Deep Metric Learning for Open World Semantic Segmentation

Jun Cen   Peng Yun   Junhao Cai   Michael Yu Wang   Ming Liu
The Hong Kong University of Science and Technology
{jcenaa, pyun, jcaiaq}@connect.ust.hk, {mywang, eelium}@ust.hk

Abstract

*Classical close-set semantic segmentation networks have limited ability to detect out-of-distribution (OOD) objects, which is important for safety-critical applications such as autonomous driving. Incrementally learning these OOD objects with few annotations is an ideal way to enlarge the knowledge base of the deep learning models. In this paper, we propose an **open world semantic segmentation system** that includes two modules: (1) an open-set semantic segmentation module to detect both in-distribution and OOD objects. (2) an incremental few-shot learning module to gradually incorporate those OOD objects into its existing knowledge base. This open world semantic segmentation system behaves like a human being, which is able to identify OOD objects and gradually learn them with corresponding supervision. We adopt the Deep Metric Learning Network (DMLNet) with contrastive clustering to implement open-set semantic segmentation. Compared to other open-set semantic segmentation methods, our DMLNet achieves state-of-the-art performance on three challenging open-set semantic segmentation datasets without using additional data or generative models. On this basis, two incremental few-shot learning methods are further proposed to progressively improve the DMLNet with the annotations of OOD objects.*

1. Introduction

Deep convolutional networks have achieved tremendous success in semantic segmentation tasks [1, 2], benefiting from high-quality datasets [3, 4, 5]. These semantic segmentation networks are used as the perception system in many applications, like autonomous driving [6], medical diagnose [7], etc. However, most of these perception systems are *close-set* and *static*. Close-set semantic segmentation is subject to the assumption that all classes in testing are already involved during training, which is not true in the open world. A close-set system could cause disastrous consequences in security-critical applications, like autonomous driving, if it wrongly assigns an in-distribution label to an OOD object [8]. Meanwhile, a static perception system cannot update its knowledge base according to what it has seen,
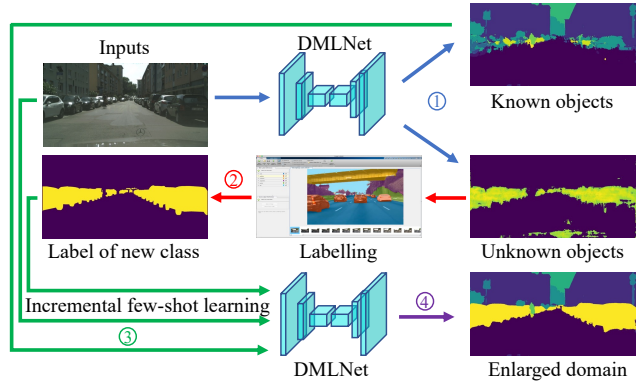

Figure 1. **Open world semantic segmentation system.** Step **1**: Identify both known and unknown objects (blue arrow). Step **2**: Annotate for unknown objects (red arrow). Step **3**: Apply incremental few-shot learning to increase the classification range of the network (green arrow). Step **4**: After incremental few-shot learning, DMLNet can output the result in a larger domain (purple arrow).

and therefore, it is limited to particular scenarios and needs to be retrained after a certain amout of time. To solve these problems, we propose an *open-set* and *dynamic* perception system called the open world semantic segmentation system. It contains two modules: (1) an **open-set semantic segmentation module** to detect OOD objects and assign correct labels to in-distribution objects. (2) an **incremental few-shot learning module** to incorporate those unknown objects progressively into its existing knowledge base. The whole pipeline of our proposed open world semantic segmentation system is shown in Fig. 1.

Both open-set semantic segmentation and incremental few-shot learning are not well solved. For open-set semantic segmentation, the essential part is to identify OOD pixels among all pixels in one image which is called *anomaly segmentation*. The typical approach of anomaly segmentation is to adapt methods of image-level open-set classification for pixel-level open-set classification. These methods include uncertainty estimation-based methods [9, 10, 11, 12] and autoencoder-based methods [13, 14]. However, both of these methods have been proved ineffective in driving sce-

narios as uncertainty estimation-based methods give many false-positive outlier detections [15] and autoencoders cannot regenerate complicated urban scenarios [16]. Recently, generative adversarial network-based (GAN-based) methods [16, 17] have been proved effective but they are far from lightweight as they need several deep networks in the pipeline. For incremental few-shot learning, we have to deal with challenges not only from the incremental learning, such as catastrophic forgetting [18], but also from the few-shot learning, including extracting representative features from a small number of samples [19].

In this paper, we propose to use the DMLNet to solve the open world semantic segmentation problem. The reasons are threefold: (1) The classification principle of the DMLNet is based on contrastive clustering, which is effective to identify anomalous objects, as shown in Fig. 2. (2) The DMLNet combined with prototypes is very suitable for few-shot tasks [19]. (3) Incremental learning of the DML-Net can be implemented by adding new prototypes, which is a natural and useful approach [20]. Based on the DML-Net architecture, we develop two unknown identification criteria for the open-set semantic segmentation module and two methods for the incremental few-shot learning module. Both modules are verified to be effective and lightweight according to our experiments. To summarize, our contributions are the following:

- We are the first to introduce the open world semantic segmentation system, which is more robust and practical in real-world applications.
- Our proposed open-set semantic segmentation module based on the DMLNet achieves state-of-the-art performance on three challenging datasets.
- Our proposed methods for the few-shot incremental learning module alleviate the catastrophic forgetting problem to a large extent.
- An open world semantic segmentation system is fulfilled by combining our proposed open-set semantic segmentation module and incremental few-shot learning module.

