Adversarial Reciprocal Points Learning for Open Set Recognition

Guangyao Chen, Peixi Peng, Xiangqian Wang and Yonghong Tian, *Senior Member, IEEE*

Abstract—Open set recognition (OSR), aiming to simultaneously classify the seen classes and identify the unseen classes as 'unknown', is essential for reliable machine learning. The key challenge of OSR is how to reduce the empirical classification risk on the labeled known data and the open space risk on the potential unknown data simultaneously. To handle the challenge, we formulate the open space risk problem from the perspective of multi-class integration, and model the unexploited extra-class space with a novel concept *Reciprocal Point*. Follow this, a novel learning framework, termed **A**dversarial **R**eciprocal **P**oint **L**earning (ARPL), is proposed to minimize the overlap of known distribution and unknown distributions without loss of known classification accuracy. Specifically, each reciprocal point is learned by the extra-class space with the corresponding known category, and the confrontation among multiple known categories are employed to reduce the empirical classification risk. Then, an adversarial margin constraint is proposed to reduce the open space risk by limiting the latent open space constructed by reciprocal points. To further estimate the unknown distribution from open space, an instantiated adversarial enhancement method is designed to generate diverse and confusing training samples, based on the adversarial mechanism between the reciprocal points and known classes. This can effectively enhance the model distinguishability to the unknown classes. Extensive experimental results on various benchmark datasets indicate that the proposed method is significantly superior to other existing approaches and achieves state-of-the-art performance. The code is released on github.com/iCGY96/ARPL.

Index Terms—Open Set Recognition, Out-of-Distribution Detection, Reciprocal Points, Generative Adversarial Learning

1 Introduction

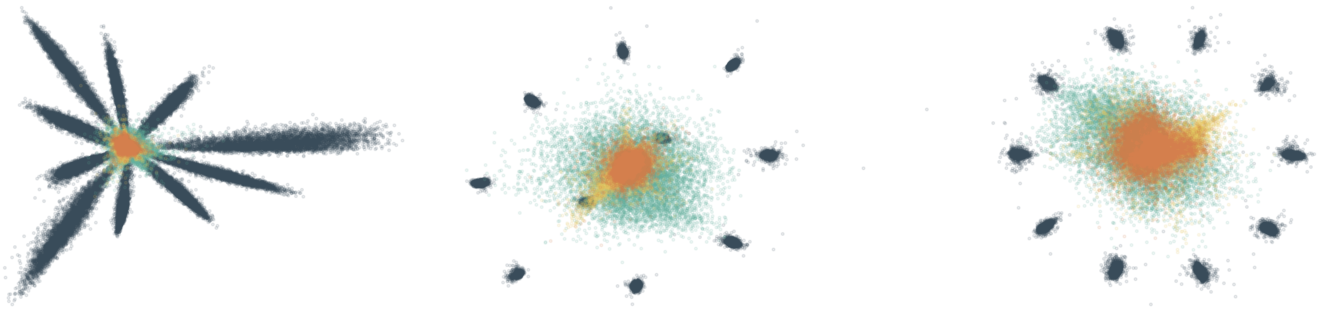
In the past few years, deep learning has achieved even surpassed human-level performances in many image recognition/classification tasks [1]. These methods follow the closed set setting which assumes that all testing classes are known or seen in the training. However, in the realistic applications, the knowledge of the classes is incomplete, and the unknown classes could be submitted to an algorithm during testing. For example, an autonomous mobile agent like a self-driving vehicle will probably encounter objects of unknown origin at some point during its lifecycle. Hence, a robust recognition system should identify test samples as known/unknown, as well as correctly classifying all test instances of the seen or known classes simultaneously, which is referred to the open set recognition (OSR) task [2].

The key of OSR is to reduce the empirical classification risk on the labeled known data and the open space risk on the potential unknown data simultaneously, where the open space risk is in labeling the open space as "positive" for any known class [2]. Follow this, a typical deep-learning-based baseline is employing a linear classification layer and the Softmax function on the embedding features to produce a probability distribution over the known classes. It typically assumes that the samples from the unknown classes should have a uniform probability distribution over the known classes. As shown in Fig. 1(a), the softmax constructs several hyperplanes to separate the embedding feature space into different sub-spaces where each sub-space corresponds to a known class. For the OSR task, the learned embedding features should be not only separable but also discriminative and generalized enough for identifying new unseen classes without label prediction. However, the softmax loss only encourages the separability of features, and cannot distinguish the known and unknown classes sufficiently. To make the features more discriminative, several methods [3], [4], [5] utilize the prototype to represent each known class in the embedding feature space and encourage the features of training samples close to the corresponding prototypes. As shown in Fig. 1(b), the learned prototypes may be converged at the space of the unknown classes in the training, and makes the known and unknown classes indistinguishable. Overall, the two types of methods both only focus on the known data and ignore the potential characteristics of the unknown data, resulting in less effectiveness on reducing the open space risk. Hence, we argue that not only the known classes, but also the unknown classes should be modeled in the training.

In order to model the unknown classes without any training samples, a novel concept *Reciprocal Point* is proposed in this article. Consider a straightforward case which only contains one known class such as *cat* in Fig. 2. *How to identify a cat?* Most classification methods aim to learn *"what is a* cat*?"*, resulting in seeing only one spot in the whole problem space. In contrast, *Reciprocal Points*, as potentially representative features of *non-cat*, identify the cat by otherness. Here a reciprocal point is typically adverse to the prototype of a known class. All these *Reciprocal Points* present an instantiated representation of the latent unexploited *extra-class* space, which can be potentially used to reduce the uncertainty when solving the problem *"what*

- *G. Chen, P. Peng,and Y. Tian are with Department of Computer Science and Technology, Peking University, Beijing, China.*
- *P. Peng and Y. Tian are also with the Pengcheng Laboratory, Shenzhen, China.*
- *X. Wang is with the Huawei, Shenzhen, China.*

Manuscript received April 19, 2020; revised August 26, 2020.



(a) Softmax (b) Prototype Learning (c) ARPL

Fig. 1: **LENET++ RESPONSES TO KNOWNS AND UNKNOWNS.** MNIST (blue) is used for known training, and KMNIST (green), SVHN (yellow) and CIFAR-100 (orange) are used for open set evaluation, where their similarity with MNIST gradually decreased. The network in (a) was only trained to classify the 10 MNIST classes using Softmax, while the networks in (b) and (c) are trained with Prototype Learning [3] and our novel Adversarial Reciprocal Prototype Learning (ARPL). This paper addresses how to improve recognition by reducing the overlap between the deep features from known samples and the features from different unknown samples. In an application, a score threshold should be chosen to optimally separate various unknown from known samples. Unfortunately, such a threshold is difficult to find for either (a) or (b). A better separation is achievable with (c).

is a cat*?"* in an OSR setting.

For the known category *cat*, most unknown samples obviously belong to the space of non-*cat* and their features should be more similar to the representation of non-*cat*, which means that the corresponding unknown information is more implicit in each non-*cat* embedding space. Therefore, a novel classification framework is proposed based on the confrontation between multiple known classes with their reciprocal points. It aims to enlarge the distance between the embedding features of the target class and the corresponding reciprocal points, as shown in Fig. 3(a). We also formulate the open set risk from the perspective of multi-class integration. To reduce the open space risk for each known class from potentially unknown data, a novel adversarial margin constraint term is proposed to limit the extra-class embedding space in a bounded range through binding the target class and its reciprocal point. Furthermore, each known class belongs to the extra-class space of other classes. When multiple classes interact with each others in the training stage, all known classes are not only pushed to the periphery of the space by the corresponding reciprocal points for classification, but also pulled in a certain bounded range by other reciprocal points with adversarial margin constraint. Finally, as shown in Fig. 3(b), all known classes are distributed around the periphery of the bounded embedding space, and the unknown samples are limited to the internal bounded space. The bounded constraint prevents the neural network from generating arbitrarily high confidence for unknown. Although only known samples are available during the training stage, the interval between known and unknown classes is separated by reciprocal points indirectly.

