Contrastive Class Anchor Learning for Open Set Object Recognition in Driving Scenes

Anonymous authors
Paper under double-blind review

Abstract

Conventional object recognition models operate under closed-set assumptions presuming that the training dataset is sufficiently comprehensive that any object detected during inference can be assigned to some known prior class. This assumption is flawed and potentially dangerous for real-world applications such as driving scene perception where diverse objects and unexpected behaviours should be expected. In order to progress towards trusted autonomous platforms object recognition models need Open Set Recognition (OSR) methods capable of identifying unknown classes while maintaining good performance on known classes. Existing OSR methods are mostly designed for image data and utilize generative models which are hard to train. In this paper, we propose S2CA, a Supervised Contrastive Class Anchor learning method which leverages contrastive learning principles to effectively reject unknown classes by increasing intra-class compactness and inter-class sparsity of known classes in feature space. We train a feature encoder through contrastive learning while ensuring that features of known classes form compact clusters, and then transfer the trained encoder to the OSR task. During inference, the model rejects unknown classes based on class-agnostic information in feature space and class-related information in logit space. The proposed OSR method is simple yet powerful. It is not only suitable for image-based object recognition models, but can also be used for a variety of lidar-based object recognition models. We demonstrate superior performance of S2CA when compared with state of the art methods on two widely used driving scene recognition datasets, i.e., KITTI and nuScenes. The source code is available at `https://github.com/343gltysprk/s2ca`.

1 Introduction

Autonomous driving has received a great deal of attention in recent years due to its potential for eliminating human error, which is the major cause of most road accidents. To ensure safe and efficient navigation, autonomous vehicles must be able to perceive their surroundings using onboard sensors and understand the driving scene. This involves localizing and recognizing surrounding objects.

Conventional object recognition models typically make the assumption of a closed set, requiring that all object categories which the model will encounter are available in the training set (Scheirer et al., 2013). When the recognition model encounters an unknown object, it will misclassify it as one of the known classes. In the real world, autonomous vehicles encounter a wide variety of objects. It is not feasible to include all object categories in the training set. For example, autonomous driving datasets designed for the northern hemisphere do not typically contain kangaroos, yet in Australia kangaroos frequently appear on the road. Therefore, the closed-set assumption will inevitably have an adverse impact on the autonomous driving system.

An ideal driving scene perception system should be able to detect all object instances on the road surface, then reject unknown objects and classify the known ones. Unknown objects can then be stored for later labelling, *e.g.* by a human supervisor. Scheirer et al. (2013) define the problem of rejecting unknown classes ($\mathcal{U}$) and simultaneously classifying known classes ($\mathcal{K}$) as open-set recognition (OSR). Unknown classes are sometimes referred to as "unknown unknowns", because they are not only unseen but also have no side

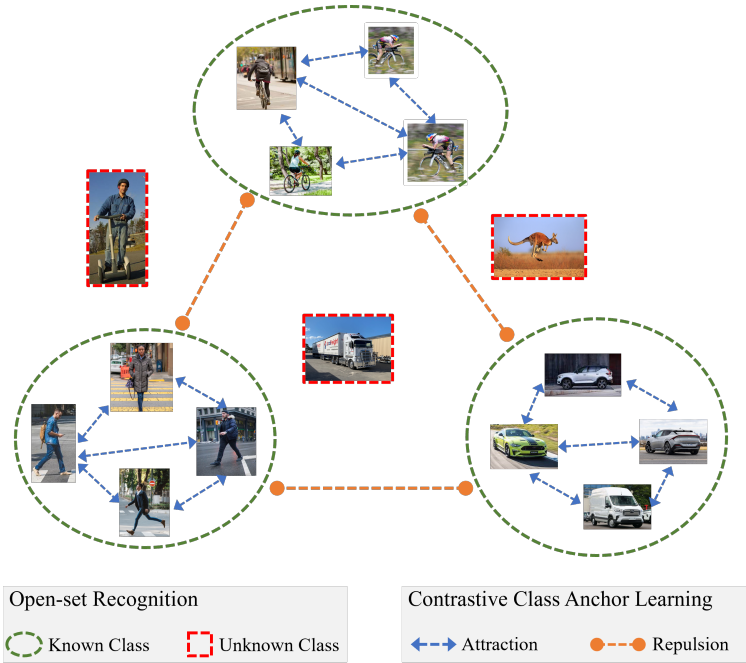


Figure 1. Our proposed approach to open-set recognition aims to effectively recognise unknown objects while correctly classifying known classes by increasing intra-class compactness and inter-class sparsity in the feature space, thereby reducing the chance of mixing known and unknown classes. This is achieved by supervised contrastive learning (SCL) which encourages intra-class attraction and inter-class repulsion. To overcome the limitation of the original SCL, which contrasts the instances only in the same mini-batch, we introduce Contrastive Class Anchor Learning, which applies a global constraint to contrastive learning by placing class anchors in the logit space. The proposed method is efficient and suitable for both image-based and lidar-based object recognition.

information (*e.g.*, semantic attributes or known properties) during training (Oza & Patel, 2019; Geng et al., 2021).

An intuitive OSR method is to set a threshold for the classification confidence score (*e.g.* softmax probability) of the most probable class produced by the model (Hendrycks & Gimpel, 2017). However, previous research has shown that samples in $\mathcal{U}$ can also produce equally high softmax scores (Bendale & Boult, 2015). State-of-the-art OSR methods often involve generative models (Ge et al., 2017; Neal et al., 2018; Oza & Patel, 2019; Perera et al., 2020; Kong & Ramanan, 2021), which help separate known and unknown classes in the feature space. While improved feature learning leads to better OSR performance, the complexity and challenges of training generative models limit their application in open-set recognition in diverse scenarios. Miller et al. (2021) proposed Class Anchor Clustering (CAC), which forces known classes to form compact clusters around class anchors in logit space and rejects unknown objects based on their distance to known clusters. However, the cluster compactness achieved by minimizing the CAC loss can limit the network's ability to learn discriminative features (Miller et al., 2021), whereas OSR is rooted in representation learning (Perera et al., 2020).