2. Related Work

2.1. Anomaly semantic segmentation

The approaches of anomaly semantic segmentation can be divided into two trends: uncertainty estimation-based methods and generative model-based methods. The baseline of uncertainty estimation is the maximum softmax probability (MSP), which was first proposed in [9]. Instead of using the softmax probability, Dan *et al.* [11] proposed to use the maximum logit (MaxLogit) and achieved better anomaly segmentation performance. Bayesian networks adopt a probabilistic view of deep learning networks,
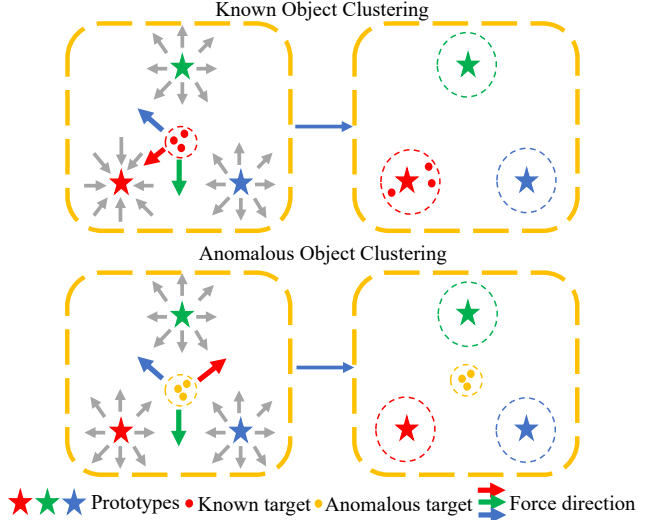

Figure 2. **Contrastive clustering of DMLNet.** During inference, known objects will be attracted by the prototype of the same class and repelled by remaining prototypes. In the end they will aggregate around the particular prototype. In contrast, anomalous objects will be repelled by all prototypes so they will aggregate at the middle of the metric space.

so their weights and outputs are probability distributions instead of specific numbers [21, 22]. In practice, Dropout [10] or ensembles [12] are usually used to approximate Bayesian inference. The autoencoder (AE) [23, 13] and RBM [14] are the typical generative methods assuming that the reconstruction error of OOD images is larger than in-distribution images.

Recently, another kind of generative model-based on GAN resynthesis was proved to achieve state-of-the-art performance based on its reliable high resolution pixel-to-pixel translation results. SynthCP [17] and DUIR [16] are two methods based on GAN resynthesis. Unfortunately, they are far away from lightweight as two or three neural networks have to be used in sequence for OOD detection. Compared to them, we demonstrate that the DMLNet based on contrastive clustering has better anomaly segmentation performance, while only needing to inference for one time.

2.2. Deep metric learning networks

DMLNets have been used in many kinds of applications, including video understanding [24] and person re-identification [25]. The DMLNet translates such problems to calculate embedded feature similarity in metric space using Euclidean, Mahalanobis, or Matusita distances [26]. Convolutional prototype networks and DMLNets are usually used together to tackle specific problems, such as detecting image-level OOD samples [27, 28, 29] and few-shot learning for semantic segmentation [19, 30, 31]. We also follow this combination to build the first DMLNet for open world semantic segmentation.

2.3. Open world classification and detection

Open world classification was first proposed by [32]. This work presented the Nearest Non-Outlier (NNO) algorithm which is efficient in incrementally adding object categories, detecting outliers, and managing open space risk. Recently Joseph et al. [33] proposed an open world object detection system based on contrastive clustering, an unknown-aware proposal network, and energy-based unknown identification criteria. The pipeline of our open world semantic segmentation system is similar to theirs except for two important differences that make our task more challenging: (1) In their open-set detection module, they rely on the fact that the Region Proposal Network (RPN) is class agnostic so that potential OOD objects which are not labeled can also be detected. In this way, the information of the OOD samples is valid for training. However, we focus on semantic segmentation where every pixel used in training is assigned an in-distribution label, so no OOD samples can be added into the training. (2) In the incremental learning module, they use all labeled data of the novel class, while we focus on the few-shot condition which is naturally more difficult. Little research focuses on the incremental few-shot learning, which includes incremental few-shot learning for classification [34], object detection [35] and semantic segmentation [36].

3. Open world semantic segmentation

In this section, we give the working pipeline of the open world semantic segmentation system. This system is composed of an open-set semantic segmentation module and an incremental few-shot learning module. Suppose $\mathcal{C}_{in} = \{C_{in,1}, C_{in,2}, ..., C_{in,N}\}$ are N in-distribution classes, which are all annotated in training datasets, and $\mathcal{C}_{out} = \{C_{out,1}, C_{out,2}, ..., C_{out,M}\}$ are M OOD classes not encountered in the training datasets.

The **open-set semantic segmentation module** is further divided into two submodules: a close-set semantic segmentation submodule and an anomaly segmentation submodule. We assume that $\hat{\mathbf{Y}}^{close}$ is the output map of the close-set semantic segmentation submodule, so the class of each pixel $\hat{\mathbf{Y}}_{i,j}^{close} \in \mathcal{C}_{in}$. The function of the anomaly segmentation submodule is to identify OOD pixels, and its output is called the anomalous probability map: $\hat{\mathbf{P}} \in [0,1]^{H \times W}$, where H and W denote the height and width of the input image. Based on $\hat{\mathbf{Y}}^{close}$ and $\hat{\mathbf{P}}$, the open-set semantic segmentation map $\hat{\mathbf{Y}}^{open}$ is given as:

$$\hat{\mathbf{Y}}_{i,j}^{open} = \begin{cases} C_{anomaly} & \hat{\mathbf{P}}_{i,j} > \lambda_{out} \\ \hat{\mathbf{Y}}_{i,j}^{close} & \hat{\mathbf{P}}_{i,j} \leqslant \lambda_{out} \end{cases} \quad (1)$$

where $C_{anomaly}$ denotes the OOD class, and λ_{out} is the threshold to determine OOD pixels. Therefore, the open-set semantic segmentation module is supposed to identify OOD pixels as well as assign correct in-distribution labels.

Then $\hat{\mathbf{Y}}^{open}$ can be forwarded to labelers who can identify $C_{anomaly}$ from $\mathcal{C}_{out}$ and give corresponding annotations of the novel class.

The **incremental few-shot learning module** is used to update the knowledge base of the close-set segmentation submodule from $\mathcal{C}_{in}$ to $\mathcal{C}_{in+M}$ one by one when new labels are available, where $\mathcal{C}_{in+t} = \mathcal{C}_{in} \cup \{C_{out,1}, C_{out,2}, ..., C_{out,t}\}, t \in \{1, 2, ..., M\}$. Fig. 1 shows the circular working pipeline of the open world semantic segmentation system.

4. Approach

We adopt the DMLNet as our feature extractor and discuss the architecture and loss function in Section 4.1. The open-set segmentation module and the incremental few-shot learning module are illustrated in Section 4.2 and 4.3.