To estimate the unknown distribution from the open space, a novel *Instantiated Adversarial Enhancement* mechanism is proposed to generate the confusing training samples, so as to enhance the model distinguishability to the known and unknown classes. Different with the common Generative Adversarial Network (GAN) [6], the proposed method involves an additional adversarial strategy between the discriminator and classifier: On the one hand, the generated samples should deceive the discriminator to judge it be known samples; On the other hand, the classifier's responses of the generated samples are encouraged close to each reciprocal point, as illustrated in Fig. 3(c). This means the generated samples should be close to the open space of the classifier's embedding space as much as possible. Finally, the generator, discriminator and classifier are trained jointly to achieve equilibrium. More diverse and confusing samples are generated in the process to promote the classifier to filter out most samples which are significantly different from the known. In addition, an auxiliary batch normalization module and a focusing training mechanism are developed to prevent the classifier from confusing prediction due to the diverse generated samples.

Our contributions are summarized as follows:

1) The open space risk is formulated from the perspective of multi-class integration, by introducing a novel concept, *Reciprocal Point*, to model the latent open space for each known class in the feature space.
2) Based on reciprocal points with an adversarial margin constraint among multiple known categories, a classification framework is introduced to reduce the empirical classification risk and the open space risk. The rationality of the adversarial margin constraint is theoretically guaranteed by Theorem 1.
3) To estimate the unknown distribution from the open space, especially the indistinguishable part with the known categories, a novel instantiated adversarial enhancement is designed to generate more diverse confusing training samples from the confrontation between the known data and reciprocal points.

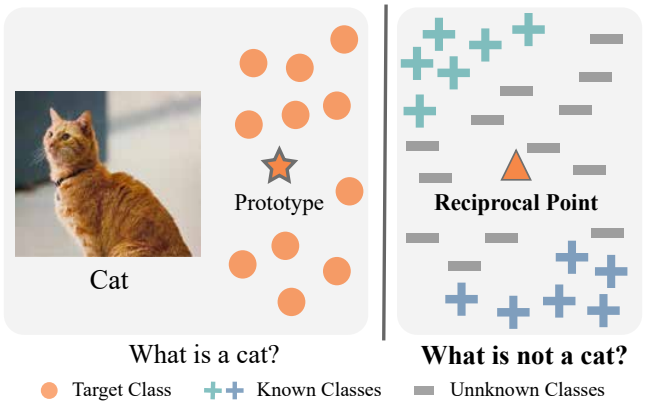


Fig. 2: How to identify a cat in the OSR setting? Most methods focus on learning the potentially representative features of cats as prototypes. In contrast, *Reciprocal Points*, as potentially representative features of *non-cat*, identify the cat by otherness. Here these *Reciprocal Points* present an instantiated representation of the *extra-class* space, which can be potentially used to reduce the uncertainty when solving the OSR problem.

This study extends our ECCV spotlight paper [7] in server aspects. 1) We develop a novel instantiated adversarial training strategy to enhance the model distinguishability to the known and unknown classes by generating confusing training samples (Section 4). The experiments are conducted on several datasets to prove the superior performance of the proposed method. 2) We improve the preliminary method by adding the cosine of the angle in the distance to measure between the known classes and their reciprocal points, which has intrinsic consistency with the classification loss (Section 3.2) and brings remarkable performance improvement. 3) The adversarial margin constraint is proposed to construct a more elastic bounded space for multi-class adversarial fusion, in order to learn more discriminative feature space to identify various unknown distributions. We present theoretical analysis to prove its rationality in (Section 3.3). 4) More qualitative and quantitative experiments are conducted to evaluate the effectiveness of the method, including: (a) A more comprehensive metric, Open Set Classification Rate [8], is developed by considering both the distinction between known and unknown and the accuracy of known classes. It is more in line with the essence of open set recognition to evaluate different algorithms in Section 5.1. (b) We add the experiments of the out-of-distribution detection task in Section 5.2, and more state-of-the-art algorithms of open set recognition for comparison. (c) More visualization illustrations and correlations are shown to better understand the embedding feature space for near-to-far unknown samples in Section 5.3.

2 Related Work

2.1 Open Set Recognition

Inspired by a classifier with rejection option [9], [10], [11], Scheirer *et al.* [2] defined the open set recognition (OSR) problem for the first time and proposed a base framework to perform training and evaluation. In recent years, OSR has been surprisingly overlooked, though it has more practical value than the common closed set setting. The few works on this topic could be broadly classified into two categories: discriminative models and generative models.

Discriminative Methods. Before the deep learning era, several OSR works utilizing traditional machine learning methods were proposed. For example, Scheirer *et al.* [12] and Jain *et al.* [13] considered a distribution of decision scores for unknown detection based on extreme value models on the Support Vector Machines (SVMs). Rudd *et al.* proposed extreme value machines [14] which modeled class-inclusion probabilities with an extreme value theory (EVT) based density function. Junior *et al.* [15] proposed an open set nearest neighbor method, which identified any test sample having low similarity as known. The similarity scores were calculated using the ratio of distances between the nearest neighbors. Zhang *et al.* [16] proposed a sparse-representation-based OSR method, which also used the EVT to identify unknown samples by residual errors. Note that these methods usually do not scale well without careful feature engineering. Recently, deep neural networks (DNNs) were also introduced to the OSR task by Bendale *et al.* [17]. They proved that the threshold on the SoftMax probability does not yield a robust model for OSR. Openmax [17] was then proposed to detect unknown classes by modeling the distance of activation vectors. Shu *et al.* [18] proposed a K-sigmoid activation-based method, which enabled the end-to-end training by eliminating outlier detectors outside the network. In these works, the sigmoid function did not have the *compact abating property* [19]. It may be activated by an infinitely distant input from all the training data, and thus its open space risk is not bounded [20].

Generative Methods. Unlike discriminative models, generator approaches generated unknown or known samples using GAN [6], Auto-Encoder [21] and Flow-based Model [22] to help the classifier to learn the decision boundary between known and unknown samples. Ge *et al.* [23] proposed G-Openmax, a direct extension of Openmax, using generative models to synthesize unknown samples to train the network. Similar to the idea in [16], Yoshihashi *et al.* [20] proposed the CROSR model, which combined the supervised learned prediction and unsupervised reconstructive latent representation to redistribute the probability distribution. [24] proposed the C2AE model for OSR, using class conditional auto-encoders to get the decision boundary from the reconstruction errors by EVT. Xin *et al.* [21] provided a conditional Gaussian distribution learning for the Variational Auto-Encoder (VAE) to detect unknowns and classify known samples by forcing different latent features to approximate different Gaussian models. Zhang *et al.* [22] offered a composed of classifier and a flow-based density estimator into a joint embedding space. However, these methods did not consider the deep distribution of unknown classes in learners, resulting in potential open space risk.

2.2 Out-of-Distribution Detection

The OSR is naturally related to some other problem settings such as out-of-distribution (OOD) detection [25], outliers detection [26], and novel detection [27], etc. Considering the safety of AI systems, the detection of OOD examples was

first introduced by Hendrycks *et al.* in [25]. Here OOD detection is the detection of samples that do not belong to the training set but could appear during testing [25]. Hendrycks *et al.* [25] demonstrated that anomalous samples had a lower maximum softmax probability than in-distribution samples. Liang *et al.* [28] proposed ODIN to allow more effective detection by using temperature scaling and adding small perturbations to the input. Lee *et al.* [29] utilized some generative models to generate the most effective samples from OOD and derived a new OOD score from this branch. Hendrycks *et al.* [30] proposed Outiler Exposure by using an auxiliary dataset to teach the network better representations for anomaly detection. OOD detection is similar to the rejection of unknown classes in OSR, because they are both studying the separation of in-distribution (known) and out-of-distribution (unkown) [2], [25] and do not require the discriminator power for known classes.