To overcome the limitations of existing OSR methods, we propose a framework based on supervised contrastive learning (Khosla et al., 2020) and class anchor clustering (Miller et al., 2021). Our Supervised Contrastive Class Anchor (S2CA) learning framework achieves intra-class compactness and inter-class sparsity in the feature space, thereby reducing the chance of confusing unknown objects with known classes. Fig. 1 illustrates the idea of contrastive class anchor learning. We train a feature encoder through contrastive learning while minimizing the class anchor clustering loss. During inference, the model rejects unknown classes by thresholding the virtual logit (Wang et al., 2022) score. We evaluate the proposed S2CA approach with a wide range of image-based and lidar-based encoder networks, and compare its performance

with state-of-the-art open-set recognition and out-of-distribution (OOD) detection methods on two widely used datasets of KITTI (Geiger et al., 2013) and nuScenes (Caesar et al., 2019).

Our paper makes the following contributions:

- We propose a novel OSR method based on contrastive class anchor learning which enables effective rejection of unknown classes by increasing intra-class compactness and inter-class sparsity of known samples.
- We demonstrate that the proposed method can be applied to different data modalities and encoder networks including image-based and lidar-based networks.
- We present a thorough performance evaluation and comparison with a wide range of OSR methods, which shows the superior performance of the proposed S2CA method.

2 Related Work

2.1 Open-set Recognition

Open-set recognition has long been an attractive topic in the computer vision community. Early studies (Scheirer et al., 2013; 2014; Jain et al., 2014; Scherreik & Rigling, 2016) often involve traditional machine learning models, such as SVM, and use the probabilistic methods to threshold the prediction score.

OpenMax (Bendale & Boult, 2015) is one of the first OSR methods based on deep neural networks, which determines the center of each known class in the logit space and then creates a statistical model of the distances between correctly classified samples. It uses extreme value theory (EVT) to detect outliers by fitting a weibull function to the tail of the distance distribution. Many subsequent works involve generative models. G-OpenMax (Ge et al., 2017) trains a Generative Adversarial Network (GAN) to synthesize open-set samples and uses these to estimate open-set class activations. Neal et al. (2018) propose a data augmentation technique called counterfacutal image generation. This approach uses GANs to generate images that are visually similar to known ones, but are actually from unknown classes. The network is trained to have low confidence scores for these "known unknowns". Kong & Ramanan (2021) propose OpenGAN, which is suitable for both image classification and semantic segmentation. Generative models can do more than synthesize known unknown classes. In (Oza & Patel, 2019), the auto-encoder only learns feature representations from known classes, which leads to higher reconstruction errors for unknown classes. Yoshihashi et al. (2019) integrate classification and input reconstruction in model training, which enhances the learned feature representation. Perera et al. (2020) combine the auto-encoder with a multi-task classifier trained by minimizing a self-supervision loss and a classification loss. The self-supervision learning branch incorporates random transformations to improve the descriptiveness of feature representations. Miller et al. (2021) propose a simple OSR method, which forces known classes to form tight clusters around class anchors in logit space and reject unknown objects based on their distance to known classes. While this model is easier to train than the previous generative models, anchoring class centers at equal distances can restrict the learning of semantically meaningful features (Miller et al., 2021).

2.2 Out-of-distribution Detection

Detection of out-of-distribution (OOD) samples is a concept closely related to OSR, and the two terms are sometimes used interchangeably. However, OOD detection is slightly different from OSR in that OOD samples can be completed unrelated to the training set, whereas OSR aims to reject valid but unknown classes (Gillert & von Lukas, 2021).

OOD detection methods usually map the input x onto a single scalar that indicates OOD-ness. Assuming that the network is more confident about known objects, the OOD-ness leads to softmax scores that are higher for known classes than the unknown ones, making it a baseline OOD detection method (Hendrycks & Gimpel, 2017). Unknown objects tend to have a uniform softmax distribution, leading to higher entropy (Chan et al., 2021). Hsu et al. (2020) propose ODIN, which adds small perturbations to the input thereby

increasing the softmax score gap between known and unknown objects. Hendrycks et al. (2022) use the maximum logit as the OOD score, and show that it outperms the softmax baseline (Hendrycks & Gimpel, 2017). Kullback–Leibler divergence between the softmax probabilities and class templates has also been used as the OOD score (Hendrycks et al., 2022). Liu et al. (2020) measure OOD-ness of objects based on the Helmholtz free energy of the network output. Lee et al. (2018) use the Mahalanobis distance (MD) between the feature vector and the closest class-conditional Gaussian distribution to measure OOD-ness. Wang et al. (2022) propose Virtual-logit Matching (ViM), which combines class-agnostic information in the feature space and the class-dependent information in the logit space for OOD detection.

2.3 Contrastive Learning

Contrastive learning aims to learn a representation of data such that similar instances are close together in the representation space, while dissimilar instances are far apart. He et al. (2020) propose the Momentum Contrast (MoCo) for large-scale unsupervised learning, which converts the image-based contrastive learning into a dictionary look-up problem. Chen et al. (2020) propose SimCLR, which maximizes the similarity between augmented versions of the same image (positive pairs) and minimizes the similarity between different images (negative pairs). SimCLR employs an MLP projection head after the encoder network, which introduces non-linear transformations to the feature space. It also incorporates a normalization step after the projection head, which ensures that the representations have unit length, facilitating the use of cosine similarity to measure similarity between pairs. Khosla et al. (2020) propose Supervised Contrastive Learning (SCL), which extends SimCLR (Chen et al., 2020) to fully supervised scenarios. SCL contrasts all samples from the same class as positives against the negatives from the remainder of the batch, thereby forming tight class clusters in the feature space. Contrastive learning methods (He et al., 2020; Chen et al., 2020; Khosla et al., 2020) usually train an encoder with contrastive loss first, then freeze the encoder and train a linear classifier. Wang et al. (2021) argue that this two-stage learning may not be the most effective approach in a fully supervised setting, as it can reduce the compatibility between the feature encoder and the classifier. To address this issue, Wang et al. (2021) propose a hybrid framework that allows simultaneous learning of features and classifiers for long-tail image classification.