4.1. Deep Metric Learning Network

Classical CNN-based semantic segmentation networks can be disentangled into two parts: a feature extractor $f(\mathbf{X}; \theta_f)$ for obtaining the embedding vector of each pixel and a classifier $g(f(\mathbf{X}; \theta_f); \theta_g)$ for generating the decision boundary, where $\mathbf{X}$, θ_f and θ_g denote the input image, parameters of the feature extractor and classifier respectively. This learnable classifier is not suitable for OOD detection because it assigns all feature space to known classes and leaves no space for OOD classes. In contrast, the classifier is replaced by the Euclidean distance representation with all prototypes $\mathcal{M}_{in} = \{\mathbf{m}_t \in \mathbb{R}^{1 \times N} | t \in \{1, 2, ..., N\}\}$ in DMLNet, where $\mathbf{m}_t$ refers to the prototype of class $C_{in,t}$. The feature extractor $f(\mathbf{X}; \theta_f)$ learns to map the input $\mathbf{X}$ to the feature vector which has the same length as the prototype in metric space. For the close-set segmentation task, the probability of one pixel $\mathbf{X}_{i,j}$ belonging to the class $C_{in,t}$ is formulated as:

$$p_t(\mathbf{X}_{i,j}) = \frac{exp(-\|f(\mathbf{X}; \theta_f)_{i,j} - \mathbf{m}_t\|^2)}{\sum_{t'=1}^{N} exp(-\|f(\mathbf{X}; \theta_f)_{i,j} - \mathbf{m}_{t'}\|^2)} \quad (2)$$

Based on this Euclidean distance-based probability, the **discriminative cross entropy (DCE)** loss function $\mathcal{L}_{DCE}(\mathbf{X}_{i,j}, \mathbf{Y}_{i,j}; \theta_f, \mathcal{M}_{in})$ [27] is defined as:

$$\mathcal{L}_{DCE} = \sum_{i,j} -log(\frac{exp(-\|f(\mathbf{X}; \theta_f)_{i,j} - \mathbf{m}_{\mathbf{Y}_{i,j}}\|^2)}{\sum_{k=1}^{N} exp(-\|f(\mathbf{X}; \theta_f)_{i,j} - \mathbf{m}_k\|^2)}) \quad (3)$$

where $\mathbf{Y}$ is the label of the input image $\mathbf{X}$. The numerator and denominator of $\mathcal{L}_{DCE}$ refer to the attractive force and repulsive force in Fig. 2 respectively. We formulate another loss function called the **variance loss (VL)** function $\mathcal{L}_{VL}(\mathbf{X}_{i,j}, \mathbf{Y}_{i,j}; \theta_f, \mathcal{M}_{in})$ which is defined as:

$$\mathcal{L}_{VL} = \sum_{i,j} \|f(\mathbf{X}; \theta_f)_{i,j} - \mathbf{m}_{\mathbf{Y}_{i,j}}\|^2 \quad (4)$$
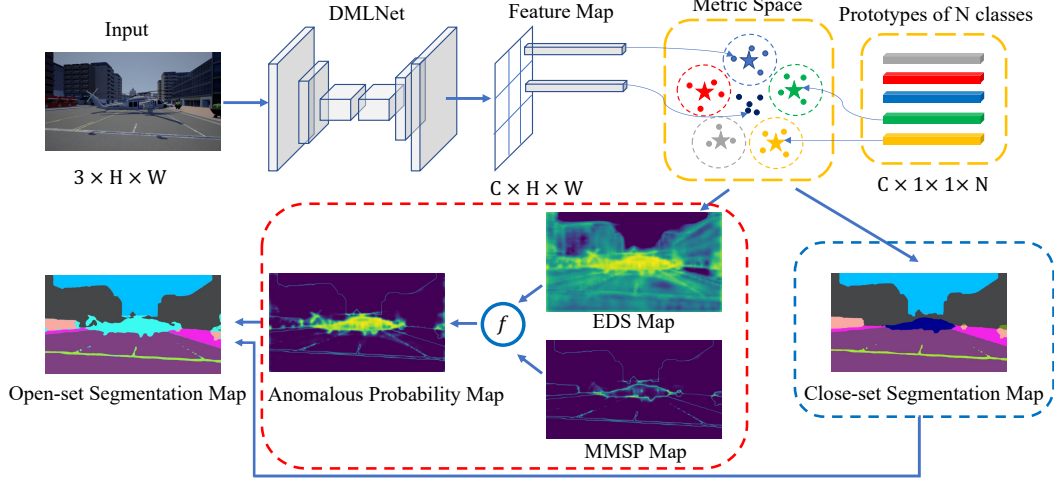
Figure 3. **Open-set semantic segmentation module.** The close-set segmentation submodule is enclosed within the blue dashed box, and the anomaly segmentation submodule is enclosed within the red dashed box. The open-set segmentation map is the combination of results generated by these two submodules. Both the in-distribution classes and OOD classes are predicted in the open-set segmentation map. Kindly refer to Section 4.2 for the definition of the EDS map and MMSP map.
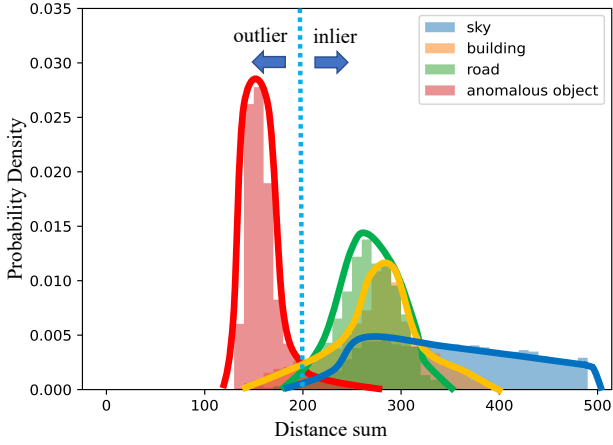

Figure 4. **Distance sum distributions.** The statistic of this figure comes from the EDS map in Fig. 3. Only the classes whose number of pixels is over 4% are demonstrated. This example demonstrates distinct peaks for inlier objects and outlier objects while anomalous classes have a smaller distance sum than known classes.

$\mathcal{L}_{\text{VL}}$ only has an attractive force effect but no repulsive force effect. With DCE and VL, the hybrid loss is defined as: $\mathcal{L} = \mathcal{L}_{\text{DCE}} + \lambda_{\text{VL}} \mathcal{L}_{\text{VL}}$, where λ_{VL} is a weight parameter.

4.2. Open-set semantic segmentation module

The open-set semantic segmentation module is composed of the close-set semantic segmentation submodule and anomaly segmentation submodule. The pipeline of the open-set semantic segmentation module is shown in Fig. 3.

