OPEN-SET RECOGNITION VIA AUGMENTATION-BASED SIMILARITY LEARNING

Sepideh Esmaeilpour, Lei Shu, Bing Liu
Department of Computer Science, University of Illinois at Chicago
`sesmae2@uic.edu, shulindt@gmail.com, liub@uic.edu`

ABSTRACT

The primary assumption of conventional supervised learning or classification is that the test samples are drawn from the same distribution as the training samples, which is called *closed set learning or classification*. In many practical scenarios, this is not the case because there are unknowns or unseen class samples in the test data, which is called the *open set scenario*, and the unknowns need to be detected. This problem is referred to as the *open set recognition* problem and is important in safety-critical applications. We propose to detect unknowns (or unseen class samples) through learning pairwise similarities. The proposed method works in two steps. It first learns a closed set classifier using the seen classes that have appeared in training, and then learns how to compare seen classes with pseudo-unseen (automatically generated unseen class samples). The pseudo-unseen generation is carried out by performing distribution shifting augmentations on the seen or training samples. We call our method OPG (Open set recognition based on Pseudo unseen data Generation). The experimental evaluation shows that the learned similarity-based features can successfully distinguish seen from unseen in benchmark datasets for open set recognition.

1 INTRODUCTION

Deep classification models trained with cross-entropy loss are expected to produce a high probability score for the ground-truth class. Although this is the expected behaviour, the system can easily be fooled when encountering test instances from unknown or unseen classes. The reason is that in this case, the test data distribution does not follow the distribution of the train data. This situation cannot be avoided or ignored in open world settings. In fact, it is a well-studied phenomena that deep models are over-confident on unseen or distributionally shifted instances (Bendale & Boult, 2016). Since the cross entropy loss aims at increasing the score for the ground-truth class in training, the learned features tend to be more discriminative than descriptive. Regardless whether it is from a seen or unseen class, as long as a test sample resides in a seen class's space (even if the test sample is far from the distribution of the seen class), it will be detected as a seen class sample. The detection of such out-of-distribution (OOD) samples is of critical importance for AI agents deployed in safety-critical applications such as medical diagnosis and autonomous driving because detecting unseen as normal/seen class samples can be catastrophic.

The terms *open set recognition (OSR)* and *out-of-distribution(OOD) detection* co-occur in the literature frequently. OSR methods often try to detect open-set samples by assuring that the closed-set accuracy remains intact. Early techniques for OSR are mainly SVM-based methods (Schölkopf et al., 1999; Scheirer et al., 2013; Scheirer et al., 2014). Due to the great success of deep learning models in classification tasks, most recent methods for OSR are based on deep learning models. Particularly, since deep models tend to output high confidence scores for the majority of unseen or fooling samples, OSR cannot be directly performed based on the output scores. Bendale & Boult (2016) proposes a calibration method that uses the penultimate layer activation of a sample. Rejection or acceptance is done according to the calibrated OpenMax scores. Generative OpenMax (Ge et al., 2017) is an extension of OpenMax that follows the same protocol except that it uses generated samples to synthesize pseudo-unseen and trains a $k + 1$ class classifier where class $k + 1$th represents the space of unseen. Perera et al. (2020) proposes a generative-discriminative technique that combines the advances in self-supervised learning and generative models. This method benefits from a generative data-augmentation technique per seen sample in training. A recent distance-based method (Miller et al., 2021) encourages the seen class samples to form tight clusters around an anchored class center. The detection is performed based on the distance of the test sample in the logit space to the anchors.

On the other hand, out-of-distribution (OOD) detection techniques mainly concern with detecting OOD/novel samples rather than preserving the closed-set accuracy. Some of well-known OOD detection methods are based on input preprocessing techniques and softmax temperature scaling (Liang et al., 2017; Hsu et al., 2020). In addition, various

generative methods have been proposed for OOD detection (Andrews et al., 2016; Chen et al., 2017; Perera et al., 2019). Another recent set of methods for OOD detection exploits self-supervision techniques. These methods often learn a pretext task on the in-distribution data. Then OOD detection is performed based on the generalization error of this task on test samples (Bergman & Hoshen, 2020; Hendrycks et al., 2019).

Recently, several authors distinguished *far* and *near* OOD detection problems(Winkens et al., 2020). For instance, detecting OOD CIFAR100 from in-distribution CIFAR10 is considered a near-OOD (hard) problem as there are visually similar categories in these datasets. Detecting OOD CIFAR10 from in-distribution SVHN (photographed digits) is considered a far-OOD (easy) problem because their categories are visually and semantically very different. From this perspective, most open-set recognition techniques fall in the hard-OOD detection category since the closed-set and open-set classes used for evaluation contain visually similar categories in many cases.

Our focus in this work is OSR for image classification. We propose to actively learn similarity features for seen classes as well as for automatically generated pseudo-unseen classes in training. By doing so, in addition to seen classes, our model learns explicit pseudo-unseen classes in training. Then, the open set detection at test is done based on the similarity of a given test sample to seen and pseudo-unseen classes. The main contributions of our work (OPG) are as follows:

1) We propose to detect open set samples by considering both the similarity to seen samples as well as to pseudo-unseen samples. *2)* Our pseudo-unseen sample generation is done by performing distribution shifting data augmentation on images of seen classes. Therefore, an advantage of the proposed method is that it does not require the training of an extra generative model along with the closed set classifier. *3)* In our experiments, we demonstrate the effectiveness of the proposed method on benchmark datasets.