2.3 Prototype Learning

Prototype indicates an average or best exemplar of a category, thus can provide a concise representation for the entire category of instances [31]. The best-known prototype learning method is k-nearest-neighbor (KNN). In [32], the learning vector quantization (LVQ) was proposed to save the storage space and improve the computation efficiency for KNN. In most previous works, prototypes are learned through optimizing the self-defined object functions [33]. Recently, some methods also combined prototype learning with a probabilistic model and the neural network for the classification tasks. Under the framework of neural networks, prototypes are learnable representations in the form of one or more latent vectors per class. In [34], the authors represented the input instance as a K-dimensional vector, then modeled each component as a mixture of probabilities, and finally applied a probabilistic model to parameterize K-prototype patterns through the likelihood maximization. Wen *et al.* [4] proposed a center loss to learn centers for deep features of each identity and used the centers to reduce intra-class variance. Yang *et al.* [3], [5] proposed the Generalized Convolutional Prototype Learning (GCPL) with a prototype loss, which was used as a regularization to improve the intra-class compactness of the feature representation. For the OSR problem, the prototype helps to reduce intra-class distance of the known classes, but it ignored the potential characteristics of the unknown data, resulting in less effective in reducing the open space risk.

3 Adversarial Reciprocal Point Learning

3.1 Problem Definition

Given a set of n labeled samples $\mathcal{D}_{\mathcal{L}} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ with N known classes, where $y_i \in \{1, \ldots, N\}$ is the label of x_i, and a larger amount of test data $\mathcal{D}_{\mathcal{T}} = \{t_1, \ldots, t_u\}$ where the label of t_i belong to $\{1, \ldots, N\} \cup \{N+1, \ldots, N+U\}$ and U is the number of unknown classes in realistic scenarios. The deep embedding space of category k is denoted by $\mathcal{S}_k$ and its corresponding **open space** is denoted as $\mathcal{O}_k = \mathbb{R}^d - \mathcal{S}_k$, where $\mathbb{R}^d$ is the d-dimensional full space. In order to formalize and manage the open space risk effectively, $\mathcal{O}_k$ is separated as two subspaces: the positive open space from other known classes as $\mathcal{O}_k^{pos}$ and the remaining infinite unknown space as the negative open space $\mathcal{O}_k^{neg}$. That is, $\mathcal{O}_k = \mathcal{O}_k^{pos} \cup \mathcal{O}_k^{neg}$.

In our method, the samples $\mathcal{D}_{\mathcal{L}}^k \in \mathcal{S}_k$ from category k, samples $\mathcal{D}_{\mathcal{L}}^{\neq k} \in \mathcal{O}_k^{pos}$ from other known classes, and samples $\mathcal{D}_{\mathcal{U}} \in \mathcal{O}_k^{neg}$ from $\mathbb{R}^d$ but except $\mathcal{D}_{\mathcal{L}}$, are defined as the positive training data, the negative training data and the potential unknown data respectively. The binary measurable prediction function $\psi_k : \mathbb{R}^d \mapsto \{0, 1\}$ is used to map the embedding x to the label k. For 1-class OSR problem, the overall goal is to optimize a discriminative binary function ψ_k by minimizing the expected error $\mathcal{R}^k$:

$$\arg\min_{\psi_k} \{\mathcal{R}^k \mid \mathcal{R}_\epsilon(\psi_k, \mathcal{S}_k \cup \mathcal{O}_k^{pos}) + \alpha \cdot \mathcal{R}_o(\psi_k, \mathcal{O}_k^{neg})\}, \quad (1)$$

where α is a positive regularization parameter, $\mathcal{R}_\epsilon$ is the empirical classification risk on the known data, and $\mathcal{R}_o$ is the **Open Space Risk** [2] that is used to measure the uncertainty of labeling the unknown samples as the known class or as unknown. It is further formulated as a non-zero integral function on space $\mathcal{O}_k^{neg}$:

$$\mathcal{R}_o(\psi_k, \mathcal{O}_k^{neg}) = \frac{\int_{\mathcal{O}_k^{neg}} \psi_k(x)dx}{\int_{\mathcal{S}_k \cup \mathcal{O}_k} \psi_k(x)dx}. \quad (2)$$

The more the open space $\mathcal{O}_k^{neg}$ is labeled as positive , the greater the open space risk is.

In the multi-class setting, the OSR problem is identified by integrating multiple binary classification tasks (*one vs. rest*) together (as shown in Fig. 3). By summarizing the expected risk in the Eq. (1) among all known categories, i.e. $\sum_{k=1}^{N} \mathcal{R}^k$, we have

$$\sum_{k=1}^{N} \mathcal{R}_\epsilon(\psi_k, \mathcal{S}_k \cup \mathcal{O}_k^{pos}) + \alpha \cdot \sum_{k=1}^{N} \mathcal{R}_o(\psi_k, \mathcal{O}_k^{neg}). \quad (3)$$

Minimizing the left part of Eq. (3) is equivalent to training multi-binary classifiers, leading to a multi-class prediction function $f = \odot(\psi_1, \psi_2, \ldots, \psi_N)$ for N-category classification, where $\odot(\cdot)$ is an integrating operation. Hereafter, Eq. (3) is further formulated as:

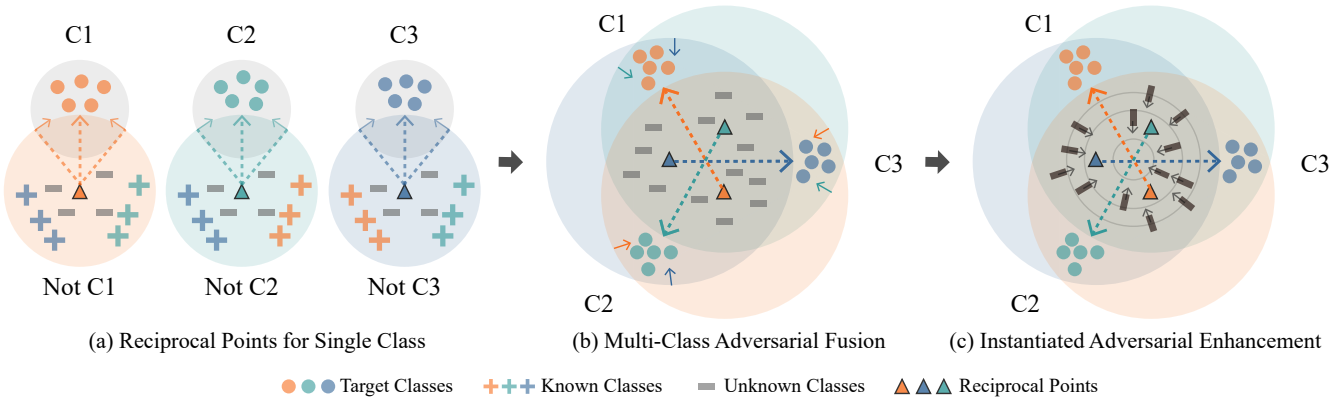
$$\arg\min_{f \in \mathcal{H}} \{\mathcal{R}_\epsilon(f, \mathcal{D}_L) + \alpha \cdot \sum\nolimits_{k=1}^{N} \mathcal{R}_o(f, \mathcal{D}_U)\} \quad (4)$$

where $f : \mathbb{R}^d \mapsto \mathbb{N}$ is a measurable multi-class recognition function, $\mathcal{D}_L$ is all labeled data during the training phase, and $\mathcal{D}_U$ is the potentially unknown data. According to Eq. (4), solving the OSR problem is equivalent to minimizing the combination of the empirical classification risk on labeled known data and the open space risk on potential unknown data simultaneously, over the space of allowable recognition functions. This makes embedding function more distinguishable between known and unknown spaces.