3 Proposed Method

Given the various data modalities and the different model architectures used for object recognition in driving scenes, we propose to perform OSR in feature space. This makes our method efficient and applicable to different data modalities and encoder networks.

Fig. 2 shows an overview of the proposed S2CA framework. We train a feature encoder with Contrastive Class Anchor learning, which combines Supervised Contrastive Learning (SCL) (Khosla et al., 2020) and Class Anchor Clustering (CAC) (Miller et al., 2021). Contrastive Class Anchor learning aims to train the encoder so that the known classes show intra-class compactness and inter-class sparsity in the feature space. In this way, feature representations of known and unknown classes are separated, allowing us to model the distribution of known classes and reject unknown classes.

3.1 Model Training

S2CA consists of a contrastive learning branch and a CAC loss based classifier learning branch. They share the same encoder f_e but optimize different objective functions. The two branches together constitute Contrastive Class Anchor Learning.

3.1.1 Contrastive Learning Branch

Supervised Contrastive Learning (SCL) (Khosla et al., 2020) encourages objects of the same class to be tightly clustered in the feature space, while clusters belonging to different classes are far apart. These characteristics are beneficial to the OSR task. The intra-class compactness of known classes helps maintain a robust recognition performance for known objects, while the inter-class sparsity makes it easier to reject unknown objects based on their distance to the known classes.

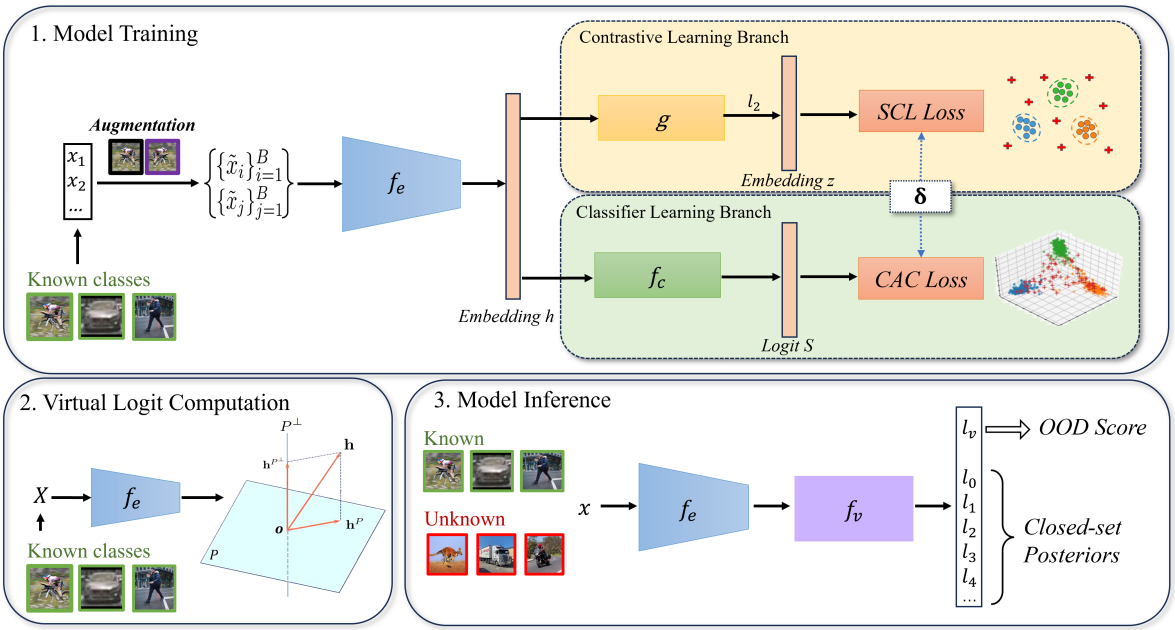


Figure 2. Overview of the proposed S2CA framework. We use a shared encoder f_e to extract feature embeddings from the closed set and feed these to a supervised contrastive learning branch (SCL) and a CAC loss based classifier learning branch (step 1). The contrastive learning branch contains a non-linear MLP g combined with l_2-normalization, which translate the feature embeddings for contrastive loss. The classifier learning branch is a single linear layer f_c, trained by the CAC loss. SCL is a batch contrastive approach. For each instance, SCL contrasts all instances (including the augmented instance) from the same class as positives against negatives from the remainder of the batch. Therefore, we create augmented pairs for a batch of training samples x of size B from the known classes, and the subsequent embeddings are also in pairs. A curriculum is designed to control the weight of these two branches, i.e., δ and $1-\delta$, during model training. Once training is complete, we compute the distribution of virtual logits l_v (step 2) based on the projection of embeddings $\|\mathbf{h}^{\mathbf{P}^\perp}\|$ which indicate OOD-ness. During model inference (step 3), we replace f_c with a $N+1$ classifier f_v, which produces closed-set posteriors and a virtual logit l_v. We threshold the virtual logit-based OOD score to reject unknown samples and classify the known ones using the normal logits.

Let x and y denote the input instance and the corresponding label, with $y \in N$ known classes. For each instance, S2CA creates two augmented views, $\tilde{x}_i$ and $\tilde{x}_j$, of x from a predefined family $\mathcal{T}$. For image input, augmentation may include random cropping, random flipping and color jittering (for range images, we only use random flipping). For lidar point clouds, we apply geometric transformations such as random translation, random rotation and random scaling. For a mini-batch of size B, the corresponding augmentations are denoted as $\{\tilde{x}_i\}_{i=1}^B$ and $\{\tilde{x}_j\}_{j=1}^B$.