2 BACKGROUND

It is a common practice in image classification task to randomly augment training samples. Using various types of augmentations makes the model invariant to the visual differences of data points at test. Therefore, it improves generalization of the model on test data. Apart from this, specific types of augmentations such as rotations have shown to be one of the most effective ones for self-supervised training in computer vision. Gidaris et al. (2018) uses the prediction of rotation angle for a given image as a pretext task for self-supervised representation learning. This is mainly attributed to the fact that the trained model needs to attend to the actual objects in the image in order to predict the rotation angle correctly. Following this work, Hendrycks et al. (2019) proposes to improve model uncertainty and robustness against out-of-distribution samples by training using cross-entropy loss with an auxiliary rotation prediction loss. Auto-novel (Han et al., 2020) is a technique for novel class discovery via transfer learning. It benefits from training of self-supervised RotNet (Gidaris et al., 2018) to learn general and transferable representations.

Furthermore, self-supervised and supervised contrastive learning frameworks (Chen et al., 2020; Khosla et al., 2020) owe their success to the careful composition of various augmentation techniques in generating positive and negative pairs of instances/samples for contrastive training. Particularly, Chen et al. (2020) applies random cropping, random color distortion, and random Gaussian blur on an instance in order to generate an augmented version. Then, the instance and its augmented version is treated as a positive pair in contrastive loss. The authors demonstrate that the $90°$, $180°$, and $270°$ rotations of the same image, when treated as a positive pair in contrastive loss, does not result in good representations on downstream tasks. Inspired by this observation, we hypothesize that 90x rotations of an image can simulate the distributional shift of unseen classes that the model encounters at test time. Therefore, we use pseudo-unseen data generated from rotations for feature similarity learning during training.

Our work is also related to meta classification loss (MCL), which was originally proposed by Hsu et al. (2019) to do multi-class classification based on binary labels. Therefore, it learns a closed-set classifier. We integrate a sample generation and pairing strategy to the MCL framework to be able to conduct open set recognition. As we will explain, the proposed detection score of OPG is a practical consequence of our design choice. Our experimental results shows that rotated versions of seen images can simulate the actual unseen samples which only appear at test time.

2.1 PROBLEM DEFINITION

An open set recognition (OSR) problem is composed of two sub-tasks: 1) classifying the test samples from seen classes and 2) detecting samples that belong to unseen classes. Formally, the training data is represented by $D^S = \{(x_i^S, y_i^S)|i = 1, \ldots, N\}$, where x_i^S is the training sample and $y_i^s \in \{1, ..., r\}$ is its corresponding label. The number of seen classes is r. The test data is shown by $D^{test} = D^{S \cup U} = \{(x_i, y_i)|i = 1, \ldots, M\}$, which includes M samples
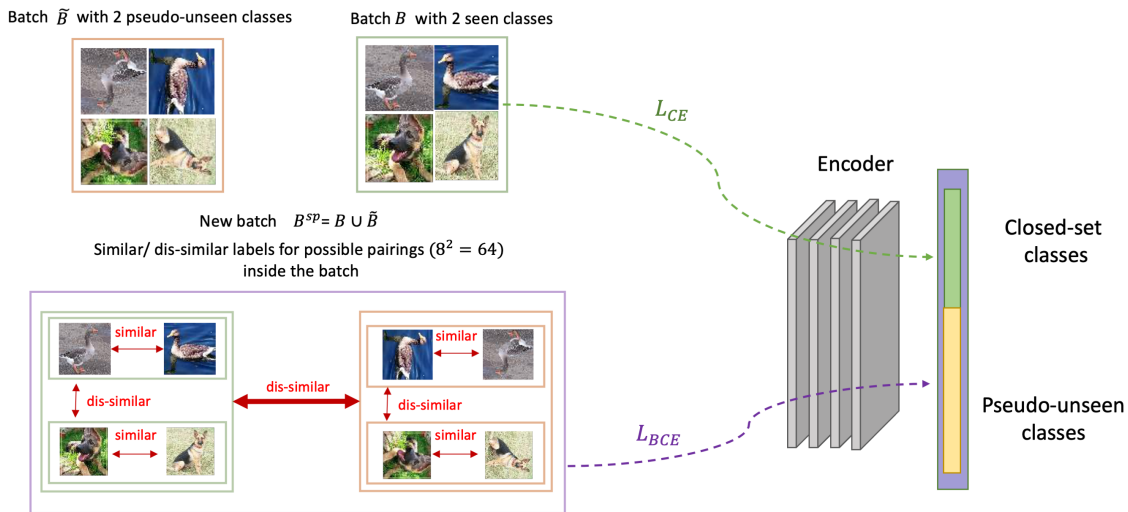
Figure 1: Illustration of pairing scheme based on the original images and their random rotations. Note that this is only a minimal example. A random batch of training samples with a sufficiently large batch size will possibly contain samples from all seen classes. The encoder in the right is trained on r closed-set classes (green) in the first step of training. In the second step of training, the fully connected layer is extended to have $r + r'$ nodes for seen (green) and pseudo-unseen (yellow) respectively. Best viewed in color.

from both seen and unseen classes($S \cap U = \emptyset$). The number of unseen classes is q which is not known to the model at training or testing.

