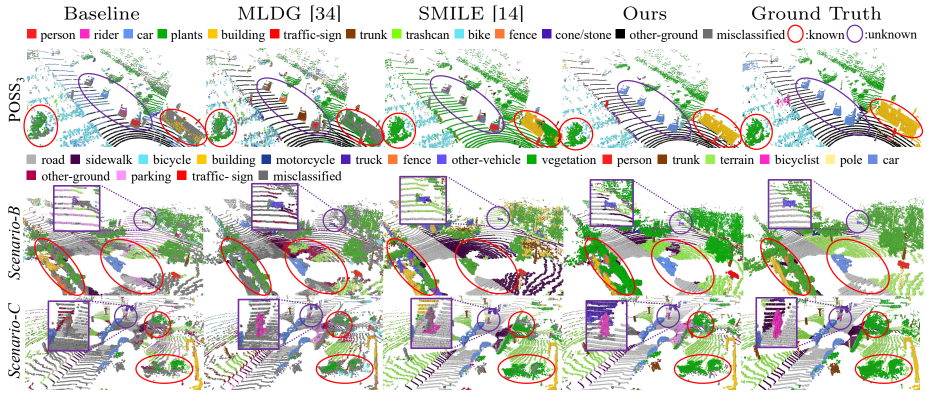


Fig. 3: Qualitative comparison results on POSS_3 within *scenario-A* (top), *scenario-B* (middle), and *scenario-C* (bottom). Best viewed when zoomed in with colors.

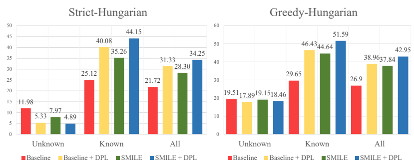
eralization to unobserved-unknown classes. However, this approach offers only limited performance improvements for known classes. In the case of SMILE, its favorable performance on unknown classes in the Greedy-Hungarian suggests that it understands the semantics of unknown classes well. Yet, the poor results in the Strict-Hungarian imply that SMILE frequently confuses known and unknown classes. Overall, the proposed method outperforms baselines in both known and unknown classes in *scenario-A* and *scenario-C*. In *scenario-B* as well, our proposed method consistently outperforms in all classes average, further validating the efficacy of our approach.

Qualitative Results. In Fig. 3, we visualize the qualitative comparison results on *scenario-A* (top), *scenario-B* (middle), and *scenario-C* (bottom). While MLDG and SMILE fail to predict the known and unknown classes correctly, the proposed method not only successfully recognizes the known classes but also cognizes the unknown classes (top: *plant*/*car*, middle: *terrain*/*other-vehicle*, bottom: *vegetation*/*bicyclist*). This implies that the proposed method learns the better representations for OCDSS through the DPL and MCL. Additional qualitative results are provided in Supplementary Material.

5.2 Ablation Studies

To demonstrate the effectiveness of our proposed method, we conduct an ablation study as depicted in Fig. 4. Here, we set the baseline as the model introduced in Sec. 3.2. The experiments are conducted on the SemanticPOSS within *scenario-A*, and the results are averaged over four splits. Please refer to the Supplementary Material for the discussion with class-wise results and analysis on the hash code dimension L, mixing ratio r, weight λ, and sub-sampling in GM.

DPL helps to recognize known classes. Comparing the baseline and baseline+DPL, we observe a significant improvement in performance for the known classes. When the model is trained only with the baseline, hash codes from the



(a)

Method	Strict-Hungarian (%) Unknown	Known	All	Greedy-Hungarian (%) Unknown	Known	All
(a) Base	11.98	25.12	21.72	19.51	29.65	26.90
(b) Base+DPL	5.33	40.08	31.33	17.89	46.43	38.96
(c) Base+DPL+DM	16.48	**48.24**	40.20	31.39	**54.83**	**48.58**
(d) Base+DPL+DM+GM	**21.54**	46.63	**40.36**	**37.09**	51.57	47.53

(b)

Fig. 4: (a) Results of using dual prototypical learning (DPL) with the baseline and SMILE. (b) Ablation study of the proposed method. Base refers to the baseline model.

learned representations are sensitive to intra-class variation, which is supported by the poor performance for known classes. Conversely, adding hash prototypes through DPL imposes additional constraints on the representation, making it more suitable for hash code-based clustering. To further analyze the effects of DPL, we conduct an experiment by integrating the DPL with the baseline and SMILE [14]. As shown in Fig. 4a, when DPL is used together, the performance for known classes of both baseline and SMILE is significantly improved in both Strict- and Greedy-Hungarian. This implies that the proposed DPL effectively improves the robustness of learned representation under intra-class variation. The proposed DPL benefits both baseline and SMILE, demonstrating its effectiveness in a hash-code-based framework for addressing the OCDSS problem.