When x is passed to the encoder network $f_e(\cdot)$, it produces the feature embedding $h = f_e(x)$. Then the MLP projector $g(\cdot)$ maps h to a 128-dimensional vector. Applying l_2 normalization to this vector gives a normalized embedding z. The projector $g(\cdot)$ itself is invariant to data transformation, while it helps the encoder $f_e(\cdot)$ preserve richer information for downstream tasks (Chen et al., 2020). The SCL loss (Khosla et al., 2020) is used to maximize the agreement between $\{\tilde{z}_i\}_{i=1}^B$ and $\{\tilde{z}_j\}_{j=1}^B$ for instances of the same class. For each vector z_i, the SCL loss is:

$$\mathcal{L}_i^{sup} = \frac{-1}{\left|\{z_i^+\}\right|} \sum_{z_j \in \{z_i^+\}} \log \frac{\exp\left(z_i \cdot z_j / \tau\right)}{\sum_{k \neq i} \exp\left(z_i \cdot z_k / \tau\right)} \tag{1}$$

where $\{z_i^+\}$ represents all instances in the batch that have the same label $\tilde{y}_i$, $\left|\{z_i^+\}\right|$ is the number of instances in $\{z_i^+\}$, and τ is a temperature scaling parameter.

Accordingly, the loss for each mini-batch is defined as:

$$\mathcal{L}_{SCL} = \sum_{i=1}^{2B} \mathcal{L}_i^{sup} \tag{2}$$

3.1.2 Classifier Learning Branch

In the classifier learning branch, the feature embedding h is passed to a linear classification layer f_c to produce the N-class logit vector s. The CAC loss (Miller et al., 2021) is defined based on the distance between the logit vectors s and a set of anchored class centers $C = (c_1, \cdots, c_n)$ with a predefined magnitude α set equally for all classes. Let d denote a vector of Euclidean distances between s and each anchored class center in C, and $f(\cdot)$ denote the classification function, the CAC loss is written as:

$$\mathcal{L}_{CAC} = log(1 + \sum_{j \neq y}^{B} e^{d_y - d_j}) + \lambda \left\| f(x) - c_y \right\|_2 \tag{3}$$

where λ is a hyperparameter to balance the two loss terms.

The first term of $\mathcal{L}_{CAC}$ is modified from the tuplet loss (Sohn, 2016). It forces s to maximize the difference between the distance to the correct class center and the distances to all other class centers. The second term minimizes the distance between s and the correct class center.

3.1.3 Transition from Contrastive Learning to Classifier Learning

To ensure compatibility between the feature encoder and the classifier, we follow a curriculum strategy (Zhou et al., 2020) to adjust the weighting of the two branches. The model will first focus on supervised contrastive learning (Khosla et al., 2020) and then smoothly transitions to classifier learning as the training progresses. The final objective function is the combination of Eq. (2) and Eq. (3).

$$\mathcal{L}_{S2CA} = \delta \mathcal{L}_{SCL} + (1 - \delta) \mathcal{L}_{CAC} \tag{4}$$

Note that $\mathcal{L}_{CAC}$ in Eq. 4 is the sum of each individual loss in the mini-batch (like the batch form of the SCL loss shown in Eq. 2). δ is a parameter that decreases as the training progresses. For each epoch t_i, $\delta = 1 - \frac{t_i}{t_{max}}$ where t_{max} is the maximum epoch number. Initially, δ is large, and the model focuses on optimizing the SCL loss to learn semantically meaningful features. As the training progresses, δ is reduced and the model smoothly shifts its focus to the CAC loss, which fine-tunes the model from a global perspective.

The combination of SCL and CAC has several advantages. While using the CAC loss alone can restrict the discriminative power of the learned features (Miller et al., 2021), the SCL helps the encoder learn a richer feature space. The SCL also benefits from the CAC. As a batch contrastive algorithm, SCL only contrasts samples in the same mini-batch. To achieve better performance, SCL requires larger batch sizes or more training epochs, which increases the computational complexity of the training (Khosla et al., 2020). The CAC mitigates this by comparing each individual sample with its corresponding class anchor. In each epoch, all training instances are pulled to the corresponding class anchors to form tight clusters. This results in faster convergence of contrastive learning with smaller batch sizes.

3.2 Model Inference

For model inference, we use virtual logit (Wang et al., 2022) to extract information in the feature space and combine it with ordinary logit to form the OOD score and reject unknown objects. Unlike the distance-based rejection score used by Miller et al. (2021), which ignores the class-agnostic information in the feature space, the virtual logit matching (ViM) combines the class-agnostic information in the feature space with the class-dependent information to reject unknown objects and classify the known ones.

3.2.1 Virtual Logit

The virtual logit is the scaled norm of the projection of the feature vector h to the orthogonal complement of the principal subspace of the feature space. To simplify the subsequent logit computation, we offset h with a new origin $o = -(W^T)^+ b$, where W is the weight in f_c, b is the bias and $(\cdot)^+$ is the Moore-Penrose inverse. This transforms the feature vectors to a new bias-free coordinate system. To distinguish these from the previous features h, the feature vectors in the new coordinate system are denoted $\mathbf{h}$. We further denote the feature matrix of the training set as $\mathbf{H}$.

The principal subspace P of the feature space is defined by the eigen decomposition of $\mathbf{H}^T\mathbf{H}$:

$$\mathbf{H}^T\mathbf{H} = Q\Lambda Q^{-1} \tag{5}$$

where Λ contains the eigenvalues in descending order, and the first D columns of Q define the D-dimensional principal subspace P. Let $P^\perp$ be the orthogonal complement of P, and $\mathbf{h}^{P^\perp}$ the projection of $\mathbf{h}$ to $P^\perp$. After this projection, known data tend to have a smaller norm, whereas unknown data exhibit a larger norm. This facilitates the differentiation between known and unknown samples. The virtual logit is then obtained as (Wang et al., 2022):

$$l_v = \beta \|\mathbf{h}^{P^\perp}\| \tag{6}$$

where β is the scaling constant:

$$\beta = \frac{\sum_{i=1}^{K} max_{j=1,\ldots,N} \{l_j^i\}}{\sum_{i=1}^{K} \|\mathbf{h}_i^{P^\perp}\|} \tag{7}$$

where N is the number of known classes, K is the number of training samples, and l_j^i is the $j-$th logit of $\mathbf{h}_i$.

Following Wang et al. (2022), we obtain the ViM score by appending the energy score (Liu et al., 2020) to the virtual logit:

$$\upsilon = \beta \|\mathbf{h}^{P^\perp}\| - \ln \sum_{i=1}^{N} e^{l_i} \tag{8}$$

which has a better performance than each individual score.