3 PROPOSED METHOD

This section presents the proposed technique. We first motivate the use of a similarity-based binary cross-entropy loss for open set recognition. We then elaborate the generation process of pseudo-unseen samples and the pairing method that we apply. Finally, we explain how our two-step training works. In an OSR problem, the detection scores are commonly designed based on the softmax scores (conditional class probabilities). It is natural to expect the distribution of two samples belonging to the same class to be consistent with each other. Therefore, we directly optimize a binary cross entropy loss in the probability outputs of the model. Some of the generative methods for OSR define a dump/reject class to model the space of unseen. We argue that modeling the large space of unknowns in a single class is not the best choice as it does not account for similarities/differences among unseen classes. A model that can explicitly partition the unknown space is a better choice. However, this might sound infeasible as *unseen* is meant to remain hidden during training. We consider using pseudo-unseen samples instead. The pseudo-unseen data can be external unseen data or the data generated from the seen classes. The former is restrictive as the external data has to be broad enough to represent possible unseen distributions and such data may not be available. Therefore, We build our method based on the latter which is the generation of pseudo-unseen through augmentations.

3.1 PSEUDO-UNSEEN SAMPLES

Using geometric transformations, i.e. 90x rotations, has been studied in the context of self-supervised training (Gidaris et al., 2018). In addition, the experimental results of Chen et al. (2020) show that two different rotations of the same image, when treated as a positive (similar) pair in the contrastive loss framework, do not result in good representations for the downstream tasks. In our case, we are interested in generating samples dissimilar to the original ones in the feature space. We hypothesize that the augmentations of the seen class samples, if chosen properly, can be regarded as pseudo-unseen data. This pseudo-unseen data can be used to expose the closed-set classifier to the actual unseen data that will appear in testing. The rotation is thus a good candidate. We have experimented with other augmentations like Gaussian blurring, Gaussian noise and color jitter, but rotation of images by multiples of $90°$ achieves the best results.

Consider the seen class sample (x_i^S, y_i^S) in batch B of size T. We randomly rotate each x_i by either $90°$, $180°$ or $270°$. The newly generated samples are (x_i^{PU}, y_i^{PU}), where PU stands for *pseudo-unseen*. The size of the newly

generated batch $\tilde{B}$ is also T. Note that rotations do not change the object class of a sample, i.e., an upside down "duck" is still a "duck". However, we give y_i^{PU} a different label from the seen class label y_i^S since we consider it to be a *distributionally shifted* version of the original sample. Then, a new batch of $B^{SP} = B \cup \tilde{B}$ is formed. The ground-truth pairwise similarity between $(x_k, y_k) \in B^{SP}$ and $(x_l, y_l) \in B^{SP}$ is given by:

$$s_{kl} = \begin{cases} 1, & y_k = y_l \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

Figure. 1 gives an illustration of similar/dis-similar pairs. Batch B (in green square) has a total of 4 samples from 2 seen classes. Batch $\tilde{B}$ (in yellow square) is the pseudo-unseen batch generated by random rotations of images in B. The union of these two batches including a total of 8 samples is shown at the bottom. Note that labels from different groups in Figure. 1 lead to a similarity score of $s_{kl} = 0$.

3.2 TRAINING

Architecture: We use a CNN encoder architecture as in many OSR baselines. This architecture is introduced in Neal et al. (2018). It consists of 10 convolutional layers with filters of size 3×3 in each layer. Batch normalization and a leaky relu non-linearity is applied after each convolutional layer. The only difference is that we set the number of output nodes of the linear classifier layer equal to the number of seen+pseudo-unseen classes(green and yellow nodes shown in Figure. 1). We denote the number of pseudo-unseen nodes with r', which is a hyperparameter of our method that can be set to an arbitrary high value regardless of the actual number of unseen classes q. The logits output from the linear classifier layer is:

$$z = \text{Classifier}(\text{Encoder}(x)) \tag{2}$$

The length of z is $r + r'$. Therefore; the output probability distribution is over $r + r'$ classes. As we will explain in the next section, by doing so we will have a probability distribution on seens as well as pseudo-unseens. Algorithm. 1 summarizes the training and testing phases for OPG.

Total Loss The total loss to be optimized is the weighted sum of cross-entropy loss and binary-cross entropy loss:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{CE} + \lambda_2 \mathcal{L}_{BCE} \tag{3}$$

In the joint loss objective, the cross-entropy loss is only applied to the samples in seen classes. We set $\lambda_2 = 0$ until the cross-entropy loss stabilizes. We call this the *first step* of training. Then we set $\lambda_2 = 1$ and continue training on the total loss in the *second step*. We need to continue training on cross-entropy loss in the second step to avoid forgetting. In fact, the two-step training plays an important role in the convergence of the proposed method. We conduct an ablation study in the experiment section to compare the two-step training with a cold-start joint model training (without the first step). $\mathcal{L}_{CE}$ is used to learn a closed set classifier on the first r classes of the model.

$$\mathcal{L}_{CE} = -\frac{1}{N} \sum_{i=1}^{N} 1_{c=y_i} \log(p(c|x_i)) \quad \text{if} \quad x_i \in B \tag{4}$$

Similarity-based Binary Cross Entropy Loss: The aim of $\mathcal{L}_{BCE}$ is to learn clusters for seen class samples as well as for pseudo-unseen samples by pairwise similarity supervision. We adopt the meta-classification loss (Hsu et al., 2019) for our purpose which is defined as follows:

$$\mathcal{L}_{BCE} = -\frac{1}{M^2} \sum_{k=1}^{M} \sum_{l=1}^{M} [s_{kl} log(p(x_k)^T p(x_l)) + (1 - s_{kl}) log(1 - p(x_k)^T p(x_l))] \tag{5}$$

M is the number of samples in the augmented training batch $B \cup \tilde{B}$. The probability distribution is over all classes, $r + r'$, for a distribution on both seen and pseudo-unseen classes. This loss function uses the dot product of the probability distribution for samples x_k and x_l as the pairwise similarity measure. Consider the case where x_k is from a seen class and x_l is from a pseudo-unseen class. The ground-truth label s_{kl} for this pair is 0. Therefore, We expect the probability distributions $p(x_k)$ and $p(x_l)$ to be different. Since $p(x_k)$ and $p(x_l)$ are two normalized vectors, their dot product gives the cosine similarity which is minimized when $p(x_k)$ and $p(x_l)$ are perpendicular to each other. Similarly, for a positive pair x_k and x_l, the loss is minimized when two probability vectors lie in the same direction.