MCL helps to cognize unknown classes. To excavate the ability to discover unknown classes, we proposed mixing-based category learning based on DPL. With DM, the prototypes become discriminative for ID and OOD. This enhancement leads to improved performance not only for known classes but also for unknown classes. The proposed GM further boosts the performance of unknown classes. Comparing the results of using (c) DM and (d) DM+GM in Fig. 4b, we observe approximately 5% and 5.7% gains in Strict- and Greedy-Hungarian, respectively. Although performance on known classes experiences a slight reduction, considering the inherent challenge of distinguishing unknown classes that the model has not seen during training, this underscores the necessity of GM.

6 Conclusion

In this paper, we introduce a new problem named on-the-fly category discovery for LiDAR semantic segmentation. Along with the baseline model based on hash coding, we propose dual prototypical learning to enhance the model's recognition ability. We further design the mixing-based category learning to inductively transfer recognition knowledge from known classes to unknown classes. Extensive experiments demonstrate the effectiveness of the proposed method for recognizing known classes and discovering unknown classes. While the proposed method effectively addresses the problem, performance disparities between the known and unknown classes remain. In this respect, exploring suitable representation learning for discovering unknown classes presents a promising research direction.

Acknowledgments

This work was supported by the Technology Innovation Program (1415187329, 20024355, Development of autonomous driving connectivity technology based on sensor-infrastructure cooperation) funded by the Ministry of Trade, Industry & Energy(MOTIE, Korea) and the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (NRF2022R1A2B5B03002636).

References

1. Alonso, I., Riazuelo, L., Montesano, L., Murillo, A.C.: 3d-mininet: Learning a 2d representation from point clouds for fast and efficient 3d lidar semantic segmentation. IEEE Robotics and Automation Letters **5**(4), 5432–5439 (2020)
2. Ando, A., Gidaris, S., Bursuc, A., Puy, G., Boulch, A., Marlet, R.: Rangevit: Towards vision transformers for 3d semantic segmentation in autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5240–5250 (2023)
3. Basart, S., Mantas, M., Mohammadreza, M., Jacob, S., Dawn, S.: Scaling out-of-distribution detection for real-world settings. In: International Conference on Machine Learning (2022)
4. Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C., Gall, J.: Semantickitti: A dataset for semantic scene understanding of lidar sequences. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9297–9307 (2019)
5. Bendale, A., Boult, T.E.: Towards open set deep networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1563–1572 (2016)
6. Cao, K., Brbic, M., Leskovec, J.: Open-world semi-supervised learning. In: International Conference on Learning Representations (ICLR) (2022)
7. Cen, J., Yun, P., Cai, J., Wang, M.Y., Liu, M.: Deep metric learning for open world semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15333–15342 (2021)
8. Cen, J., Yun, P., Zhang, S., Cai, J., Luan, D., Tang, M., Liu, M., Yu Wang, M.: Open-world semantic segmentation for lidar point clouds. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022. pp. 318–334 (2022)
9. Chen, G., Peng, P., Wang, X., Tian, Y.: Adversarial reciprocal points learning for open set recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(11), 8065–8081 (2021)
10. Chen, K., Oldja, R., Smolyanskiy, N., Birchfield, S., Popov, A., Wehr, D., Eden, I., Pehserl, J.: Mvlidarnet: Real-time multi-class scene understanding for autonomous driving using multiple views. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 2288–2294. IEEE (2020)
11. Cheng, R., Razani, R., Taghavi, E., Li, E., Liu, B.: 2-s3net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12547–12556 (2021)
12. Choy, C., Gwak, J., Savarese, S.: 4d spatio-temporal convnets: Minkowski convolutional neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3075–3084 (2019)