The **close-set semantic segmentation submodule** assigns in-distribution labels to all pixels of one image. As the probability of one pixel $\mathbf{X}_{i,j}$ belonging to class $C_{in,t}$ is

formulated in Equation 2, the close-set segmentation map is:

$$\hat{\mathbf{Y}}_{i,j}^{close} = \underset{t}{\arg\max}\ p_t(\mathbf{X}_{i,j}) \tag{5}$$

The **anomaly segmentation submodule** detects OOD pixels. We propose two unknown identification criteria to measure the anomalous probability including *metric-based maximum softmax probability (MMSP)* and *Euclidean distance sum (EDS)*. Following is the anomalous probability based on the MMSP:

$$\hat{\mathbf{P}}_{i,j}^{\text{MMSP}} = 1 - max\ p_t(\mathbf{X}_{i,j}),\ t \in \{1, 2, ..., N\} \tag{6}$$

The EDS is proposed from the finding that the Euclidean distance sum with all prototypes is smaller if the feature is located at the center of the metric space where OOD pixels aggregate. The EDS is defined as:

$$S(\mathbf{X}_{i,j}) = \sum_{t=1}^{N} \|f(\mathbf{X}; \theta_f)_{i,j} - m_t\|^2 \tag{7}$$

The anomalous probability based on the EDS is calculated as:

$$\hat{\mathbf{P}}_{i,j}^{\text{EDS}} = 1 - \frac{S(\mathbf{X}_{i,j})}{max\ S(\mathbf{X})} \tag{8}$$

The EDS is class-independent, so prototypes of all classes should be evenly distributed in metric space and not moved during training. Learnable prototypes will cause instability during training and make no contribution to better performance [37]. Therefore, we define the prototype in a one-hot vector form: only the tth element of m_t is T, while others remain zero, where $t \in \{1, 2, ..., N\}$.

Considering that the EDS is a ratio relative to the maximum distance sum among all pixels, the high anomalous score area definitely exists in every image even though there are no OOD objects in an image. Additionally, the distance
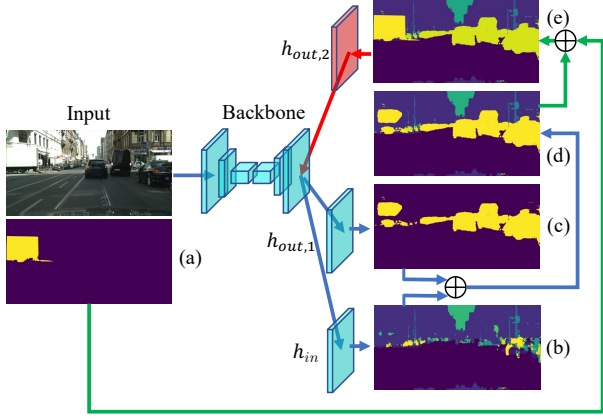
Figure 5. **Pseudo Label Method.** This flowchart shows an example of how to train the new branch head $h_{out,2}$ which is in red. h_{in} and $h_{out,1}$ provide the prediction maps (b) $\mathbf{M}_{in}$ and (c) $\mathbf{M}_{out,1}$ of the original in-distribution classes and the first learned OOD class respectively. Two of them are combined to generate (d) $\mathbf{M}_{in,1}$ which contains all known classes. Then the annotation map (a) $\mathbf{Y}_{out,2}$ of the novel class is merged to obtain the pseudo label map (e) $\mathbf{PL}_{in+1}$ for training. Different colors indicate different classes.

sum distribution of every in-distribution class is slightly different from each other, as shown in Fig. 4. Therefore, we combine the MMSP with the EDS to suppress those pixels with middle response which are actually in-distribution. The mixture function is:

$$\hat{\mathbf{P}} = \alpha \, \hat{\mathbf{P}}^{\text{EDS}} + (1 - \alpha) \, \hat{\mathbf{P}}^{\text{MMSP}} \qquad (9)$$

and α is determined by:

$$\alpha = \frac{1}{1 + exp(-\beta \, (\hat{\mathbf{P}}^{\text{EDS}} - \gamma))} \qquad (10)$$

where β and γ are hyperparameters to control the suppressing effect and threshold.

After obtaining the anomalous probability map by Equation 9 and the close-set segmentation map by Equation 5, we apply Equation 1 to generate the final open-set segmentation map.

4.3. Incremental few-shot learning module

The open-set segmentation results obtained from Section 4.2 can be delivered to labelers who can give corresponding annotations for a certain new class. As the labeling process is extremely time-consuming, especially for segmentation tasks, we make following two assumptions: (1) Only the new class will be annotated while other pixels will be ignored. (2) Less than five images will be annotated.

We propose two methods for the DMLNet to implement incremental few-shot learning. The first method is called the **Pseudo Label Method (PLM)**, as shown in Fig. 5. In this method, DMLNet is divided into two parts: the final branch heads and the backbone. The trained branch heads
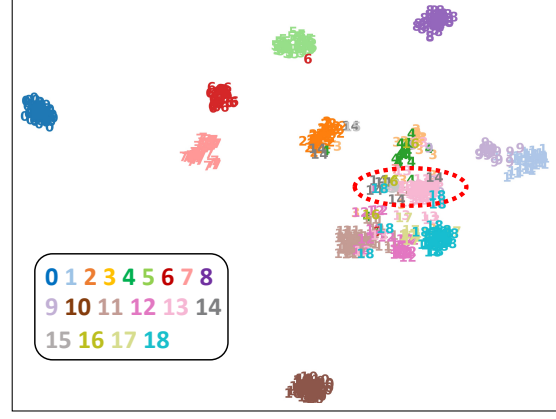

Figure 6. **t-SNE visualization for learned metrics of open-set semantic segmentation module**. All learned metrics of 19 classes of the Cityscapes dataset are included. 13 (car), 14 (truck) and 15 (bus) are OOD classes and are enclosed within the red dotted circle. They are separated from known classes and they also aggregate closely in metric space.

will provide prediction maps of old classes, and labelers will provide the annotation of the new class. In this way, the whole image will be annotated and used to train the new branch head. The second method is called the **Novel Prototype Method (NPM)**. We find the prototype of the novel class directly using the original DMLNet. Based on the novel prototype and in-distribution prototypes, we develop several criteria to classify all classes.