Note that Han et al. (2020) exploited a similar form of loss to Eq. (3). But they solve a different problem, identifying classes in the novel data via a transfer learning based clustering method. As a result, $\mathcal{L}_{BCE}$ in Han et al. (2020) learns the similarity/dissimilarity only for samples of unseen classes, but in case of OPG, $\mathcal{L}_{BCE}$ learns similarities/dissimilarities between seen-seen, unseen-unseen and seen-unseen samples.

Algorithm 1 OPG training and detection

Require: labeled seen data D^s, test data D^{test}
 1: **Training**
 2: Initialize *Encoder* model with output layer of size $(r + r')$
 3:     $\lambda_1 \leftarrow 1, \lambda_2 \leftarrow 0$
 4:     train on $\mathcal{L} = \lambda_1 \mathcal{L}_{CE}$ until convergence
 5:     $\lambda_2 \leftarrow 1$
 6: **repeat**
 7:         Get batch B of size $|N|$ from D^s.
 8:         Apply one random rotation per sample in batch B to generate batch $\tilde{B}$
 9:         $B^{SP} = B \cup \tilde{B}$
10:         calculate $\mathcal{L}_{BCE}$ for all $(x_i, x_j) \in B^{SP} \times B^{SP}$
11:         calculate $\mathcal{L}_{CE}$ only on B
12:         Backpropagate $\mathcal{L} = \lambda_1 \mathcal{L}_{CE} + \lambda_2 \mathcal{L}_{BCE}$
13: **until** convergence
14: **Testing(detection)**
15: **for** $x^{test} \in D^{test}$ **do**
16:         Calculate $S(x_i^{test})$ according to Eq. 6
17: **end for**

3.3 TESTING

Our model learns the similarity between seen and pseudo-unseen classes during training. In testing, when a sample is more similar to a pseudo-unseen class, the similarity-based features will pull it towards one of the pre-allocated pseudo-unseen classes and push it away from the seen class centers. Therefore, the accumulative probability scores over pseudo-unseen classes can naturally be inferred as the confidence of the model in detecting a test sample as unseen. As the result, the open set detection score is defined as:

$$S(x) = 1 - \sum_{0 \leq c < r} p(c|x) \tag{6}$$

where $p(c|x)$ is the softmax output for seen class c. This score is used to calculate AUROC (Area under the ROC curve for open set detection). This section evaluates the proposed method, present the experimental setting, and analyze the experimental results.

3.4 DATASETS

The difficulty level of an OSR task is measured by the openness metric defined in Scheirer et al. (2012). A task is harder when more unseen classes are presented to the model at the test time. Openness is defined as $Openness = (1 - \sqrt{\frac{2 * N_{train}}{N_{test} + N_{target}}}) * 100$ Where N_{train} is the number of seen classes, N_{target} is the number of seen classes at testing and N_{test} is the total number of seen and unseen classes at test. Following the protocol used in recent OSR techniques, we evaluate the performance of our proposed method using the following datasets.

CIFAR10.(Krizhevsky et al., 2009) The training set of seen classes is a set of 6 classes of CIFAR10. The 4 remaining classes are used as open set (unseen) classes. (Openness = 13.39%) *CIFAR+10.*(Krizhevsky et al., 2009) This dataset uses 4 non-animal classes of CIFAR10 as the closed set (seen) classes for training. 10 animal classes are randomly chosen from CIFAR100 as the open set (unseen) classes. (Openness = 33.33%) *CIFAR+50.*(Krizhevsky et al., 2009) This dataset uses 4 non-animal classes from CIFAR10 as a closed set. All 50 animal classes from CIFAR100 are used as the open set classes. So it is a harder task than CIFAR+10. (Openness = 62.86%) *SVHN.*(Netzer et al., 2011) 6 random classes from SVHN (street view house numbers) are used for closed set training while the remaining 4 classes are used as open set (unseen) classes. (Openness = 13.39%)
TinyImagenet.(Le & Yang, 2015) It is a 200-class subset of ImageNet. 20 random classes are used as the closed set (seen) classes. The remaining 180 classes are used as open set (unseen) classes. (Openness = 57.35%)

The reported scores are averaged over 5 seen-unseen split for each dataset. The class splits that we have used are publicly available in the github repository of Miller et al. (2021) [1]