13. Cortinhal, T., Tzelepis, G., Erdal Aksoy, E.: Salsanext: Fast, uncertainty-aware semantic segmentation of lidar point clouds. In: International Symposium on Visual Computing. pp. 207–222. Springer (2020)
14. Du, R., Chang, D., Liang, K., Hospedales, T., Song, Y.Z., Ma, Z.: On-the-fly category discovery. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11691–11700 (2023)
15. Fini, E., Sangineto, E., Lathuiliere, S., Zhong, Z., Nabi, M., Ricci, E.: A unified objective for novel class discovery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9284–9292 (2021)
16. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: international conference on machine learning. pp. 1050–1059. PMLR (2016)
17. Gasperini, S., Marcos-Ramiro, A., Schmidt, M., Navab, N., Busam, B., Tombari, F.: Segmenting known objects and unseen unknowns without prior knowledge. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 19321–19332 (2023)
18. Ge, Z., Demyanov, S., Chen, Z., Garnavi, R.: Generative openmax for multi-class open set classification. In: British Machine Vision Conference. BMVA Press (2017)
19. Gerdzhev, M., Razani, R., Taghavi, E., Bingbing, L.: Tornado-net: multiview total variation semantic segmentation with diamond inception module. In: 2021 IEEE International Conference on Robotics and Automation (ICRA). pp. 9543–9549. IEEE (2021)
20. Graham, B., Engelcke, M., Van Der Maaten, L.: 3d semantic segmentation with submanifold sparse convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 9224–9232 (2018)
21. Gu, P., Zhang, C., Xu, R., He, X.: Class-relation knowledge distillation for novel class discovery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16474–16483 (2023)
22. Guo, Y., Camporese, G., Yang, W., Sperduti, A., Ballan, L.: Conditional variational capsule network for open set recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 103–111 (2021)
23. Han, K., Rebuffi, S.A., Ehrhardt, S., Vedaldi, A., Zisserman, A.: Automatically discovering and learning new visual categories with ranking statistics. In: International Conference on Learning Representations (2019)
24. Han, K., Vedaldi, A., Zisserman, A.: Learning to discover novel visual categories via deep transfer clustering. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8401–8409 (2019)
25. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. In: International Conference on Learning Representations (2017), https://openreview.net/forum?id=Hkg4TI9xl
26. Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A.: Randla-net: Efficient semantic segmentation of large-scale point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11108–11117 (2020)
27. Hwang, J., Oh, S.W., Lee, J.Y., Han, B.: Exemplar-based open-set panoptic segmentation network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1175–1184 (2021)
28. Jia, X., Han, K., Zhu, Y., Green, B.: Joint representation learning and novel category discovery on single-and multi-modal data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 610–619 (2021)

29. Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., Krishnan, D.: Supervised contrastive learning. Advances in neural information processing systems **33**, 18661–18673 (2020)
30. Kong, L., Liu, Y., Chen, R., Ma, Y., Zhu, X., Li, Y., Hou, Y., Qiao, Y., Liu, Z.: Rethinking range view representation for lidar segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 228–240 (2023)
31. Kong, S., Ramanan, D.: Opengan: Open-set recognition via open data generation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 813–822 (2021)
32. Kuhn, H.W.: The hungarian method for the assignment problem. Naval research logistics quarterly **2**(1-2), 83–97 (1955)
33. Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive uncertainty estimation using deep ensembles. Advances in neural information processing systems **30** (2017)
34. Li, D., Yang, Y., Song, Y.Z., Hospedales, T.: Learning to generalize: Meta-learning for domain generalization. In: Proceedings of the AAAI conference on artificial intelligence. vol. 32 (2018)
35. Li, J., Dong, Q.: Open-set semantic segmentation for point clouds via adversarial prototype framework. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9425–9434 (2023)
36. Li, Y., Bu, R., Sun, M., Wu, W., Di, X., Chen, B.: Pointcnn: Convolution on x-transformed points. Advances in neural information processing systems **31** (2018)
37. Lis, K., Nakka, K., Fua, P., Salzmann, M.: Detecting the unexpected via image resynthesis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2152–2161 (2019)
38. Lu, T., Ding, X., Liu, H., Wu, G., Wang, L.: Link: Linear kernel for lidar-based 3d perception. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1105–1115 (2023)
39. Milioto, A., Vizzo, I., Behley, J., Stachniss, C.: Rangenet++: Fast and accurate lidar semantic segmentation. In: 2019 IEEE/RSJ international conference on intelligent robots and systems (IROS). pp. 4213–4220. IEEE (2019)
40. Neal, L., Olson, M., Fern, X., Wong, W.K., Li, F.: Open set learning with counterfactual images. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 613–628 (2018)
41. Oza, P., Patel, V.M.: C2ae: Class conditioned auto-encoder for open-set recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2307–2316 (2019)
42. Pan, Y., Gao, B., Mei, J., Geng, S., Li, C., Zhao, H.: Semanticposs: A point cloud dataset with large quantity of dynamic instances. In: 2020 IEEE Intelligent Vehicles Symposium (IV). pp. 687–693. IEEE (2020)
43. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 652–660 (2017)
44. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems **30** (2017)
45. Riz, L., Saltori, C., Ricci, E., Poiesi, F.: Novel class discovery for 3d point cloud semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9393–9402 (2023)
46. Ruder, S.: An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747 (2016)