3.2 Reciprocal Points for Classification

The **reciprocal point** $\mathcal{P}^k$ of category k is regarded as the latent representation of the sub-dataset $\mathcal{D}_{\mathcal{L}}^{\neq k} \cup \mathcal{D}_{\mathcal{U}}$. Hence, the samples of $\mathcal{O}_k$ should be closer to the reciprocal point $\mathcal{P}^k$ than the samples of $\mathcal{S}_k$, which is formulated as:

$$\max(\zeta(\mathcal{D}_L^{\neq k} \cup \mathcal{D}_U, \mathcal{P}^k)) \leq d, \forall d \in \zeta(\mathcal{D}_L^k, \mathcal{P}^k), \quad (5)$$



(a) Reciprocal Points for Single Class (b) Multi-Class Adversarial Fusion (c) Instantiated Adversarial Enhancement

Fig. 3: An overview of the proposed **Adversarial Reciprocal Point Learning** (ARPL) approach for open set recognition. (a) *Reciprocal Points for Single Class* promote each known class far away from their reciprocal points. (b) *Multi-Class Adversarial Fusion* induces the confrontation between multi-category bounded spaces constructed by reciprocal points. As a result, the known classes are pushed to the periphery of the feature space, and the unknown classes are limited in the bounded space. (c) *Instantiated Adversarial Enhancement* generates more valid and more diverse confusing samples to promote the reliability of the classifier.

where $\zeta(\cdot,\ \cdot)$ calculates a set of distances of all samples between two sets. Based on Eq. (5), the samples could be classified by the opposition between the reciprocal points and the corresponding known classes.

Specifically, the reciprocal point of a class is represented by a m-dimensional representation, and can be optimized by an deep embedding function $\mathcal{C}$ with the learnable parameters θ. Given the sample x and reciprocal points $\mathcal{P}_k$, their distance $d(\mathcal{C}(x), \mathcal{P}^k)$ is calculated by combining the Euclidean distance d_e and dot product d_d:

$$\begin{aligned} d_e(\mathcal{C}(x), \mathcal{P}^k) &= \frac{1}{m} \cdot \|\mathcal{C}(x) - \mathcal{P}^k\|_2^2, \\ d_d(\mathcal{C}(x), \mathcal{P}^k) &= \mathcal{C}(x) \cdot \mathcal{P}^k, \\ d\ (\mathcal{C}(x), \mathcal{P}^k) &= d_e(\mathcal{C}(x), p_i^k) - d_d(\mathcal{C}(x), \mathcal{P}^k). \end{aligned} \tag{6}$$

Each known class is opposite to its reciprocal point in both the spatial position and the angle direction. The combination of the Euclidean similarity and the dot product is capable of better evaluating the similarity between the embedding features of known classes and their reciprocal points.

Based on the proposed distance metrics, our framework estimates the otherness between the embedding feature $\mathcal{C}(x)$ and the reciprocal points of all known classes to determine which category it belongs to. Following the nature of reciprocal points, the probability of sample x belonging to category k is proportional to the otherness between $\mathcal{C}(x)$ and the reciprocal point $\mathcal{P}^k$, where a greater distance between $\mathcal{C}(x)$ and $\mathcal{P}^k$ leads to assign the sample x by label k with a larger probability. According to the sum-to-one property, the final classification probability is normalized with the softmax function:

$$p(y = k|x, \mathcal{C}, \mathcal{P}) = \frac{e^{\gamma d(\mathcal{C}(x),\ \mathcal{P}^k)}}{\sum_{i=1}^{N} e^{\gamma d(\mathcal{C}(x),\ \mathcal{P}^i)}}, \tag{7}$$

where γ is a hyper-parameter to control the hardness of the probability assignment. The learning of θ is achieved by minimizing the reciprocal points classification loss based the negative log-probability of the true class k:

$$\mathcal{L}_c(x; \theta, \mathcal{P}) = -\ log\ p(y = k|x, \mathcal{C}, \mathcal{P}). \tag{8}$$

Through minimizing Eq. (8) which corresponds to $\mathcal{R}_\epsilon(f, \mathcal{D}_L)$ in the Eq. (4), the reciprocal points classification loss reduces the empirical classification risk through the reciprocal points.

Beyond classifying the known classes, an additional advantage of minimizing Eq. (8) is to separate known and unknown spaces by maximizing the distance between the reciprocal points of the category and its corresponding training samples as:

$$\arg\max_{f \in \mathcal{H}} \{\zeta(\mathcal{D}_L^k, \mathcal{P}^k)\}. \tag{9}$$

Although Eq. (8) and Eq. (9) facilitate the maximization of the interval between the closed space $\mathcal{S}_k$ and the center of open space $\mathcal{O}_k$, $\mathcal{O}_k$ is not constrained in Eq. (8). Hence, $\mathcal{S}_k$ and $\mathcal{O}_k$ may have an inestimable overlap (as shown in Fig. 8(b)), resulting in that the open space risk still exists.

3.3 Adversarial Margin Constraint

To reduce the open space risk $\mathcal{R}_o(f, \mathcal{D}_U)$ in Eq. (4), a novel *Adversarial Margin Constraint* (AMC) is proposed to constrain the open space, where each particular category k contains the positive open space $\mathcal{O}_k^{pos}$ and the infinite negative open space $\mathcal{O}_k^{neg}$. For the multi-class OSR scenarios, multiple class-wise open spaces are united into a global open space $\mathcal{O}_G$:

$$\mathcal{O}_G = \bigcap_{k=1}^{N} (\mathcal{O}_k^{pos} \cup \mathcal{O}_k^{neg}), \tag{10}$$

where the total open space risk is able to be restricted by limiting the open space risk for each known class.

To separate $\mathcal{S}_k$ and $\mathcal{O}_k$ as much as possible, the open space $\mathcal{O}_k$ must be restricted so that the open set space could be estimated. We aim to reduce the open space risk of each known class by limiting the open space $\mathcal{O}_k$ in a bounded range. It has a facilitating effect on promoting the maximum value of the distance between the negative/unknown data

Algorithm 1 The adversarial reciprocal point learning algorithm.

Input: Training data $\{x_i\}$. Initialized parameters θ in the convolutional layers, and parameters $\mathcal{P}$ and R in the loss layers, respectively. Hyperparameter λ, γ and learning rate μ. The number of iteration $t \leftarrow 0$.

Output: The parameters θ, $\mathcal{P}$ and R.