---

[1]https://github.com/dimitymiller/cac-openset

3.5 BASELINES

We compare OPG with 8 OSR baselines. *DOC* (Shu et al., 2017) is a method originally proposed for open set recognition of text data. It uses one-vs-rest sigmoid function in the output layer. It compares the maximum score over sigmoid outputs to a predefined threshold to reject or accept a test sample. *OpenMax* (Bendale & Boult, 2016) is an early technique for open set recognition in deep models. It does calibration on the penultimate layer of the network to bound the open space risk. *G-OpenMax* and *OSRCI* Ge et al. (2017)(Neal et al., 2018) are both generative models that use the set of generated samples to learn an extra class. So, the model is a $K + 1$ class classifier of seen and pseudo-unseen. *C2AE* (Oza & Patel, 2019) is a class-conditioned generation method that uses the reconstruction error of unseen samples as the detection score. *CAC*(Miller et al., 2021) is a recent approach that uses anchored class centers in the logit space to encourage forming of dense clusters around each known class. Detection is done based on the distance of the test sample to these seen class centers. *GFROR* (Perera et al., 2020) combines generation-based models with self-supervision learning to solve the problem. *Generalized ODIN* (Hsu et al., 2020) Generalized ODIN (G-ODIN) is a recent out-of-distribution (OOD) detection method which uses a decomposed confidence score for OOD detection. However, the code of G-ODIN have not been released. We thus could not run their official code for comparison. We implemented the G-ODIN algorithm by closely following the algorithm in their paper, which produced the results reported in Table 1. We also applied the same hyperparameters suggested in their paper. Table 1 shows that G-ODIN does not do well for the OSR setting, which was called *semantic shift detection* in Hsu et al. (2020) and the authors explicitly stated that semantic shift detection is a more challenging problem for G-ODIN than OOD detection. ARPL (Chen et al., 2021) is a recent baseline. It proposes a classification framework with the adversarial margin constraint to reduce the open space risk. In order to be consistent with our results, we generated the reported scores in Table 1 by running ARPL official code on the same seen and unseen splits as OPG.

3.6 EVALUATION METRICS

AUROC: Area under the ROC Curve is the primary measure for open set detection performance. It is threshold-free and indicates the trade-off between true positive rate (seen samples correctly detected as seen) and the false positive rate (unseen samples incorrectly detected as seen). *Classification accuracy:* Classification accuracy is used to show the performance of our method when only seen classes (closed-set samples) are input to the model.

3.7 IMPLEMENTATION DETAILS

We use the same encoder architecture as the baselines (Neal et al., 2018; Perera et al., 2020; Oza & Patel, 2019; Miller et al., 2021) for a fair comparison. The encoder is a CNN architecture with a total of 10 convolutional layers. Batch normalization and leaky relu (0.2) are applied after each convolutional layer. For DOC (Shu et al., 2017), we conducted the experiments with the same encoder with a sigmoid layer in the output as required by DOC algorithm. Since we use 90x rotations to achieve distribution shift from the original data points, we do not apply random rotations on the original set of data points as part of the standard augmentations for classification problems. The size of the output layer of the encoder is set to $r + r'$(*seen+pseudo-unseen*). r' is a hyperparameter of OPG, which we will discuss in Section 3.8.2. We set the total number of classees to $r + r' = 100$ for CIFAR10, CIFAR+10, CIFAR+50 and SVHN experiments. For Tinyimagenet we set $r + r' = 200$. We use Stochastic Gradient Descent (SGD) to optimize the objective. At the first step of training ($\lambda_1 = 1$, $\lambda_2 = 0$) we train the model for 200 epochs with a batch size of 128 for CIFAR10, CIFAR+10, CIFAR+50 and SVHN. The initial learning rate is 0.01 and it is decayed by a factor of 10 at 100th and 150th epochs respectively. For TinyImagenet, we train the model for 500 epochs with a batch size of 128. The initial learning rate is 0.01 and it is decayed by a factor of 10 at the 300th epoch. In the second step, when we optimize both $\mathcal{L}_{CE}$ and $\mathcal{L}_{BCE}$ ($\lambda_1 = 1$, $\lambda_2 = 1$) the network is trained with the batch-size of 256 for 100 epochs and a constant learning rate of 0.01 is maintained for CIFAR10, CIFAR+10, CIFAR+50 and SVHN. For Tinyimagenet, we train for 200 epochs with a batch size of 128. The learning rate is 0.01 and it is decayed by a factor of 10 at 100th epoch.

3.8 RESULTS AND DISCUSSIONS

The experimental results of our proposed OPG and 8 baselines are summarized in Table 1. The last column gives the average score of each system or row on all datasets. We can see that on average OPG outperforms all baselines. Considering ARPL as the second best performing method, OPG performs better than ARPL on CIFAR10, CIFAR+10, CIFAR+50, and Tinyimagenet. Based on these results, we can conclude that rotations can successfully shift the distribution of seen classes in the case of natural images such as CIFAR10 and TinyImagenet and help produce effective OSR models. On the other hand, the performance on SVHN indicates that rotations of seen samples cannot simulate

Table 1: Open set detection performance in terms of AUROC (%). The results are averaged over 5 random splits of each dataset (± standard deviation). All scores except the results of DOC and G-ODIN are taken from Miller et al. (2021). The best scores are in bold. For TinyImagenet, the AUROC score of OPG is comparable to the best score. The last column gives the average value of each row or system.