Pseudo Label Method. Suppose the DMLNet has $k + 1$ trained final branch heads $\{h_{in}, h_{out,1}, ..., h_{out,k}\}$, 1 head $h_{out,k+1}$ to be trained and the backbone B. The initial head h_{in} and backbone B are trained using the large training dataset of $\mathcal{C}_{in}$ based on the method in Section 4.1. Therefore, h_{in} provides the prediction map $\mathbf{M}_{in} \in \mathbb{R}^{H \times W}$ for in-distribution classes $\mathcal{C}_{in}$. Suppose $\{\mathbf{M}_{out,1}, ..., \mathbf{M}_{out,k}\}$ are binary maps generated by $\{h_{out,1}, ..., h_{out,k}\}$, and each of them is responsible for predicting only one particular OOD class, *e.g.*, $\mathbf{M}_{out,t}$ is responsible for predicting $C_{out,t}, t \in \{1, 2, ..., k\}$. $\mathbf{Y}_{out,k+1}$ is the binary annotation map of $C_{out,k+1}$ from the labelers. Algorithm 1 is the procedure for generating the pseudo label $\mathbf{PL}_{in+k}$.

Algorithm 1: Pseudo Label Generation Method

Input: $\mathbf{M}_{in}, \{\mathbf{M}_{out,1}, ..., \mathbf{M}_{out,k}\}, \mathbf{Y}_{out,k+1}$
Output: Pseudo Label Map $\mathbf{PL}_{in+k}$

1   $t \leftarrow 1$;
2   **while** $t \leqslant k$ **do**
3     $\mathbf{M}_{in+t} \leftarrow \mathbf{M}_{in+t-1}$;
4     $\mathbf{M}_{in+t}[\mathbf{M}_{out,t} = 1] \leftarrow N + t$;
5     $t \leftarrow t + 1$
6   **end**
7   $\mathbf{PL}_{in+k} \leftarrow \mathbf{M}_{in+k}$;
8   $\mathbf{PL}_{in+k}[\mathbf{Y}_{out,k+1} = 1] \leftarrow N + k + 1$;

$\mathbf{PL}_{in+k}$ is regarded as the ground truth and used to

train the head h_{in+k+1} based on the loss functions in Section 4.1. The dimension of the metric space will also increase one by one when we add the new head. When training the branch head h_{in+k+1}, the output feature of each pixel and all prototypes will have length $N + k + 1$. In other words, the prototype set now becomes $\mathcal{M}_{in+k+1} = \left\{ \mathbf{m}_t \in \mathbb{R}^{1 \times (N+k+1)} | t \in \{1, 2, ..., N + k + 1\} \right\}$. Note that when training the new branch head, the parameters of all the other heads and backbone will be frozen. Fig. 5 gives the pipeline of training head $h_{out,2}$. The inference procedure is the same as how to generate a pseudo label map, as shown in Algorithm 1 except that the last two steps are not needed. In this way, the old knowledge is preserved because the old branch still has a reliable ability to classify base classes.

Novel Prototype Method. This method is inspired by the finding that all feature vectors of certain OOD novel classes will also aggregate together in metric space. Fig. 6 demonstrates that OOD features separate very well with in-distribution features while aggregate closely. Therefore, we determine the prototype of the novel class $C_{out,k}$ as the mean of all features belonging to this novel class:

$$\mathbf{m}_{N+k} = \frac{\sum_{q=1}^{Q} \sum_{i,j} \mathbf{F}_{i,j} \cdot \mathbf{Y}_{i,j}}{\sum_{q=1}^{Q} \sum_{i,j} \mathbf{Y}_{i,j}} \quad (11)$$

where Q is the number of annotated samples, and $\mathbf{Y}$ is the ground truth binary mask of novel class $C_{out,k}$, in which only those locations with novel class are 1, else 0. $\mathbf{F} = f(\mathbf{X}; \theta_f) \in \mathbb{R}^{N \times H \times W}$ is the feature map generated by the DMLNet.

The new prototype set is no longer evenly distributed because original in-distribution prototypes are all one-hot vectors, but the newly added prototype of the novel class is not. So using Equation 2 to classify novel classes is not suitable anymore. We develop these two criteria to decide whether one pixel belongs to $C_{out,k}$:

$$\begin{cases} \|\mathbf{F}_{i,j} - \mathbf{m}_{N+k}\|^2 < \lambda_{novel} \\ \|\mathbf{F}_{i,j} - \mathbf{m}_{N+k}\|^2 < \|\mathbf{F}_{i,j} - \mathbf{m}_{k'}\|^2 \quad k' \neq N + k \end{cases} \quad (12)$$

where λ_{novel} is a threshold we can manually design. The intuition is that the novel pixel should be close enough with its corresponding prototype as well as closest to its corresponding prototype among all prototypes. This method also handles catastrophic forgetting very well because the model is not updated at all. There is no training procedure in this method. Simply calculating the novel prototype and evaluating the input image based on our criteria is the pipeline.

5. Experiments

Our experiments are divided into three parts. We first evaluate our open-set semantic segmentation approach in Section 5.1. Then we demonstrate our incremental few-shot learning results in Section 5.2. Based on the open-set semantic segmentation module and incremental few-show learning module, the whole open world semantic segmentation is realized in Section 5.3.

5.1. Open-set semantic segmentation

Datasets. Three datasets including StreetHazards [11], Lost and Found [38] and Road Anomaly [16] are used to testify the robustness and effectiveness of our DMLNet based open-set semantic segmentation method. Most anomalous objects of StreetHazards are large rare transportation machines such as helicopters, planes, and tractors. The Lost and Found dataset contains lots of small anomalous items like cargo, toys, and boxes. The Road Anomaly dataset no longer constrains the scenario within urban situations but also contains images of villages and mountains.

Metrics. Open-set semantic segmentation is the combination of close-set segmentation and anomaly segmentation as discussed in Section 4.2. For the close-set semantic segmentation task, we use mIoU to evaluate the performance. For the anomaly segmentation task, three metrics are used according to [11], including area under ROC curve (AUROC), false-positive rate at 95% recall (FPR95), and area under the precision-recall curve (AUPR).

Implementation details. For StreetHazards, we follow the same training procedure as [11] to train a PSPNet [2] on the training set of StreetHazards. For Lost and Found and Road Anomaly, we follow [16] to use BDD100k [39] to train a PSPNet. Note that PSPNet is only used to extract features as we discussed in Section 4.1. λ_{VL} of hybrid loss is 0.01. None zero element T of all prototypes is 3. β and γ are 20 and 0.8 respectively in Equation 10.

Baselines. For StreetHazards, several baselines have been evaluated including MSP [9], Dropout [10], AE [13], MaxLogit [11] and SynthCP [17]. For Lost and Found and Road Anomaly, baseline approaches contain MSP, MaxLogit, Ensemble [12], RBM [14] and DUIR [16].