47. Scheirer, W.J., de Rezende Rocha, A., Sapkota, A., Boult, T.E.: Toward open set recognition. IEEE transactions on pattern analysis and machine intelligence **35**(7), 1757–1772 (2012)
48. Shu, Y., Shi, Y., Wang, Y., Huang, T., Tian, Y.: P-odn: Prototype-based open deep network for open set recognition. Scientific reports **10**(1), 7146 (2020)
49. Sun, X., Yang, Z., Zhang, C., Ling, K.V., Peng, G.: Conditional gaussian distribution learning for open set recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13480–13489 (2020)
50. Tang, H., Liu, Z., Zhao, S., Lin, Y., Lin, J., Wang, H., Han, S.: Searching efficient 3d architectures with sparse point-voxel convolution. In: European conference on computer vision. pp. 685–702. Springer (2020)
51. Thomas, H., Qi, C.R., Deschaud, J.E., Marcotegui, B., Goulette, F., Guibas, L.J.: Kpconv: Flexible and deformable convolution for point clouds. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6411–6420 (2019)
52. Vaze, S., Han, K., Vedaldi, A., Zisserman, A.: Generalized category discovery. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7492–7501 (2022)
53. Verma, V., Lamb, A., Beckham, C., Najafi, A., Mitliagkas, I., Lopez-Paz, D., Bengio, Y.: Manifold mixup: Better representations by interpolating hidden states. In: International conference on machine learning. pp. 6438–6447. PMLR (2019)
54. Wang, Y., Li, B., Che, T., Zhou, K., Liu, Z., Li, D.: Energy-based open-world uncertainty modeling for confidence calibration. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9302–9311 (2021)
55. Wu, B., Wan, A., Yue, X., Keutzer, K.: Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3d lidar point cloud. In: 2018 IEEE International Conference on Robotics and Automation (ICRA). pp. 1887–1893. IEEE (2018)
56. Wu, B., Zhou, X., Zhao, S., Yue, X., Keutzer, K.: Squeezesegv2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a lidar point cloud. In: 2019 International Conference on Robotics and Automation (ICRA). pp. 4376–4382. IEEE (2019)
57. Xia, Y., Zhang, Y., Liu, F., Shen, W., Yuille, A.L.: Synthesize then compare: Detecting failures and anomalies for semantic segmentation. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16. pp. 145–161. Springer (2020)
58. Yoshihashi, R., Shao, W., Kawakami, R., You, S., Iida, M., Naemura, T.: Classification-reconstruction learning for open-set recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4016–4025 (2019)
59. Zhang, Y., Zhou, Z., David, P., Yue, X., Xi, Z., Gong, B., Foroosh, H.: Polarnet: An improved grid representation for online lidar point clouds semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9601–9610 (2020)
60. Zhao, B., Han, K.: Novel visual category discovery with dual ranking statistics and mutual knowledge distillation. Advances in Neural Information Processing Systems **34**, 22982–22994 (2021)
61. Zhao, Y., Zhong, Z., Sebe, N., Lee, G.H.: Novel class discovery in semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4340–4349 (2022)

62. Zhong, Z., Fini, E., Roy, S., Luo, Z., Ricci, E., Sebe, N.: Neighborhood contrastive learning for novel class discovery. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10867–10875 (2021)
63. Zhong, Z., Zhu, L., Luo, Z., Li, S., Yang, Y., Sebe, N.: Openmix: Reviving known knowledge for discovering novel visual categories in an open world. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9462–9470 (2021)
64. Zhou, D.W., Ye, H.J., Zhan, D.C.: Learning placeholders for open-set recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4401–4410 (2021)
65. Zhou, H., Zhu, X., Song, X., Ma, Y., Wang, Z., Li, H., Lin, D.: Cylinder3d: An effective 3d framework for driving-scene lidar semantic segmentation. arXiv preprint arXiv:2008.01550 (2020)
66. Zhou, Y., Tuzel, O.: Voxelnet: End-to-end learning for point cloud based 3d object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4490–4499 (2018)