	CIFAR10	CIFAR+10	CIFAR+50	SVHN	TinyImageNet	Average
OpenMax Bendale & Boult (2016)	$69.5_{\pm 4.4}$	$81.7_{\pm NR}$	$79.6_{\pm NR}$	$89.4_{\pm 1.3}$	$57.6_{\pm NR}$	75.6
G-OpenMax Ge et al. (2017)	$67.5_{\pm 4.4}$	$82.7_{\pm NR}$	$81.9_{\pm NR}$	$89.6_{\pm 1.7}$	$58.0_{\pm NR}$	75.9
DOC Shu et al. (2017)	$66.5_{\pm 6.0}$	$46.1_{\pm 1.7}$	$53.6_{\pm 0.0}$	$74.5_{\pm 3.2}$	$50.2_{\pm 0.5}$	58.2
OSRCI Neal et al. (2018)	$69.9_{\pm 3.8}$	$83.8_{\pm NR}$	$82.7_{\pm 0.0}$	$91.0_{\pm 0.1}$	$58.6_{\pm NR}$	77.2
C2AE Oza & Patel (2019)	$71.1_{\pm 0.8}$	$81.0_{\pm 0.5}$	$80.3_{\pm 0.0}$	$89.2_{\pm 1.3}$	$58.1_{\pm 1.9}$	75.9
GFROR Perera et al. (2020)	$80.7_{\pm 3.0}$	$92.8_{\pm 0.2}$	$92.6_{\pm 0.0}$	$93.5_{\pm 1.8}$	$60.8_{\pm 1.7}$	84.0
G-ODIN Hsu et al. (2020)	$72.2_{\pm 9.6}$	$51.7_{\pm 0.9}$	$89.8_{\pm 0.0}$	$84.1_{\pm 5.7}$	$57.0_{\pm 0.2}$	70.9
CAC Miller et al. (2021)	$80.1_{\pm 3.0}$	$87.7_{\pm 1.2}$	$87.0_{\pm 0.0}$	$94.1_{\pm 0.7}$	$\mathbf{76.0}_{\pm 1.5}$	84.9
ARPL Chen et al. (2021)	$83.1_{\pm 4.1}$	$91.8_{\pm 0.7}$	$92.3_{\pm 0.0}$	$\mathbf{94.2}_{\pm 0.5}$	$74.0_{\pm 3.0}$	87.0
OPG*	$82.4_{\pm 2.6}$	$89.6_{\pm 0.3}$	$90.4_{\pm 0.0}$	$81.2_{\pm 1.7}$	$73.5_{\pm 2.0}$	83.4
OPG(ours)	$\mathbf{83.1}_{\pm 5.0}$	$\mathbf{96.2}_{\pm 0.5}$	$\mathbf{96.1}_{\pm 0.0}$	$89.0_{\pm 2.8}$	$75.6_{\pm 2.6}$	$\mathbf{88.0}$

the actual unseen class samples. We suspect that the rotations might have a misleading effect in generating pseudo-unseen samples for digit datasets. For example, some images rotated by $\{90°, 180°, 270°\}$ can still be very similar to the original ones(e.g., 8, 0 or 5 still look similar to the original after a $180°$ rotation). We can conclude that such noisy pairwise labels negatively affect the open set detection performance on digit datasets. Figure 2(a) shows the
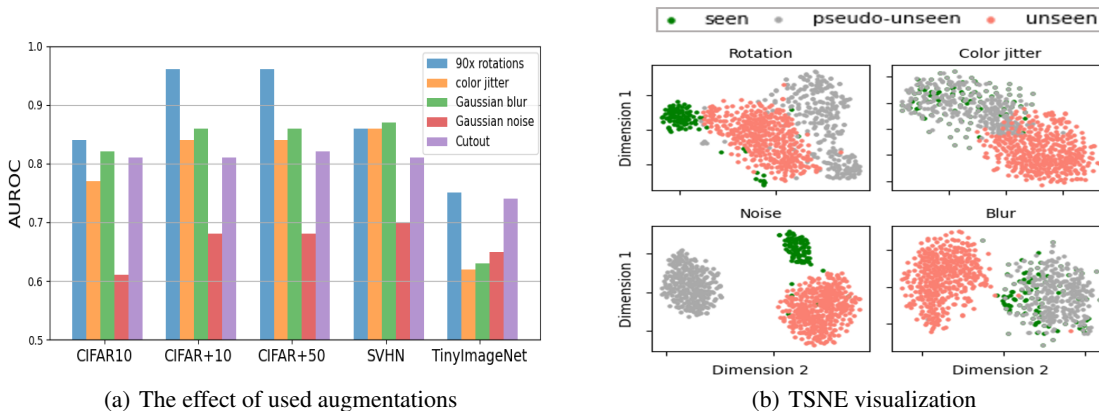

(a) The effect of used augmentations

(b) TSNE visualization

Figure 2: (a) The effect of using 5 types of augmentations applied to seen samples to generate pseudo-unseen data for CIFAR10. (b)TSNE visualization of seen, pseudo-unseen and unseen samples where pseudo-unseen is generated by seen+rotation, seen+color jitter, seen+noise and seen+blur in each subplot. (Best viewed in color)

AUROC scores when OPG is trained by using 5 different types of augmentations to generate pseudo-unseen samples. According to the results, OPG works best when 90x rotations are used for pseudo-unseen generation. We argue that using other types of augmentations including Gaussian blur, noise, color jitter or cutout can merely generate easy-to-detect pseudo-unseen data. In this case, the model can easily learn to discriminate these low-level features during training without focusing on high-level semantic features. On the other hand, 90x rotation of an image can generate hard-to-detect pseudo-unseen samples. The results summarized as OPG* in Table 1 strengthens our claim. We used random rotations between $30°$ to $60°$ to generate pseudo-unseen samples for OPG*. The detection performance of OPG* degrades compared to OPG with 90x rotation augmentations. As noted by Gidaris et al. (2018) rotating images by multiples of $90°$ do not cause easy-to-detect artifacts on the images which in our case leads to the generation of hard-to-detect pseudo-unseen samples. On the other hand, rotations in range of $30°$ to $60°$ generate easy-to-detect pseudo-unseen samples, hurting the open-set detection at inference.