3.2.2 Inference Pipeline

During inference, the final classification layer f_c is replaced by a new module f_v, which produces a logit vector of length $N+1$, that is N normal logits and one virtual logit. f_v is able to invoke both the encoder f_e and the classification layer f_c. It uses the encoder f_e to extract the feature matrix H from the training set X, project H into a bias-free coordinate system and calculate the D dimensional principal subspace P. We calculate the scaling variable β based on Eq. (7). For normal logits, f_v directly passes the feature embedding to f_c.

Now, we have all the parameters to compute the OOD score υ (Eq. (8)). A larger value for υ indicates that the object is more likely to be unknown. As such, unknown objects can be rejected by setting a threshold θ. The decision rule is as follows:

$$prediction = \begin{cases} class\ i = argmax(l), & \upsilon \leq \theta \\ unknown, & \upsilon > \theta \end{cases} \tag{9}$$

The threshold θ can be determined without requiring auxiliary unknown data by analyzing the distribution of OOD scores of the known samples in the training set.

Table 1. Data Modalities and Encoder Networks

Input Type	Network	Architecture
RGB Image	WRN-40-2	Wide ResNet (Zagoruyko & Komodakis, 2016)
Range Image	WRN-40-2	Wide ResNet (Zagoruyko & Komodakis, 2016)
3D Voxel Grid	VoxNet	Volumetric CNN (Maturana & Scherer, 2015)
3D Voxel Grid	LightNet	Volumetric CNN (Zhi et al., 2018)
Point Cloud	DGCNN	Dynamic Graph CNN (Wang et al., 2019)
Point Cloud	PCT	Point Transformer (Guo et al., 2021)

3.3 Implementation Details

For training, we use the Cosine Annealing Learning Rate (Loshchilov & Hutter, 2017), with a maximum learning rate of 0.1 and a minimum of 10^{-4}. In the initial epochs, the learning rate decreases slowly to speed up the contrastive learning process. The learning rate begins to drop rapidly midway through the training to facilitate classifier learning. Towards the end of the training, the learning rate becomes very small, so that the classifier can be fine-tuned. The training takes up to 200 epochs, and is optimized by stochastic gradient descent (SGD) with a momentum of 0.9. For the SCL loss, the temperature is set to 0.1 and the batch size is limited to 256. For the CAC loss, the anchor magnitude α is set to 10 and the weight variable λ is set to 0.3. For the virtual logit, the dimensionality of the principal subspace is set to 64.

4 Experiments

4.1 Comparison with State-of-the-art Methods

We compare the performance of S2CA with existing OSR and OOD detection methods, for which source code is available from PyTorch-OOD library (Kirchheim et al., 2022) or GitHub. Specifically, we select SoftMax (Hendrycks & Gimpel, 2017), ODIN (Hsu et al., 2020), MD (Lee et al., 2018), Energy (Liu et al., 2020), Entropy (Chan et al., 2021), MaxLogit (Hendrycks et al., 2022), KL-M (Hendrycks et al., 2022), ViM (Wang et al., 2022), and CAC (Miller et al., 2021). The selected methods are compatible with different data modalities and encoder networks.

4.2 Encoder Networks

To evaluate the open-set recognition performance with both image and lidar data several encoder networks were used as shown in Tab. 1. For RGB image data, we use a 40-layer wide residual network with a widening factor 2 (WRN-40-2) (Zagoruyko & Komodakis, 2016) for feature extraction. For lidar data, we use encoder networks operating on raw point clouds, range images, and 3D voxel grids. For range images, we use the same Wide ResNet as for RGB image data. For lidar data represented as 3D voxel grids, we use VoxNet (Maturana & Scherer, 2015) and LightNet (Zhi et al., 2018) as the encoder networks. For lidar point clouds, we use DGCNN (Wang et al., 2019) and Point Cloud Transformer (PCT) (Guo et al., 2021), which directly take point clouds as input.

4.3 Datasets

We use two public autonomous driving datasets for evaluation: nuScenes (Caesar et al., 2019) and KITTI (Geiger et al., 2013). For the object detection task, nuScenes has 23 classes, while KITTI (Geiger et al., 2013) has 8 classes. Each dataset was divided into three parts: x_{train}, x_{test} and x_{OOD}. x_{train} and x_{test} contained samples of known classes for training and testing. x_{OOD} refers to unknown classes (out of distribution). We selected the classes with the fewest annotations as unknown. In nuScenes, unknown classes were *stroller*, *personal mobility vehicle*, *animal*, and *ambulance*. In KITTI, unknown classes were *truck*, *misc* (miscellaneous objects, such as trailers and segways), *tram*, and *sitting person*. We first located each object using the given bounding box and then extracted these from the image and lidar point cloud data. Heavily occluded or completely invisible objects were filtered out. Following Neal et al. (2018), we calculate the *openness* of our datasets according to: $openness = 1 - \sqrt{\frac{K}{K+U}}$, where K is the number of known classes and U is the

number of unknown classes. Accordingly, the *openness* of our nuScenes dataset was found to be 9.1%, while the *openness* of KITTI was 29.3%.

4.4 Evaluation Metrics

The effectiveness of an open set recognition model can be evaluated by its overall classification accuracy or F-score when detecting unknown classes in mixed data. However, these combined metrics are affected not only by the quality of the OSR model, but also by the calibration parameters chosen (Neal et al., 2018). As a result, we evaluate the performance of OSR models using two metrics: Area Under the ROC Curve (AUROC) and Closed Set Accuracy.

Area Under the ROC Curve (AUROC) AUROC uses all possible thresholds to plot an ROC curve of the True Postive Rate (TPR) against False Positive Rate (FPR). Ideally, known objects should have a lower OOD-ness score, whereas unknown objects should have a higher score. If all unknown classes have higher scores than the known ones, the TPR will be 1 and the FPR will be 0 after trying all thresholds, resulting in a perfect AUROC value of 1. Lower AUROC values indicate that unknown objects are mixed with the known ones.