1: while *not converge* do
2: $t \leftarrow t + 1$.
3: Compute the joint loss by $\mathcal{L}^t = \mathcal{L}_c^t + \lambda \cdot \mathcal{L}_o^t$.
4: Compute the backpropagation error $\frac{\partial \mathcal{L}^t}{\partial x^t}$ for each i by $\frac{\partial \mathcal{L}^t}{\partial x^t} = \frac{\partial \mathcal{L}_c^t}{\partial x^t} + \lambda \cdot \frac{\partial \mathcal{L}_o^t}{\partial x^t}$.
5: Update the parameters $\mathcal{P}$ by $\mathcal{P}^{t+1} = \mathcal{P}^t - \mu^t \cdot \frac{\partial \mathcal{L}^t}{\partial \mathcal{P}^t} = \mathcal{P}^t - \mu^t \cdot (\frac{\partial \mathcal{L}_c^t}{\partial \mathcal{P}^t} + \lambda \cdot \frac{\partial \mathcal{L}_o^t}{\partial \mathcal{P}^t})$.
6: Update the parameters R by $R^{t+1} = R^t - \mu^t \cdot \frac{\partial \mathcal{L}^t}{\partial R^t} = R^t - \lambda \cdot \mu^t \cdot \frac{\partial \mathcal{L}_o^t}{\partial R^t}$.
7: Update the parameters θ by $\theta^{t+1} = \theta^t - \mu^t \cdot \frac{\partial \mathcal{L}^t}{\partial \theta^t} = \theta^t - \mu^t \cdot (\frac{\partial \mathcal{L}_c^t}{\partial \theta^t} + \lambda \cdot \frac{\partial \mathcal{L}_o^t}{\partial \theta^t})$.
8: end while

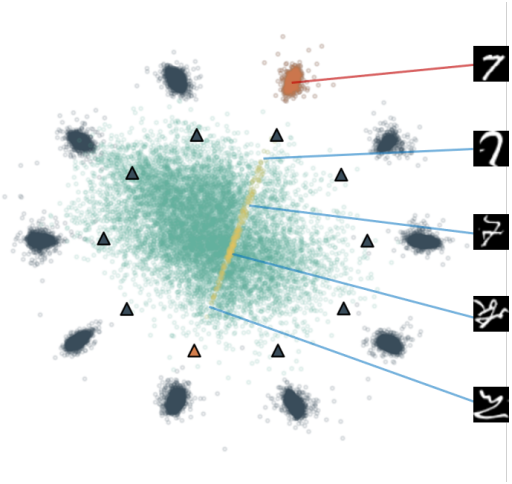


Fig. 4: Nearest neighbor retrieval examples from KMNIST for number 7. The orange circle represents the known class number 7, and the orange triangle represents the reciprocal point corresponding to number 7.

and reciprocal points less than R. Hence the following formula is established as:

$$\max(\zeta(\mathcal{D}_L^{\neq k} \cup \mathcal{D}_U,\ \mathcal{P}^k)) \ \leq \ R. \tag{11}$$

Obviously, it is almost impossible to manage the open space risk by restricting the open space, because the open space contains a large number of unknown samples $\mathcal{D}_U$. However, considering the spaces $\mathcal{S}_k$ and $\mathcal{O}_k$ are complementary to each other, the open space risk could be bounded indirectly by promoting the distance between the samples from $\mathcal{S}_k$ and the reciprocal points $\mathcal{P}^k$ to be larger than R as:

$$\mathcal{L}_o(x; \theta, \mathcal{P}^k, R^k) = max(d_e(\mathcal{C}(x), \mathcal{P}^k) - R, 0), \tag{12}$$

where R is a learnable margin. Specifically, minimizing the Eq. (12) with the classification loss $\mathcal{L}_c$ is equivalent to making $\zeta(\mathcal{D}_L^{\neq k} \cup \mathcal{D}_U,\ \mathcal{P}^k)$ in Eq. (5) smaller than R as possible. Here we use a theorem to better illustrate the advantage of our method.

Theorem 1. *For neural networks whose logit layer is based on reciprocal points, and $x \in \mathcal{D}_L^k$, $\mathcal{L}_c$ and $\mathcal{L}_o$ are minimized simultaneously if and only if* $\max(\zeta(\mathcal{D}_L^{\neq k},\ \mathcal{P}^k)) \ \leq \ R$.

Proof. Here we give a proof by contradiction. For $x \in \mathcal{D}_L^k$, if there is a sample $s \in \mathcal{D}^t$, where $t \neq k$, and $\zeta(s,\ \mathcal{P}^k) > R$. Firstly, minimizing $\mathcal{L}_c$ is to maximize the distance between each category k and its $\mathcal{P}^k$. Secondly, the loss $\mathcal{L}_o$ is minimized when $\forall k \in \{1, \ldots, N\}, \max(\zeta(\mathcal{D}_L^k,\ \mathcal{P}^k)) \leq R$ from Eq. (12). The loss $\mathcal{L}_o$ is minimized, so $\zeta(s,\ \mathcal{P}^t) \leq \ R$. Then, $\zeta(s,\ \mathcal{P}^t) < \zeta(s,\ \mathcal{P}^k)$. The sample s would be classified into category k in Eq. (7) in order to increase the loss $\mathcal{L}_c$. As a result, for $x \in \mathcal{D}_L^k$, $\mathcal{L}_c$ and $\mathcal{L}_o$ are minimized simultaneously if and only if $\max(\zeta(\mathcal{D}_L^{\neq k},\ \mathcal{P}^k)) \ \leq \ R$. □

Theorem 1 further indicates that Eq. (11) could be achieved by limiting the target known class as Eq. (12) with the classification loss $\mathcal{L}_c$. If the target class k and its reciprocal points are contained in a bounded range, the extra-class of class k (including other known classes and potential open space) is also constrained into a bounded space. With such multi-category interaction, the known categories are constrained to each other. On the one hand, the former classification loss in Eq. (9) expects to increase the distance between class k and its reciprocal point $\mathcal{P}^k$. On the other hand, the class k is bounded by the other reciprocal points $\mathcal{P}^{\neq k}$ as follows:

$$\arg\min_{f \in \mathcal{H}} \{\max(\{\zeta(\mathcal{D}_L^k, \mathcal{P}^{\neq k}) - R\} \cup \{0\})\}. \tag{13}$$

Through this adversarial mechanism between Eq. (9) and Eq. (13), each known class is pushed to the edge of finite feature space to the maximum extent, making it far away from its potential unknown space.

In addition, considering the bounded space $\mathcal{B}(\mathcal{P}^k, R)$ with the reciprocal points $\mathcal{P}^k$ as centers and R as its corresponding intervals, in order to separate the known and unknown space, we further utilize these bounded spaces to approximate the global unknown space $\mathcal{O}_G$ as much as possible. As a result, the calculation of the loss in Eq. (12) could be viewed as reducing the open space risk $\mathcal{R}_o(f, \mathcal{D}_U)$ in Eq. (4).

3.4 Learning Open Set Network

In the adversarial reciprocal points learning, the overall loss function combines Eq. (8) and Eq. (12) together to handle the empirical classification risk and the open space risk simultaneously:

$$\mathcal{L}(x, y; \theta, \mathcal{P}, R) = \mathcal{L}_c(x; \theta, \mathcal{P}) + \lambda \mathcal{L}_o(x; \theta, \mathcal{P}, R), \tag{14}$$

where λ is a hyper-parameter of controlling the weight of the adversarial open space risk module and $\theta, \mathcal{P}, R$ represent the learnable parameters. Alg. 1 summarizes the learning details of the open set network with joint supervision. Some additional explanations are also listed here.

Firstly, *Unknown Classes for the Neural Network.* Based on the principle of maximum entropy, for an unknown sample x_u without any prior information, a well-trained closed set discriminant function tends to assign the known labels to

x_u with a uniform probability. The DNNs usually embed the features of unknown samples into the space with lower magnitudes rather than random positions in the full space. This phenomenon is also consistent with the observation of [8] and the visualization results shown in Fig. 1. For real images space, *"All positive examples are alike; each negative example is negative in its own way"* [2]. On the contrary, the responses of neural networks for most unknown classes are alike. As illustrated in Fig. 4, the retrieved images are gradually different from the class center to its reciprocal point. The learned reciprocal points and the unknown classes have more similarities in the deep feature space. Actually, it could not find the specific realistic sample of the reciprocal point. Basically, the difference between a large number of unknown classes is still unknown for the classifier. Most unknown classes have great commonality for a classifier, and this part of commonness is *"unknown"*.