It is worth noting that many contrastive training methods rely on careful composition of various augmentation types. However, their effectiveness is only shown empirically and there is little theoretical investigation of why a specific

setting or composition yields better results than others. Figure 2(b) demonstrates the differences of learned features when each of these augmentations used for the pseudo-unseen generation on CIFAR10. The top-left plot shows seen, rotated seen and unseen classes in green, gray and orange respectively. The rotated seen (pseudo-unseen) and actual unseen classes form a mixed cluster of orange and gray while the seen class forms a separate cluster. This confirms our expected behaviour which is 1) the similarity of actual unseen to pseudo-unseen and 2) the dis-similarity of unseen to seen samples in the feature space. In the other three plots, either condition 1 or 2 are not satisfied.

3.8.1 SEEN AND UNSEEN CLUSTERS IN THE FEATURE SPACE

We conducted a case study on CIFAR10 to investigate the learned feature space by OPG. Figure. 3(a) illustrates where the actual unseen classes reside in the feature space compared to the seen classes and pseudo-unseen classes (the $90°, 180°, 270°$ rotations of the same seen samples). Ideally, our technique should learn a feature space where pseudo-unseen features are a simulation of actual unseen classes. TSNE (van der Maaten & Hinton, 2008) visualization shows that in most cases, unseen classes (in orange) and pseudo-unseen classes (in gray) can be interpreted as one large cluster compared to the seen classes which form separate clusters. This confirms that the distribution shift by rotations can simulate the actual shift from seen classes to unseen classes in testing. The unseen class samples at test time are closer to the generated pseudo-unseen samples than samples that the model has seen in training. Therefore, it can successfully distinguish such samples in testing. Figure. 3(b) shows the distribution of the confidence score $S(x)$ on a seen and unseen class of CIFAR+10.

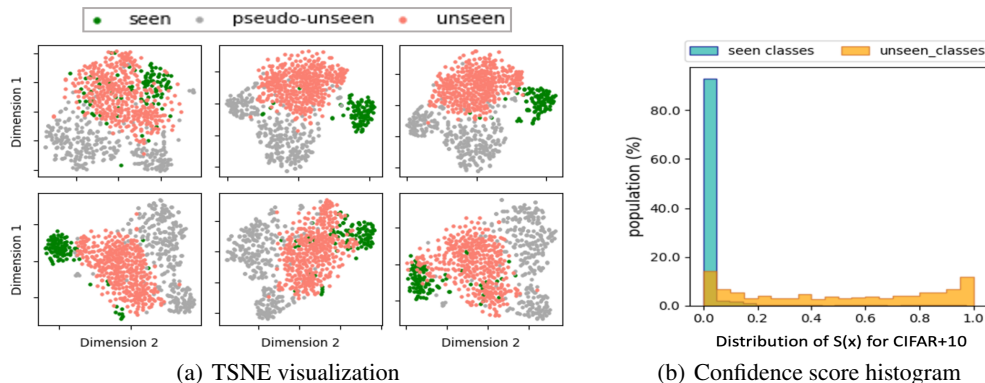(a) TSNE visualization   (b) Confidence score histogram

Figure 3: (a) Each subplot shows a TSNE visualization of a seen class from CIFAR10 in green, the features of the rotated versions of the same seen class in gray and the actual 4 unseen classes in orange. (Best viewed in color) (b) The distribution of detection score $S(x)$ over seen and unseen classes of CIFAR+10, and SVHN respectively. (Best viewed in color)

3.8.2 UNKNOWN NUMBER OF UNSEEN CLASSES

Since OPG pre-allocates the pseudo-unseen classes in training, it is natural to think that by knowing the actual number of unseen classes we can learn a better open set classifier. To study the effect of the total number of training classes $r + r'$ on OSR performance, we performed an ablation study. Table. 2 illustrates how the AUROC score is affected when the total number of classes in training is varied from the ground-truth number of seen+unseen classes to a large number of classes.
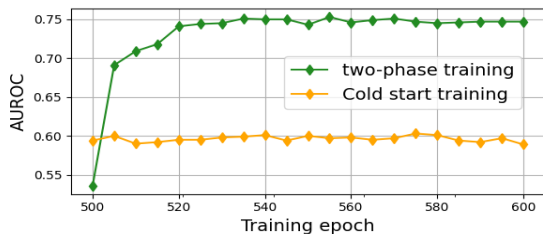
Table 2: The effect of the number of pre-allocated pseudo-unseen classes is summarized in the table. We can confirm that knowing the ground truth $r + r'$ apriori does not affect the performance of OPG.

CIFAR10 / SVHN / CIFAR+10 / CIFAR+50 / TinyImageNet	
$r + r'$	AUROC
{10/10/14/50/200}	0.83 / 0.89 / 0.96 / 0.96 / 0.75
{50/50/50/100/250}	0.82 / 0.88 / 0.96 / 0.96 / 0.76
{100/100/100/150/300}	0.83 / 0.89 / 0.96 / 0.95 / 0.75
{200/200/200/200/350}	0.83 / 0.88 / 0.95 / 0.96 / 0.74

The first row corresponds to the case $r + r' = seen + unseen$. For the other 3 experiments, $r + r' > seen + unseen$. We can confirm from the table that the number of pre-allocated $r + r'$ is not a major factor in OSR performance. Consider the experiment with the CIFAR10 dataset. The ground truth number of unseen classes is 4. In the extreme case experiment, we have set this number to $194(200 - 6)$ in training. Intuitively, this situation can be interpreted as having many empty pseudo-unseen clusters at training. Since the probability values for the extra pre-allocated classes $r\prime$ is very low, it does not affect the dot product value in $\mathcal{L}_{BCE}$; therefore, the performance remains stable.