Results. The result of StreetHazards is demonstrated in Table 1. For Lost and Found and Road Anomaly, mIoU is invalid as they only provide OOD class labels but no specific in-distribution class labels. The results are in Table 2. Our experiments show that our DMLNet-based method achieves state-of-the-art performance in all three anomaly segmentation related metrics. Compared to recently proposed GAN-based approaches including DUIR and SynthCP, our method outperforms them in anomaly segmentation quality with a more lightweight structure, as they need two or three deep neural networks in the whole pipeline while we only need to inference for one time. The mIoU value of close-set segmentation in StreetHazards demonstrate that our method has no harm to close-set segmentation. Some qualitative results are shown in Fig. 8.

Ablation studies. We carefully conduct ablation experiments to study the effect of different loss functions (VL and DCE) and anomaly judgment criteria (EDS and MMSP), as shown in Table 3. The fact that DCE has better performance in mIoU than VL indicates the significance of the repulsive force. EDS outperforms MMSP a lot under all loss

Method	AUPR↑	AUROC↑	FPR95↓	mIoU↑
AE	2.2	66.1	91.7	53.2
MSP+CRF	6.5	88.1	29.9	53.2
MSP	6.6	87.7	33.7	53.2
Dropout	7.5	69.9	79.4	-
SynthCP	9.3	88.5	28.4	53.2
MaxLogit	10.6	89.3	26.5	53.2
DML	**14.7**	**93.7**	**17.3**	**53.9**

Table 1. Open-set segmentation results on **StreetHazards**. All baselines except Dropout have the same close-set mIoU because they do not change the network structure nor the inference pipeline for the close-set segmentation submodule.

Dataset	Lost and Found			Road Anomaly		
Method	AUPR↑	AUROC↑	FPR95↓	AUPR↑	AUROC↑	FPR95↓
Ensemble	-	57	-	-	67	-
RBM	-	86	-	-	59	-
MSP	21	83	31	19	70	61
MaxLogit	37	91	21	32	78	49
DUIR	-	93	-	-	83	-
DML	**45**	**97**	**10**	**37**	**84**	**37**

Table 2. Anomaly segmentation results on **Lost and Found** and **Road Anomaly**.

VL	DCE	EDS	MMSP	AUPR↑	AUROC↑	FPR95↓	mIoU↑
✓		✓		**15.1**	91.5	25.1	
✓			✓	6.8	88.6	27.7	49.0
✓		✓	✓	13.7	91.0	25.5	
	✓	✓		13.3	89.7	27.8	
	✓		✓	8.2	91.1	21.3	52.6
	✓	✓	✓	13.3	92.4	20.5	
✓	✓	✓		14.7	**93.7**	**17.3**	
✓	✓		✓	8.8	91.7	19.8	**53.9**
✓	✓	✓	✓	14.1	93.5	18.0	

Table 3. Ablation experiment results on **StreetHazards**. VL and DCE are loss functions when training the DMLNet. EDS and MMSP are unknown identification criteria for anomaly segmentation. Kindly refer to Section 4.1 and Section 4.2 for details.

functions, meaning that the class-agnostic criterion is more suitable in the anomaly segmentation task than the class-dependent criterion.

5.2. Incremental few-shot learning

Two incremental few-shot learning methods PLM and NPM are tested on the Cityscapes dataset. Car, truck, and bus are 3 OOD classes not involved in initial training while the other 16 classes are regarded as in-distribution classes. For PLM, the performance of the latest head PLM_{latest} and all heads PLM_{all} are both evaluated. λ_{novel} is 0.15 to achieve the best performance of NPM. We show the incremental few-shot learning result Tab. 4 of two settings: only adding car and adding car, truck, and bus one by one. The metric $mIoU_{harm}$ is the comprehensive index [40] that balances $mIoU_{old}$ and $mIoU_{novel}$.

We see that PLM_{all} performs better than PLM_{latest} be-
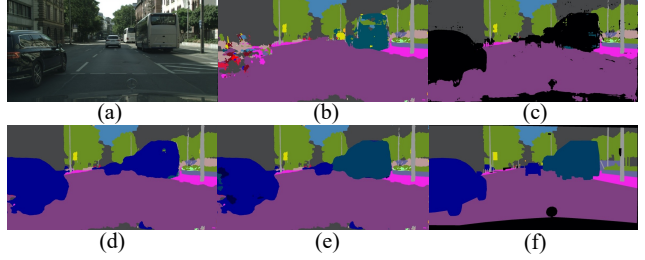

Figure 7. Visualization results of our **open world semantic segmentation system**. (a) is input. (b) is close-set prediction. (c) is open-set prediction. (d) is prediction after 16+1 incremental few-shot learning. (e) is prediction after 16+3 incremental few-shot learning. (f) is ground truth.

cause the latest head only fits classes encountered during incremental few-shot learning, while older heads keep the information of prior knowledge. Even for classes involved in few new training samples, the latest head has worse results than all heads as the latest head cannot obtain robust features from only few samples. In most cases, PLM has the better performance in novel classes but the worse result in old classes than NPM, because the new added heads in PLM that are responsible for predicting novel classes are more likely to classify a pixel as a novel class so that more false-negative detections will be generated for old classes.

5.3. Open world semantic segmentation

We realize an open world semantic segmentation system following these steps: **(1)** Train DMLNet for 16 classes of the Cityscapes dataset based on Section 4.1. Three classes including car, truck, and bus are excluded during training which is the same as Section 5.2. **(2)** Obtain the close-set segmentation result based on Equation 5. A visualization example is shown as (b) in Fig. 7. **(3)** Apply open-set semantic segmentation module based on Section 4.2 to get open-set results like (c) in Fig. 7. The black area denotes anomalous objects. **(4)** Conduct incremental few-shot learning as discussed in Section 4.3 to integrate car, truck, and bus one by one into existing perception knowledge. Results are shown as (d) and (e) in Fig. 7.

Close-set segmentation will assign an in-distribution label to OOD objects as shown in (b) of Fig. 7, which may cause safety-related accidents in autonomous driving. Open-set segmentation gives *unknown* label to OOD objects, which not only makes the perception system more robust, but also provides information for labelers to make annotations of the new class. Incremental few-shot learning can be applied based on these new annotations to enlarge the perception knowledge base. From (d) and (e) of Fig. 7 we can see our DMLNet can gradually classify car and bus. This *open-set segmentation - incremental few-shot learning* cycle can progressively improve the performance of open world semantic segmentation system.