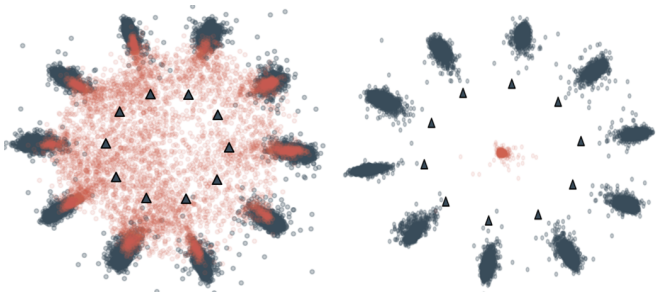
Secondly, *Unknown Classes and Reciprocal Points.* Since the global open space $\mathcal{O}_G$ is more aggregated, the open set space is able to be constrained through reciprocal points in the deep embedding space. As shown in Fig. 3 and Fig. 4, learning with Eq. (14) pushes the known spaces to the periphery of $\mathcal{O}_G$ and then separates two spaces as much as possible. As a result, an excellent embedding space structure is thus formed via adversarial reciprocal point learning (ARPL), which can be used to further divide the known classes and most unknown classes.

4 Instantiated Adversarial Enhancement

As shown in Fig. 5(a), the ARPL classifier is able to distinguish the unknown distribution without any prior knowledge of the unknown data, but still is vulnerable to the confusing samples generated from a simple generator, whatever these samples are even quite different from the known categories. In order to further reduce the open space risk caused by such unknown data, a good solution requires to minimize the open space from the learned neural networks, by the support of a reasonable optimization strategy for unknown data. However, it is a haystack to find valid unknown samples in the real scene. Therefore, we further generate the *Confusing Samples* (CS) as unknown data $\mathcal{D}_U$ to improve the discriminability of the classifier for various novelty distribution.

4.1 Learning the Confused Generator

Here, a new training strategy is proposed to learn a confused generator. Unlike the common GAN [6], we want to employ the generator to recover some confusing samples from $\mathcal{O}_G$ instead of known samples from $\mathcal{S}_k$. As shown in Fig. 6, the proposed instantiated adversarial enhancement framework contains three main components: the discriminator D, the generator G, and the classifier C with an deep embedding function $\mathcal{C}$. The classifier by ARPL represents a probability of the sample belongs to each known category. The generator maps a latent variable z from a prior distribution $P_{pri}(z)$ to the generated outputs $G(z)$, and the discriminator $D : \mathcal{X} \to [0, 1]$ represents a probability of whether the sample x is from real distribution or fake distribution. Then, given $\{z_1, \cdots, z_n\}$ from the prior $P_{pri}(z)$



(a) ARPL (b) ARPL+CS

Fig. 5: The visualization of the feature responses of the neural network to the samples from the confused generator. (a) is trained only by ARPL. (b) is trained by ARPL+CS. The red points represent the embedding of the confusing samples generated through instantiated adversarial enhancement.

and the known samples $\{x_1, \cdots, x_n\}$, the discriminator is optimized to judge the real and generated samples:

$$\max_D \frac{1}{n}\sum_{i=1}^{n}[\log D(x_i) + \log(1 - D(G(z_i)))]. \tag{15}$$

In contrast, the generator expects the generated samples closer to the known classes so as to deceive the discriminator:

$$\max_G \frac{1}{n}\sum_{i=1}^{n}[\log D(G(z_i))]. \tag{16}$$

In order to confuse the generator, an adversarial mechanism between the known classes and reciprocal points is introduced here. It promotes the generator to create samples close to each center $\mathcal{P}^k$ of the open space $\mathcal{O}_k$. As similar to Eq. (10), it is equivalent to promoting the generated images close to the global open space $\mathcal{O}_G$. Formally, the generator is optimized through the classifier:

$$\max_G \frac{1}{n}\sum_{i=1}^{n}[-\frac{1}{N}\sum_{k=1}^{N} S(z_i, \mathcal{P}^k) \cdot log(S(z_i, \mathcal{P}^k))], \tag{17}$$

where $S(z_i, \mathcal{P}^k) = softmax(d_e(\mathcal{C}(G(z_i)), \mathcal{P}^k))$. The maximum of Eq. (17) for confusing samples is achieved when the embedding of these samples are close to all reciprocal points. A theorem is introduced to better illustrate it.

Lemma 1. *For networks whose logit layer is based on reciprocal points and $x = G(z)$, the Eq. (17) is maximized when the distances between the deep feature vector $\mathcal{C}(x)$ and all reciprocal points are equal: $\forall n \in N : S(z_i, \mathcal{P}^k) = \frac{1}{N}$ and the entropy of distance distribution is maximized.*

For $x = G(z)$, the Eq. (17) is the same in form to information entropy over the per-class softmax scores. Thues, based on Shannon [35], it is intuitive that the Eq. (17) is maximized when all values are equal.

By combining these two mechanisms for confrontation, the generator is optimized by:

$$\max_G \frac{1}{n}\sum_{i=1}^{n}[\log D(G(z_i)) + \beta \cdot H(z_i, \mathcal{P})], \tag{18}$$

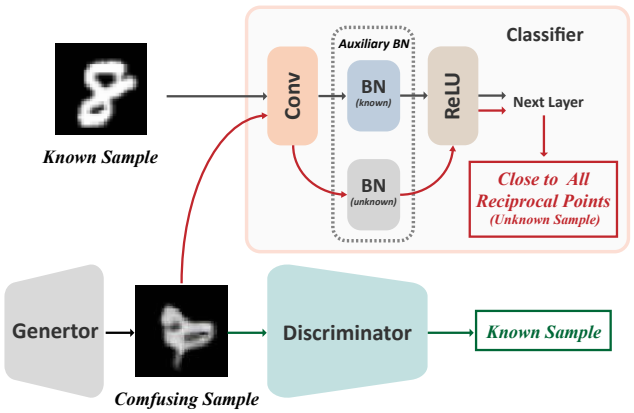


Fig. 6: The basic framework of training the confused generator based on adversarial learning between known and unknown. On the one hand, the generated images let the discriminator judge positive samples (i.e., close to known samples). On the other hand, the generated images should be unknown samples to the classifier, so the embedding feature of the neural network to these samples is close to all reciprocal points.

where β is a hyper-parameter of controlling the weight of the information entropy loss, and $H(z_i, \mathcal{P}) = -\frac{1}{N}\sum_{k=1}^{N} S(z_i, \mathcal{P}^k) \cdot log(S(z_i, \mathcal{P}^k))$ is the information entropy function. As the framework illustrated in Fig. 6, we would like to generate samples more similar with known samples; meanwhile, it also forces the generator to create samples that balance the distance for all the reciprocal points, so as to be close to the global open space $\mathcal{O}_G$. If the generated samples are far away from the boundary of the known, the loss in Eq. (16) should be large. In order to deceive the discriminator, the generator generates samples similar to the known classes, which also makes the features of the generated samples close to known classes and far away with some reciprocal points. Hence the loss in Eq. (17) should be large. So, one expects that the proposed loss encourages the generator to produce the samples which are on the boundary of the global open space, as shown in Fig. 5(a) and Fig. 7.

4.2 Reliability Enhancement

Consider the generated samples as unknown data $\mathcal{D}_U$, and the ultimate goal is to train a better feature space, where the open space is minimized. Therefore, the classifier C is optimized by the generated confusing samples as:

$$\min_{C} \frac{1}{n}\sum_{i=1}^{n}[\mathcal{L}(x_i, y_i) - \beta \cdot H(z_i, \mathcal{P})], \tag{19}$$

where $\mathcal{L}$ is the overall loss of ARPL. These generated samples are used to estimate the unknown distribution of $\mathcal{O}_G$, in order to reduce the open space risk by reducing the size of $\mathcal{O}_G$ (as shown in Fig. 5(b)).