3.8.3 TWO STEP TRAINING

Our proposed total loss is composed of two loss terms $\mathcal{L}_{BCE}$ and $\mathcal{L}_{CE}$. It might seem sufficient to optimize on the joint loss $\mathcal{L}_{CE} + \mathcal{L}_{BCE}$ together in the first place. However, $\mathcal{L}_{BCE}$ requires stable probability outputs of the network for similarity calculation. By training the encoder in a cold-start fashion, the model might converge to a local minimum. We did an ablation experiment on TinyImagenet to study the importance of the first warm-up step. Figure 4(a) illustrates the AUROC score evolution on the test data as the training reaches the final 100 epochs. It is clear from the plot that the cold-start training has difficulty in convergence. Moreover, we observed that the closed set accuracy on seen classes tends to improve after the second step of training finishes. We have tabulated the classification accuracy at the end of the first step and second step of training. According to the Table 4(b), $\mathcal{L}_{BCE}$ not only contributes to similarity learning for OSR but also improves the accuracy on almost all datasets. TinyImagenet accuracy drops by only 2%. This is intuitive as $\mathcal{L}_{BCE}$ groups samples of each seen class together far from other seen and pseudo-unseen clusters.

	SVHN	CIFAR10	CIFAR+	TinyImagenet
$\mathcal{L}_{CE}$	0.94	0.94	0.95	0.74
$\mathcal{L}_{CE} + \mathcal{L}_{BCE}$	0.96	0.95	0.96	0.72

(a) Two step training effect   (b) closed-set accuracy after each training step

Figure 4: (a) The AUROC scores on test data for the last 100 epochs of training on TinyImagenet. Note that cold-start training optimizes the total loss for 600 epochs. In two-step training, the training starts by optimizing $\mathcal{L}_{CE}$ for 500 epochs, and then the joint loss is optimized for the last 100 epochs of training. Best viewed in color. (b) Closed-set accuracy recorded at the end of first and second step of training.

4 CONCLUSION

We proposed an open set recognition method that models pseudo-unseen classes along with seen classes at training. Our technique benefits from distribution shifting data augmentations to automatically generate pseudo-unseen samples; therefore, it eliminates the need for training generative models. Our method pre-allocates pseudo-unseen classes at training. A practical consequence of this design choice is a natural detection score that is simply the summation over softmax probabilities of pseudo-unseen classes at test. We showed in our experiments that the open set detection performance is independent of the number of pseudo-unseen classes that we use at training which makes our approach flexible in real world settings. The proposed method can perform better or similar to other state of the art baselines on benchmark datasets for open set recognition.

ACKNOWLEDGMENTS

REFERENCES

Jerone TA Andrews, Edward J Morton, and Lewis D Griffin. Detecting anomalous data using auto-encoders. *International Journal of Machine Learning and Computing*, 6(1):21, 2016.

Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1563–1572, 2016.

Liron Bergman and Yedid Hoshen. Classification-based anomaly detection for general data. *arXiv preprint arXiv:2005.02359*, 2020.

Guangyao Chen, Peixi Peng, Xiangqian Wang, and Yonghong Tian. Adversarial reciprocal points learning for open set recognition. *arXiv preprint arXiv:2103.00953*, 2021.

Jinghui Chen, Saket Sathe, Charu Aggarwal, and Deepak Turaga. Outlier detection with autoencoder ensembles. In *Proceedings of the 2017 SIAM international conference on data mining*, pp. 90–98. SIAM, 2017.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

ZongYuan Ge, Sergey Demyanov, Zetao Chen, and Rahil Garnavi. Generative openmax for multi-class open set classification. *arXiv preprint arXiv:1707.07418*, 2017.

Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.

Kai Han, Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Andrea Vedaldi, and Andrew Zisserman. Automatically discovering and learning new visual categories with ranking statistics. *arXiv preprint arXiv:2002.05714*, 2020.

Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. In *Advances in Neural Information Processing Systems*, pp. 15663–15674, 2019.

Yen-Chang Hsu, Zhaoyang Lv, Joel Schlosser, Phillip Odom, and Zsolt Kira. Multi-class classification without multi-class labels. *arXiv preprint arXiv:1901.00544*, 2019.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10951–10960, 2020.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*, 2020.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv preprint arXiv:1706.02690*, 2017.

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*, pp. 3570–3578, 2021.

Lawrence Neal, Matthew Olson, Xiaoli Fern, Weng-Keen Wong, and Fuxin Li. Open set learning with counterfactual images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 613–628, 2018.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

Poojan Oza and Vishal M Patel. C2ae: Class conditioned auto-encoder for open-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2307–2316, 2019.

Pramuditha Perera, Ramesh Nallapati, and Bing Xiang. Ocgan: One-class novelty detection using gans with constrained latent representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2898–2906, 2019.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11814–11823, 2020.

W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. 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, 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, 2012.

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.

Bernhard Schölkopf, Robert C Williamson, Alexander J Smola, John Shawe-Taylor, John C Platt, et al. Support vector method for novelty detection. In *NIPS*, volume 12, pp. 582–588. Citeseer, 1999.

Lei Shu, Hu Xu, and Bing Liu. Doc: Deep open classification of text documents. *arXiv preprint arXiv:1709.08716*, 2017.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008. URL http://www.jmlr.org/papers/v9/vandermaaten08a.html.

Jim Winkens, Rudy Bunel, Abhijit Guha Roy, Robert Stanforth, Vivek Natarajan, Joseph R Ledsam, Patricia MacWilliams, Pushmeet Kohli, Alan Karthikesalingam, Simon Kohl, et al. Contrastive training for improved out-of-distribution detection. *arXiv preprint arXiv:2007.05566*, 2020.