Closed-set Accuracy An open-set classifier must maintain its capacity for recognizing closed-set objects as well. As a result, we will further evaluate the classification accuracy of known classes.

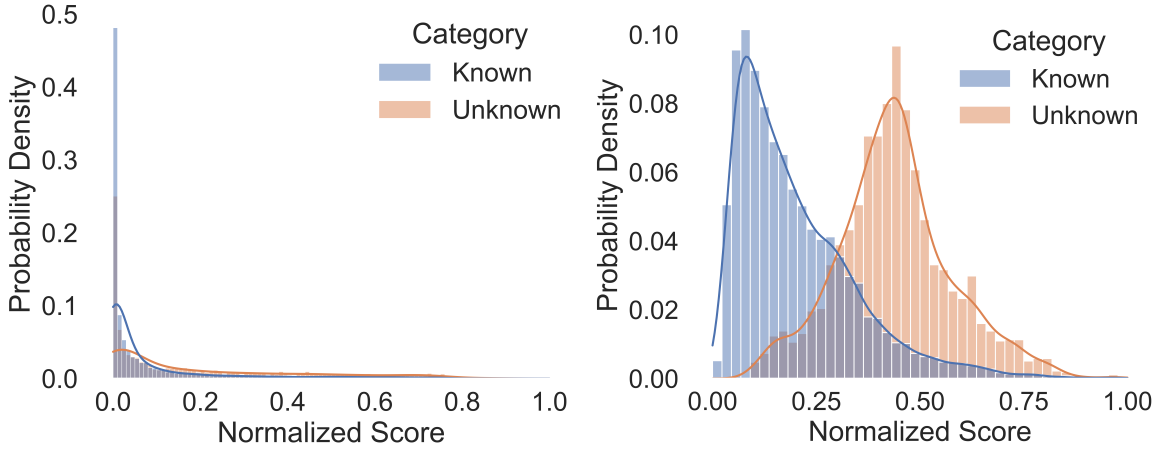
5 Results and Discussion

5.1 Open-set Recognition Performance

Table 2. Performance of S2CA compared with state of the art OSR methods. Open-set recognition performance is measured using AUROC (%). For each encoder network, the best and second-best OSR results are emphasized.

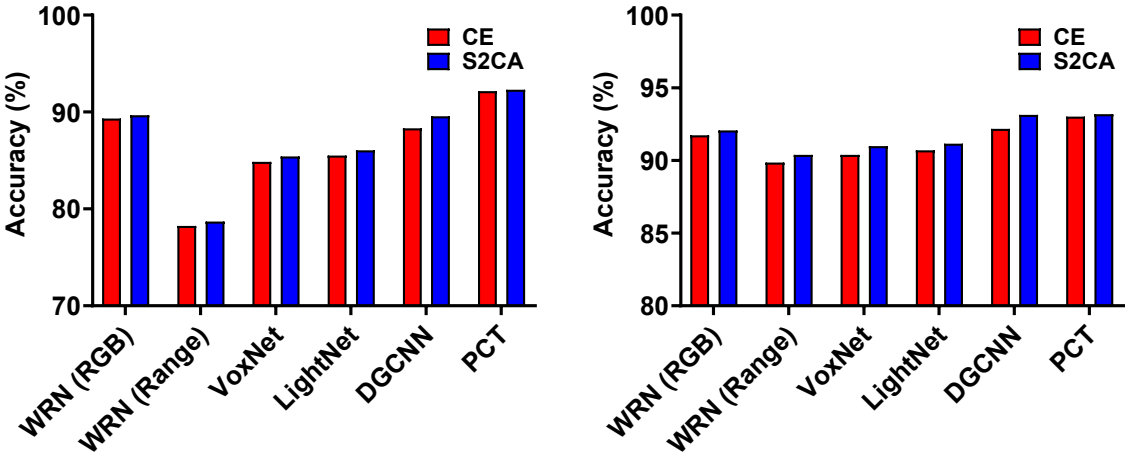
Dataset	OSR Method	Encoder Network					
		WRN (RGB)	WRN (Range)	VoxNet (Voxel)	LightNet (Voxel)	DGCNN (Point)	PCT (Point)
nuScenes	SoftMax (Hendrycks & Gimpel, 2017)	79.26	60.52	71.78	69.17	80.50	80.92
	ODIN (Hsu et al., 2020)	73.44	34.30	71.70	69.30	81.16	80.20
	MD (Lee et al., 2018)	78.07	23.42	74.43	72.02	78.94	80.51
	Energy (Liu et al., 2020)	79.72	62.53	73.93	71.11	**81.73**	80.82
	Entropy (Chan et al., 2021)	79.56	61.80	73.05	70.27	81.14	81.09
	MaxLogit (Hendrycks et al., 2022)	79.65	62.04	73.20	70.36	81.45	81.25
	KL-M (Hendrycks et al., 2022)	78.42	52.58	56.87	52.93	61.69	67.52
	ViM (Wang et al., 2022)	80.73	**70.59**	**75.53**	**74.59**	80.41	82.28
	CAC (Miller et al., 2021)	**85.32**	69.63	72.40	74.04	77.65	**82.80**
	S2CA (ours)	**82.88**	**70.99**	**76.05**	**77.25**	**83.12**	**83.21**
KITTI	SoftMax (Hendrycks & Gimpel, 2017)	74.07	75.68	76.88	77.92	72.50	75.40
	ODIN (Hsu et al., 2020)	62.09	61.72	76.98	77.90	70.85	75.31
	MD (Lee et al., 2018)	79.78	61.23	74.01	76.77	**86.90**	**78.19**
	Energy (Liu et al., 2020)	74.91	72.10	76.03	77.49	72.89	74.88
	Entropy (Chan et al., 2021)	74.08	75.66	**77.01**	**78.29**	72.60	75.27
	MaxLogit (Hendrycks et al., 2022)	74.90	71.99	76.07	77.48	72.82	74.93
	KL-M (Hendrycks et al., 2022)	62.55	69.15	61.70	60.47	72.88	75.33
	ViM (Wang et al., 2022)	**80.38**	**76.56**	76.68	77.41	85.27	77.39
	CAC (Miller et al., 2021)	73.71	72.12	74.44	77.58	71.83	74.39
	S2CA (ours)	**81.23**	**79.64**	**77.39**	**79.28**	**87.48**	**81.94**