Table 4. Incremental few-shot learning results on Cityscapes for 16 + 1 (car) setting and 16 + 3 (car, truck, bus) setting. The novel classes are in blue and the old classes not involved in new training samples are in red. Finetune (FT) is the baseline with catastrophic forgetting. The upper-bound is to regard the novel class as the original in-distribution class and retrain the model. All results are reported in IoU.
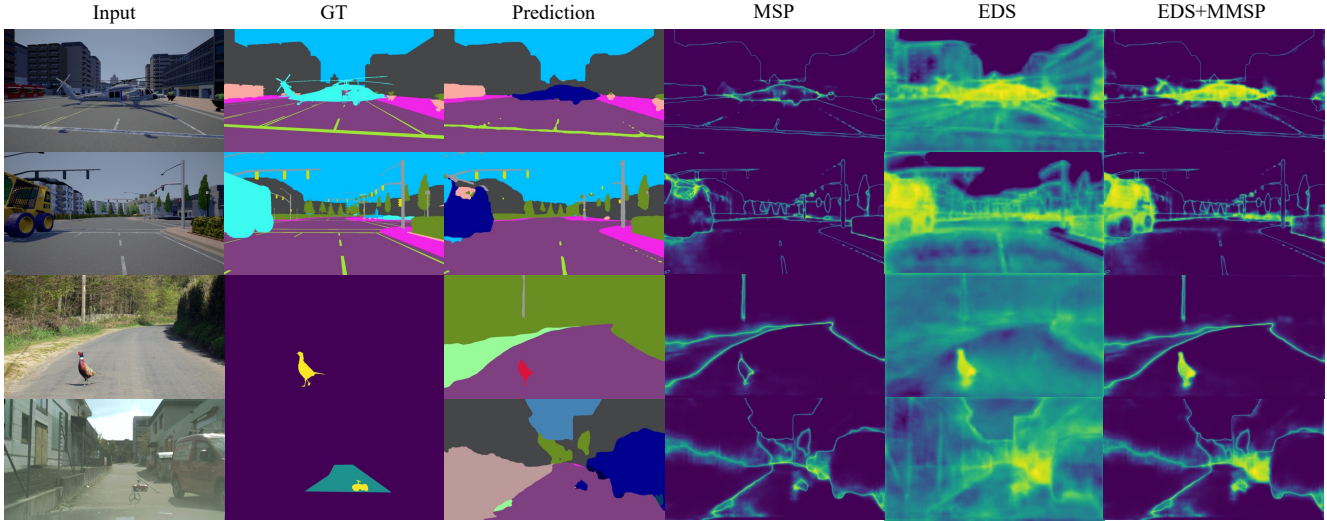
16 + 1 setting	Method	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrain	sky	person	rider	train	motorcycle	bicycle	car	truck	bus	mIoU	mIoU$_{novel}$	mIoU$_{old}$	mIoU$_{harm}$
Baseline	All 17	97.8	82.4	91.8	52.3	57.5	59.9	64.1	74.2	91.9	61.4	94.6	79.4	58.8	75.6	61.7	74.9	94.8	-	-	74.9	-	-	-
	First 16	98.0	82.1	91.4	43.6	56.4	58.9	61.4	72.6	91.6	60.5	94.4	79.1	57.6	67.9	61.1	75.1	-	-	-	72.0	-	-	-
	FT	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6.6	-	-	0.4	6.6	0.0	0.0
5 shot	PLM$_{latest}$	87.6	10.1	80.4	0.7	13.8	23.9	0	39.3	85.4	0.0	85.4	21.3	3.8	2.1	0.0	1.3	75.7	-	-	31.2	**75.7**	28.4	41.3
	PLM$_{all}$	97.1	79.3	89.2	41.9	55.3	57.5	60.8	71.0	91.1	59.4	93.9	73.3	49.2	34.2	14.3	51.8	75.7	-	-	64.4	**75.7**	63.7	**69.2**
	NPM	96.2	79.3	89.2	41.6	52.0	56.3	61.1	69.4	90.4	58.8	94.1	74.4	55.3	53.4	39.2	70.3	64.6	-	-	**67.4**	64.6	**67.6**	66.1
1 shot	PLM$_{latest}$	80.3	0.3	72.6	0.0	0.0	0.0	0.0	17.5	68.8	0.0	61.1	2.2	5.9	0.0	0.0	0.0	64.5	-	-	22.0	**64.5**	19.3	29.7
	PLM$_{all}$	96.8	77.1	89.6	41.4	48.7	53.2	60.3	64.5	90.3	55.6	94.3	59.1	43.6	39.5	12.0	35.7	64.5	-	-	60.4	**64.5**	60.1	62.2
	NPM	95.9	79.2	88.8	41.3	50.5	56.0	61.0	69.1	90.2	58.6	94.1	73.6	55.1	49.7	37.4	69.6	60.1	-	-	**66.5**	60.1	**66.9**	**63.3**
16 + 3 setting																								
Baseline	All 19	97.9	83.0	91.7	51.5	58.3	59.8	64.2	74.2	92.0	61.2	94.6	79.7	59.1	63.9	61.5	75.0	94.2	78.5	81.4	74.8	-	-	-
	First 16	98.0	82.1	91.4	43.6	56.4	58.9	61.4	72.6	91.6	60.5	94.4	79.1	57.6	67.9	61.1	75.1	-	-	-	72.0	-	-	-
	FT	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.4	0.0	0.1	0.0	0.0
5 shot	PLM$_{latest}$	91.6	58.3	78.4	7.4	3.3	34.1	34.0	42.7	85.9	23.4	86.2	5.4	0.0	0.0	0.0	3.1	63.8	1.3	12.0	33.2	25.7	34.6	29.5
	PLM$_{all}$	97.1	79.2	84.8	38.1	46.4	56.8	58.8	61.0	91.0	59.3	92.9	63.6	47.5	3.4	13.8	47.5	67.0	5.7	12.0	54.0	**28.2**	58.8	**38.1**
	NPM	96.1	79.3	58.7	41.5	51.5	56.3	60.7	69.0	90.4	58.8	94.1	74.3	55.1	32.0	39.1	70.2	55.7	1.6	21.0	**58.2**	26.1	**64.2**	37.1
1 shot	PLM$_{latest}$	87.6	34.6	52.4	0.0	0.0	25.7	2.3	25.7	69.8	16.6	0.0	0.0	0.0	0.0	0.0	0.0	41.0	0.7	9.5	19.3	17.1	19.7	18.3
	PLM$_{all}$	96.8	75.2	49.0	33.1	31.4	48.0	33.2	44.6	89.7	55.3	23.0	42.1	32.8	5.3	8.0	27.7	30.4	0.7	9.5	38.7	13.5	43.4	20.6
	NPM	95.8	79.2	44.6	41.2	50.2	56.0	60.5	67.5	90.1	58.6	94.0	73.5	54.9	24.9	37.2	69.6	54.5	1.1	22.0	**56.6**	25.9	**62.3**	**36.5**

Figure 8. Visualization results of **anomaly segmentation**. The first two rows are from **StreetHazards**, while the third and last rows are from **Road Anomaly** and **Lost and Found**. MSP is the baseline while EDS and EDS+MMSP are our methods.