Note that the known samples and generated samples are processed independently in Eq. (19). In this circumstance, the generated samples could confuse the classifier because of their different distributions with known samples,

Fig. 7: Generated confusing images from adversarial training with the ARPL classifier on MNIST. The leftmost and rightmost correspond to the known training images. Color depth represents the similarity to the corresponding category.

resulting in inaccurate statistics. To disentangle this mixture distribution into two underlying distributions respectively for the known and confusing samples, we hereby propose an *Auxiliary Batch Normalization* (ABN) to guarantee that its normalization statistics are exclusively performed on the confusing examples. Specifically, Batch Normalization [36] normalizes the input features by the mean and variance computed within each mini-batch, where the input features should come from a single or similar distribution [37]. As illustrated in Fig. 6, ABN helps to disentangle the mixed distributions by keeping the separate BNs to features that belong to different domains. Compared with the two-component mixture distribution (known and confusing samples), this auxiliary BN is able to effectively prevent the negative impact of confusing samples on known class discrimination. Ablation studies in Sec. 5.2 demonstrate that such disentangled learning with multiple BNs could improve the performance.

Finally, the discriminator and the classifier can be used for improving each other through the confused generator. This naturally suggests a joint training scheme where the classifier improves the generator and vice versa. In the same way, it is inevitably valid for the discriminator. An alternating algorithm is designed to optimize the above objective efficiently, as shown in Alg. 2. After training each classifier with the confusing samples (step 6 in Alg. 2), we add *Focus Training* (FT) and use the known class to train the classifier again. The purpose of this scheme is to promote the classifier to focus on the known classification and correct the deviation from too much attention to the confusing samples.

Compared with [23], [29], [38], [39], the main differences of the proposed instantiated adversarial enhancement are as follows: Firstly, the proposed method uses the adversarial mechanism between the close space $\mathcal{S}_k$ and the global open space $\mathcal{O}_G$ formed by reciprocal points. Secondly, the generated confusing samples and known samples are validated as two different distributions to accurate statistics estimation for known and unknown classes. In addition, the images generated by our method cover the whole low response of unknown feature space (as shown in Fig. 5), and also contain certain quantity of confusing images similar to the known classes (as illustrated in Fig. 7).

4.3 Unknown Classes Detection

Based on Eq. (5), the unknown samples naturally have a closer distance with all reciprocal points than the samples

Algorithm 2 The instantiated adversarial enhancement algorithm.

Input: Training data $\{x_i\}$. Initialized parameters θ_D of the discriminator D, θ_G of the confusing generator G and θ_C of the classifier C with $\mathcal{P}$ and R in loss layers, respectively. Hyperparameter λ, γ, β.

Output: The parameters θ_D, θ_G, θ_C, $\mathcal{P}$ and R.

1: **repeat**

2: Sample $\{z_1, ..., z_N\}$ from prior $P_{pri}(z)$ and known samples $\{(x_1, y_1), ..., (x_N, y_N)\}$.

3: Update the discriminator parameters θ_D by ascending its stochastic gradient:

$$\nabla_{\theta_D} \frac{1}{n} \sum_{i=1}^{n} [\log D(x_i) + \log(1 - D(G(z_i)))].$$

4: Update the generator parameters θ_G by ascending its stochastic gradient:

$$\nabla_{\theta_G} \frac{1}{N} \sum_{i=1}^{N} [\log D(G(z_i)) + \beta \cdot H(z_i \mathcal{P})].$$

5: Update the classifier parameters θ_C with $\mathcal{P}$ and R by by descending its stochastic gradient:

$$\nabla_{\theta_C} \frac{1}{n} \sum_{i=1}^{n} [L(x_i, y_i) - \beta \cdot H(z_i, \mathcal{P})].$$

6: Update the classifier parameters θ_C with $\mathcal{P}$ and R by minimizing $\frac{1}{n} \sum_{i=1}^{n} L(x_i, y_i)$.

7: **until** *convergence*

of known classes. Therefore, the probability that the test instance x belongs to one of the known classes is proportional to the distance between x and the farthest reciprocal point corresponding to category k:

$$p(known|x) \propto \max_{k \in \{1,...,N\}} d(f(x), \mathcal{P}^k). \tag{20}$$

One of key issues in the OSR models is *what's a good score for open set recognition?* (i.e., identifying a class as known or unknown). Thus, the difference between known and unknown probability belonging to any known classes is used to measure the learned models' ability to detect unknown, which provides a calibration-free measure of detection performance.

5 Experiments

5.1 Experiments for Open Set Recognition

Datasets. Similar to [24], a simple summary of these protocols for each dataset is provided:

- **MNIST**, **SVHN**, **CIFAR10**. For MNIST [40], SVHN [41] and CIFAR10 [42], by randomly sampling 6 known classes and 4 unknown classes.
- **CIFAR+10**, **CIFAR+50**. For CIFAR+N experiments, 4 classes are sampled from CIFAR10 for training. N non-overlapping classes are used as unknown, which are sampled from the CIFAR100 dataset [42].
- **TinyImageNet**. For experiments with TinyImageNet [43], 20 known classes and 180 unknown classes are randomly sampled for evaluation.

Evaluation Metrics. Since how rare or common samples from unknown space are not known in the actual scenario, the approaches to the OSR which require an arbitrary threshold or sensitivity for comparison are unreasonable [39]. A threshold-independent metric, the Area Under the Receiver Operating Characteristic (AUROC) curve [39], is considered as one of the evaluation metrics. The AUROC curve is threshold-independent metric [44] by plotting the true positive rate against the false positive rate by varying a threshold [29]. It could be interpreted as the probability that a positive example is assigned a higher detection score than a negative example [45].

However, AUROC only evaluates the distinction between known and unknown, and does not consider the accuracy of known classes in open set recognition, which has been however hidden by this gold-standard "fair" metric. In order to adapt to the setting of open set recognition, we introduce *Open Set Classification Rate* (OSCR) [8] as a new evaluation metric. Let δ is a score threshold. The *Correct Classification Rate* (CCR) is the fraction of the samples where the correct class k has maximum probability and has a probability greater than δ:

$$CCR(\delta) = \frac{|\{x \in \mathcal{D}_{\mathcal{T}}^k \wedge argmax_k P(k|x) = \hat{k} \wedge P(\hat{k}|x) \geq \delta\}}{|\mathcal{D}_{\mathcal{T}}^k|}. \tag{21}$$

The *False Positive Rate* (FPR) is the fraction of samples from unknown data $\mathcal{D}_U$ that are classified as *any* known class k with a probability greater than δ:

$$FPR(\delta) = \frac{|\{x | x \in \mathcal{D}_U \wedge max_k P(k|x) \geq \delta\}|}{|\mathcal{D}_U|}. \tag{22}$$

A larger value of OSCR indicate a better detection performance. Following the protocol in [39], the AUROC and the OSCR are averaged over five randomized trials.

Network Architecture. The classifier for this experiment is same to the neural network used in [39]. Apart from the Adam optimizer [46] used in TinyImageNet, all classifier are trained with the momentum stochastic gradient descent (Momentum SGD) optimizer [47]. The learning rate of the classifier starts from 0.1 and is dropped by a factor of 0.1 every 30 epochs in the training progress. The confused generator and the discriminator are the same with [29], and trained by the Adam optimizer [46] with the learning rate as 0.0002. More details in the Section 6.

Result Comparisons. As shown in Table 1, ARPL using only known training samples outperforms most other approaches (including traditional discriminative methods based neural networks and some complicated generative methods [20], [24], [39] for OSR) significantly. These generative methods [20], [24], [39] consider using decoder to optimize the deep feature space, but they do not pay attention to the characteristics of unknown distribution in deep feature space. Instead, ARPL pushes the known classes away from the unknown classes through reciprocal points to form a better discriminative feature space. Furthermore, ARPL with Confusing Samples (ARPL+CS) performs significantly better than other recent state-of-the-art generative

TABLE 1: The AUROC results of on detecting known and unknown samples. Results are averaged among five randomized trials.