Tab. 2 shows the performance of the OSR methods with different encoder networks as measured by AUROC. It can be seen that overall S2CA achieves leading performance on both datasets across different data modality and encoder networks. Interestingly, the OSR methods perform well with lidar data when paired with powerful encoder networks. Point-based encoder networks generally outperform projection-based and voxel-based models. The range image representation of lidar data does not seem to benefit the open-set recognition performance, especially when the lidar point cloud is sparse. This can be seen from the performance of the range image-based models on nuScenes, which is noticeably worse than on KITTI. This is because nuScenes lidar data are captured by a 32-channel lidar, so the point cloud is relatively sparse and the range images have a lower resolution as compared to KITTI which uses a 64-channel lidar. Notably, ODIN (Hsu et al.,



(a) SoftMax (b) S2CA

Figure 3. Distribution of OOD scores for known and unknown samples obtained by the Softmax (inverted) baseline (a), and the proposed S2CA (b). For fair comparison, all scores are normalized to [0,1].



(a) nuScenes (b) KITTI

Figure 4. Closed-set accuracy of S2CA with different encoder networks compared with the cross-entropy baseline on nuScenes (a) and KITTI (b).

2020) and MD (Lee et al., 2018) perform quite poorly on nuScenes range images. This can be attributed to the addition of noise during preprocessing step in these methods, which has a negative impact on the recognition performance. The comparable performance of the OSR methods with different image-based and lidar-based encoder networks implies that the power of open-set recognition lies in the feature learning. The modality of the data is not as crucial as the information it carries.

The superior performance of the proposed S2CA can be attributed to its ability to learn discriminative features, which results in OOD scores that are higher for unknown classes. Fig. 3 shows the probability density distributions of the normalized OOD scores for both known and unknown classes in the KITTI dataset. For the SoftMax baseline (Hendrycks & Gimpel, 2017) in Fig. 3a, both distributions exhibit a long-tailed pattern and there is a high degree of overlap between them, whereas S2CA (Fig. 3b) presents a noticeable difference between the two distributions, making it easier to determine an appropriate threshold to distinguish the known and unknown classes effectively. The significant difference between the two figures explains the superior performance of S2CA in the OSR task.

(a) Van (b) Truck

Figure 5. Examples of misclassified samples. Van (a) is a known class, while Truck (b) is unknown.

Fig. 4 shows the closed-set accuracy of S2CA as compared with the conventional cross-entropy baseline. It can be seen that for all encoder networks the S2CA framework maintains a comparable closed-set performance as the conventional cross-entropy baseline. This again indicates that the ability of S2CA to extract rich features from the data is not only useful for rejecting unknown objects, but also helps maintain a high classification accuracy on known classes.

A close examination of the errors reveals that most misclassified samples are objects that are similar in appearence and are too far from the sensor. Fig. 5 shows an example of a known (van) and an unknown object (truck) which were misclassified by S2CA. In such cases, where the unknown object looks very similar to a known class, rejecting the unknown objects and correctly classifying the known ones is challenging.

5.2 Ablation Study

The S2CA framework consists of three important components: contrastive learning, class anchor clustering, and virtual logit matching. To evaluate the contribution of each individual component to the final performance, we systematically remove these components from the S2CA framework to create a set of baseline models:

CLS - A standard N class classifier, denoted f, trained with cross-entropy loss. Unknown objects are identified using the SoftMax score.

CLS + ViM (CLS V.) - This baseline uses the same classifier f, but rejects unknown objects based on the ViM (Wang et al., 2022) score.

CLS + CAC + ViM (C.C.V) - The classifier f is trained with CAC (Miller et al., 2021) loss, but rejects unknown objects based on ViM (Wang et al., 2022) score.

Contrastive CLS (Cont. CLS) - We further add the supervised contrastive learning (SCL) (Khosla et al., 2020) branch to f. The model rejects unknown objects based on the ViM (Wang et al., 2022) score.

S2CA - The complete S2CA framework.

Tab. 3 illustrates the results of the ablation study. The complete S2CA framework with different encoder networks consistently outperforms all competing baseline models in terms of AUROC. The virtual logit based score and contrastive learning make the largest contributions to the overall performance. The CAC loss is useful when combined with contrastive learning, however using CAC alone provides limited improvements in open-set recognition performance. Contrastive learning helps the model learn more discriminative features, while CAC loss improves intra-class compactness resulting in higher confidence in recognizing known classes.

Table 3. Performance of the complete S2CA framework in terms of AUROC compared to other baselines

Dataset	Baseline	Encoder Network					
		WRN (RGB)	WRN (Range)	VoxNet (Voxel)	LightNet (Voxel)	DGCNN (Point)	PCT (Point)
nuScenes	CLS	79.26	60.52	71.78	69.17	80.5	80.92
	CLS V.	80.73	70.59	75.53	74.55	80.41	82.28
	C.C.V.	80.32	70.63	75.4	74.04	80.43	82.8
	Cont. CLS	81.59	70.89	75.8	76.12	82.53	82.87
	S2CA	**82.88**	**70.99**	**76.05**	**77.25**	**83.12**	**83.21**
KITTI	CLS	74.07	75.68	76.88	77.92	72.5	75.4
	CLS V.	80.38	76.56	76.68	77.41	85.27	77.39
	C.C.V.	80.71	76.12	76.44	77.58	85.83	77.56
	Cont. CLS	80.33	77.58	77.23	78.56	86.21	80.04
	S2CA	**81.23**	**79.64**	**77.39**	**79.28**	**87.48**	**81.94**

6 Conclusion

We proposed an effective and adaptive OSR method based on contrastive class anchor learning. Our experimental results show that the proposed method achieves state-of-the-art performance with different data modalities and encoder networks. Future work will focus on developing a complete perception pipeline that includes class-agnostic object detection, open-set object recognition, active learning, and incremental learning, enabling autonomous vehicles to incrementally learn and improve their driving scene recognition capabilities.