6. Conclusion

We introduce an open world semantic segmentation system which incorporates two modules: an open-set segmentation module and an incremental few-shot learning module. Our proposed open-set segmentation module is based on the deep metric learning network, and it uses the Euclidean distance sum criterion to achieve state-of-the-art performance. Two incremental few-shot learning methods are proposed to broaden the perception knowledge of the network. Both modules of the open world semantic segmentation system can be further studied to improve the performance. We hope our work can draw more researchers to contribute to this practically valuable research direction.

References

[1] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

[2] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2881–2890, 2017.

[3] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016.

[4] Xinyu Huang, Xinjing Cheng, Qichuan Geng, Binbin Cao, Dingfu Zhou, Peng Wang, Yuanqing Lin, and Ruigang Yang. The apolloscape dataset for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 954–960, 2018.

[5] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4990–4999, 2017.

[6] Joel Janai, Fatma Güney, Aseem Behl, Andreas Geiger, et al. Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trends® in Computer Graphics and Vision*, 12(1–3):1–308, 2020.

[7] Robert Challen, Joshua Denny, Martin Pitt, Luke Gompels, Tom Edwards, and Krasimira Tsaneva-Atanasova. Artificial intelligence, bias and clinical safety. *BMJ Quality & Safety*, 28(3):231–237, 2019.

[8] Darko Bozhinoski, Davide Di Ruscio, Ivano Malavolta, Patrizio Pelliccione, and Ivica Crnkovic. Safety for mobile robotic system: A systematic mapping study from a software engineering perspective. *Journal of Systems and Software*, 151:150–179, 2019.

[9] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[10] 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, 2016.

[11] Dan Hendrycks, Steven Basart, Mantas Mazeika, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings. *arXiv preprint arXiv:1911.11132*, 2019.

[12] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6402–6413, 2017.

[13] Christoph Baur, Benedikt Wiestler, Shadi Albarqouni, and Nassir Navab. Deep autoencoding models for unsupervised anomaly segmentation in brain mr images. In *International MICCAI Brainlesion Workshop*, pages 161–169, 2018.

[14] Clement Creusot and Asim Munawar. Real-time small obstacle detection on highways using compressive rbm road reconstruction. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, pages 162–167, 2015.

[15] Petra Bevandić, Ivan Krešo, Marin Oršić, and Siniša Šegvić. Dense outlier detection and open-set recognition based on training with noisy negative images. *arXiv preprint arXiv:2101.09193*, 2021.

[16] Krzysztof Lis, Krishna Nakka, Pascal Fua, and Mathieu Salzmann. Detecting the unexpected via image resynthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2152–2161, 2019.

[17] Yingda Xia, Yi Zhang, Fengze Liu, Wei Shen, and Alan L Yuille. Synthesize then compare: Detecting failures and anomalies for semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 145–161. Springer, 2020.

[18] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989.

[19] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 4077–4087, 2017.

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

[21] David JC MacKay. Bayesian neural networks and density networks. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 354(1):73–80, 1995.

[22] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5574–5584, 2017.

[23] Samet Akcay, Amir Atapour-Abarghouei, and Toby P Breckon. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Asian conference on computer vision*, pages 622–637, 2018.

[24] Joonseok Lee, Sami Abu-El-Haija, Balakrishnan Varadarajan, and Apostol Natsev. Collaborative deep metric learning for video understanding. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 481–490, 2018.

[25] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Deep metric learning for person re-identification. In *International Conference on Pattern Recognition*, pages 34–39. IEEE, 2014.

[26] Kameo Matusita et al. Decision rules, based on the distance, for problems of fit, two samples, and estimation. *The Annals of Mathematical Statistics*, 26(4):631–640, 1955.

[27] Hong-Ming Yang, Xu-Yao Zhang, Fei Yin, and Cheng-Lin Liu. Robust classification with convolutional prototype learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3474–3482, 2018.

[28] Hong-Ming Yang, Xu-Yao Zhang, Fei Yin, Qing Yang, and Cheng-Lin Liu. Convolutional prototype network for open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

[29] Guangyao Chen, Limeng Qiao, Yemin Shi, Peixi Peng, Jia Li, Tiejun Huang, Shiliang Pu, and Yonghong Tian. Learning open set network with discriminative reciprocal points. In *Proceedings of the European conference on computer vision (ECCV)*, pages 507–522, 2020.

[30] Boris N. Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. TADAM: task dependent adaptive metric for improved few-shot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 719–729, 2018.

[31] Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7278–7286, 2018.

[32] A. Bendale and T. Boult. Towards open world recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1893–1902, 2015.

[33] K J Joseph, Salman Khan, Fahad Shahbaz Khan, and Vineeth N Balasubramanian. Towards open world object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[34] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4367–4375, 2018.

[35] Juan-Manuel Perez-Rua, Xiatian Zhu, Timothy M Hospedales, and Tao Xiang. Incremental few-shot object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13846–13855, 2020.

[36] Fabio Cermelli, Massimiliano Mancini, Yongqin Xian, Zeynep Akata, and Barbara Caputo. A few guidelines for incremental few-shot segmentation. *arXiv preprint arXiv:2012.01415*, 2020.

[37] Dimity Miller, Niko Sunderhauf, Michael Milford, and Feras Dayoub. Class anchor clustering: A loss for distance-based open set recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3570–3578, 2021.

[38] Peter Pinggera, Sebastian Ramos, Stefan Gehrig, Uwe Franke, Carsten Rother, and Rudolf Mester. Lost and found: detecting small road hazards for self-driving vehicles. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1099–1106, 2016.

[39] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. BDD100K: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2633–2642, 2020.

[40] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. Semantic projection network for zero- and few-label semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.