7 Acknowledgement

This research was supported by The University of Melbourne's Research Computing Services and the Petascale Campus Initiative. The first author acknowledges the financial support from the University of Melbourne through the Melbourne Research Scholarship.

References

Abhijit Bendale and Terrance Boult. Towards open world recognition. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1893–1902, 2015. doi: 10.1109/CVPR.2015.7298799.

Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. *arXiv preprint arXiv:1903.11027*, 2019.

Robin Chan, Matthias Rottmann, and Hanno Gottschalk. Entropy maximization and meta classification for out-of-distribution detection in semantic segmentation. In *ICCV*, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

ZongYuan Ge, Sergey Demyanov, Zetao Chen, and Rahil Garnavi. Generative openmax for multi-class open set classification. In *Proceedings of the British Machine Vision Conference*, pp. 42.1–42.12, 2017.

Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013.

Chuanxing Geng, Sheng-Jun Huang, and Songcan Chen. Recent advances in open set recognition: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3614–3631, 2021. doi: 10.1109/TPAMI.2020.2981604.

Alexander Gillert and Uwe Freiherr von Lukas. Towards combined open set recognition and out-of-distribution detection for fine-grained classification. In *16th International Joint Conference on Computer Vision*, 2021.

Meng-Hao Guo, Jun-Xiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph R. Martin, and Shi-Min Hu. Pct: Point cloud transformer. *Computational Visual Media*, 7(2):187–199, 2021.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9726–9735, 2020. doi: 10.1109/CVPR42600.2020.00975.

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)*, pp. 1010–1019, 2017.

Dan Hendrycks, Steven Basart, Mantas Mazeika, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings. In *International Conference on Machine Learning (ICML)*, 2022.

Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10948–10957, 2020. doi: 10.1109/CVPR42600.2020.01096.

Lalit P Jain, Walter J Scheirer, and Terrance E Boult. Multiclass open set recognition using probability of inclusion. In *European Conference on Computer Vision*, 2014.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Adv. Neural Inform. Process. Syst.*, 2020.

Konstantin Kirchheim, Marco Filax, and Frank Ortmeier. Pytorch-ood: A library for out-of-distribution detection based on pytorch. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 4351–4360, June 2022.

S. Kong and D. Ramanan. Opengan: Open-set recognition via open data generation. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 793–802, Los Alamitos, CA, USA, oct 2021. IEEE Computer Society. doi: 10.1109/ICCV48922.2021.00085. URL `https://doi.ieeecomputersociety.org/10.1109/ICCV48922.2021.00085`.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Adv. Neural Inform. Process. Syst.*, 2018.

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. In *Adv. Neural Inform. Process. Syst.*, 2020.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *ICLR*, 2017.

Daniel Maturana and Sebastian Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 922–928, 2015. doi: 10.1109/IROS.2015.7353481.

Dimity Miller, Niko Sünderhauf, Michael Milford, and Feras Dayoub. Class anchor clustering: A loss for distance-based open set recognition. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 3569–3577, 2021. doi: 10.1109/WACV48630.2021.00361.

Lawrence Neal, Matthew Olson, Xiaoli Fern, Weng-Keen Wong, , and Fuxin Li. Open set learning with counterfactual images. Open set learning with counterfactual images. In *ECCV*, pp. 613–628, 2018.

Poojan Oza and Vishal M. Patel. C2ae: Class conditioned auto-encoder for open-set recognition. In *CVPR*, pp. 2302–2311, 2019. doi: 10.1109/CVPR.2019.00241.

Pramuditha Perera, Vlad I. Morariu, Rajiv Jain, Varun Manjunatha, Curtis Wigington, Vicente Ordonez, and Vishal M. Patel. Generative-discriminative feature representations for open-set recognition. In *CVPR*, pp. 11811–11820, 2020. doi: 10.1109/CVPR42600.2020.01183.

Walter J. Scheirer, Anderson de Rezende Rocha, Archana Sapkota, and Terrance E. Boult. Toward open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(7):1757–1772, 2013. doi: 10.1109/TPAMI.2012.256.

Walter J. Scheirer, Lalit P. Jain, and Terrance E. Boult. Probability models for open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(11):2317–2324, 2014. doi: 10.1109/TPAMI.2014.2321392.

Matthew D. Scherreik and Brian D. Rigling. Open set recognition for automatic target classification with rejection. *IEEE Transactions on Aerospace and Electronic Systems*, 52(2):632–642, 2016. doi: 10.1109/TAES.2015.150027.

Kihyuk Sohn. Improved deep metric learning with multiclass n-pair loss objective. In *Adv. Neural Inform. Process. Syst.*, 2016.

Haoqi Wang, Zhizhong Li, Litong Feng, and Wayne Zhang. Vim: Out-of-distribution with virtual-logit matching. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4911–4920, 2022. doi: 10.1109/CVPR52688.2022.00487.

Peng Wang, Kai Han, Xiu-Shen Wei, Lei Zhang, and Lei Wang. Contrastive learning based hybrid networks for long-tailed image classification. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 943–952, 2021. doi: 10.1109/CVPR46437.2021.00100.

Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds. *ACM Trans. Graph.*, 38(5), oct 2019. ISSN 0730-0301. doi: 10.1145/3326362. URL `https://doi.org/10.1145/3326362`.

Ryota Yoshihashi, Wen Shao, Rei Kawakami, Shaodi You, Makoto Iida, and Takeshi Naemura. Classification-reconstruction learning for open-set recognition. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4011–4020, 2019. doi: 10.1109/CVPR.2019.00414.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *BMVC*. BMVA Press, 2016.

Shuaifeng Zhi, Yongxiang Liu, Xiang Li, and Yulan Guo. Toward real-time 3d object recognition: A lightweight volumetric cnn framework using multitask learning. *Computers & Graphics*, 71:199–207, 2018. ISSN 0097-8493. doi: https://doi.org/10.1016/j.cag.2017.10.007. URL `https://www.sciencedirect.com/science/article/pii/S0097849317301735`.

Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9716–9725, 2020. doi: 10.1109/CVPR42600.2020.00974.

A Appendix