Method	MNIST	SVHN	CIFAR10	CIFAR+10	CIFAR+50	TinyImageNet
Softmax	97.8	88.6	67.7	81.6	80.5	57.7
Openmax [17]	98.1	89.4	69.5	81.7	79.6	57.6
G-OpenMax [23]	98.4	89.6	67.5	82.7	81.9	58.0
OSRCI [39]	98.8	91.0	69.9	83.8	82.7	58.6
C2AE [24]	98.9	92.2	89.5	95.5	93.7	74.8
CROSR [20]	99.1	89.9	88.3	91.2	90.5	58.9
CGDL [21]	99.4	93.5	90.3	95.9	95.0	76.2
RPL [7]	99.3	95.1	86.1	85.6	85.0	70.2
ARPL	99.6	96.3	90.1	96.5	94.3	76.2
ARPL+CS	**99.7**	**96.7**	**91.0**	**97.1**	**95.1**	**78.2**

TABLE 2: The open set classification rate (OSCR) curve results of open set recognition. Results are averaged among five randomized trials.

Method	MNIST	SVHN	CIFAR10	CIFAR+10	CIFAR+50	TinyImageNet
Cross Entropy	99.2	92.8	83.8	90.9	88.5	60.8
GCPL [3]	99.1	93.4	84.3	91.0	88.3	59.3
RPL [7]	99.4	93.6	85.2	91.8	89.6	53.2
ARPL	99.4	94.0	86.6	93.5	91.6	62.3
ARPL+CS	**99.5**	**94.3**	**87.9**	**94.7**	**92.9**	**65.9**

methods [20], [21], [24], [39] and ARPL, especially on SVHN, CIFAR, and TinyImageNet. This further demonstrates the superiority of the proposed method, and these confusing samples could effectively improve the reliability of the neural network by ARPL.

Moreover, we redesign a new OSR experiment for a more reasonable comparison. Firstly, we abandon the baseline with the hing loss in [39], which could lead to some optimization difficulties. The more robust cross entropy loss is used as a new baseline in this experiment. Secondly, we introduce a new evaluation metric, OSCR [8], to comprehensively evaluate the performance of classification for known classes and unknown classes detection under different thresholds. Finally, under the same five known and unknown splits, we report the performance of the average of five trials.

Compared with the experiment based on the AUROC in Table 1, most tasks in this new OSR experiment become more difficult because these methods should balance the unknown detection with the classification for known classes. We compared four discriminative methods based on the neural network as shown in Table 2. The ARPL shows excellent performances than cross entropy loss, GCPL [3], and RPL [7]. Moreover, assisted confusing samples, the OSCR of ARPL has been greatly improved. Especially on TinyImageNet, the performance is improved 3.6% compared with ARPL. These results validate that ARPL and ARPL+CS are able to effectively improve the detection ability of unknown samples under the premise of ensuring the accuracy of known class classification.

5.2 Experiments for Out-of-distribution Detection

Datasets. we adopt three image datasets which represent the most challenging pairs of common OOD detection benchmarks [25], including CIFAR10, CIFAR100 [42], SVHN [41] to evaluation. CIFAR100 and SVHN are the near OOD dataset and far OOD dataset for CIFAR10, respectively. Note that the CIFAR10 and CIFAR100 classes are mutually exclusive.

Evaluation Metrics. Referring to the evaluation index in [25], [28], [29], [30], AUROC, the true negative rate (TNR) at 95% true positive rate (TPR), the area under the precision-recall curve (AUPR), and the detection accuracyare are adopted for evaluation:

- **True negative rate (TNR) at 95% true positive rate (TPR).** let TP, TN, FP, and FN denote true positive, true negative, false positive and false negative, respectively. We measure $TNR = TN/(TP + TN)$, when $TPR = TP/(FP + FN)$ is 95%
- **Area under the precision-recall curve (AUPR).** The PR curve is graph plotting the precision $= TP/(TP + FP)$ against recall $= TP/(TP + FN)$ by varying a threshold. **AUIN** (or **AUOUT**) is AUPR where in- (or out-of-) distribution samples are specified as positive.
- **Detection accuracy (DTACC).** This metric corresponds to the maximum classification probability over all possible thresholds δ. We assume that both positive and negative examples have equal probability of appearing in the test set, i.e., $P(x \in P_{in}) = P(x \in P_{out}) = 0.5$

Network Architecture. We demonstrate the effectiveness of the proposed method using ResNet with 34 layers [48] on

TABLE 3: Distinguishing in- and out-of-distribution test set data for image classification under various validation setups. All values are percentages and the best results are indicated in bold.

Method	In: **CIFAR10** / Out: **CIFAR100**					In: **CIFAR10** / Out: **SVHN**				
	TNR	AUROC	DTACC	AUIN	AUOUT	TNR	AUROC	DTACC	AUIN	AUOUT
Cross Entropy	31.9	86.3	79.8	88.4	82.5	32.1	90.6	86.4	88.3	93.6
GCPL [3]	35.7	86.4	80.2	86.6	84.1	41.4	91.3	86.1	86.6	94.8
RPL [7]	32.6	87.1	80.6	88.8	83.8	41.9	92.0	87.1	89.6	95.1
ARPL	**47.0**	**89.7**	**82.6**	**90.5**	**87.8**	**53.8**	**93.2**	**87.2**	**90.3**	**95.8**
JCL [29]	35.8	87.8	81.2	89.1	88.6	34.7	92.2	88.5	90.6	92.9
ARPL+CS (w/o ABN)	46.3	88.8	81.7	89.1	87.1	44.7	90.9	84.1	86.2	94.6
ARPL+CS (w/o FT)	46.7	89.7	82.4	90.7	87.8	71.0	95.6	91.1	94.0	96.8
ARPL+CS	**48.5**	**90.3**	**83.4**	**91.1**	**88.4**	**79.1**	**96.6**	**91.6**	**94.8**	**98.0**

various vision datasets. All classifiers are trained for 100 epochs with batch size 128 with the Adam optimizer [46]. The learning rate of the classifier starts from 0.1 and is dropped by a factor of 0.1 every 30 epochs in the training progress. The confused generator and the discriminator are the same with [29], and trained by the Adam optimizer [46] with the learning rate as 0.0002. More details in the Section 6.

Result Comparisons. As shown in Table 3, the ARPL outperforms three method training only with in-distribution data: cross-entropy loss (Baseline), GCPL [3] and RPL [7]. For CIFAR100 containing both test samples that are near as well as far OOD, ARPL is more than 3% better than the RPL in AUROC. This validates that ARPL could differentiate OOD classes that are near or far away from in-distribution data. Meanwhile, our methods are compared with a generative OOD model, named Joint Confidence Loss (JCL) [29], which adopts a similar mechanism with our instantiated adversarial enhancement. Due to considering the difference between the unknown and the known in deep feature space, the performance of ARPL without the auxiliary training of confusing samples even is better than JCL.

Here we analyze the role of ABN and FN for instantiated adversarial enhancement. The performance of ARPL+CS w/o ABN is even much lower than ARPL. This is consistent with our assumption that confusing samples and known images have different underlying distributions. Although the generator aims to generate images that are consistent with the distribution of known classes, the distribution of generated images and known classes are gradually separated after adding our adversarial mechanism. One BN for mixed distributions would influence the performance to detect OOD samples. After adding ABN (ARPL+CS w/o FT), the performance is gradually improved, especially for far OOD detection.

However, the detection of near OOD is not improved compared with ARPL. It could be that confusing samples affect the discriminative feature of the known class in the training stage. Based on the prior initialization of confusing samples, we further use FN to promote the classifier pay more attention to the classification of known classes. The performance of both near and far OOD has been further improved as shown in Table 3.

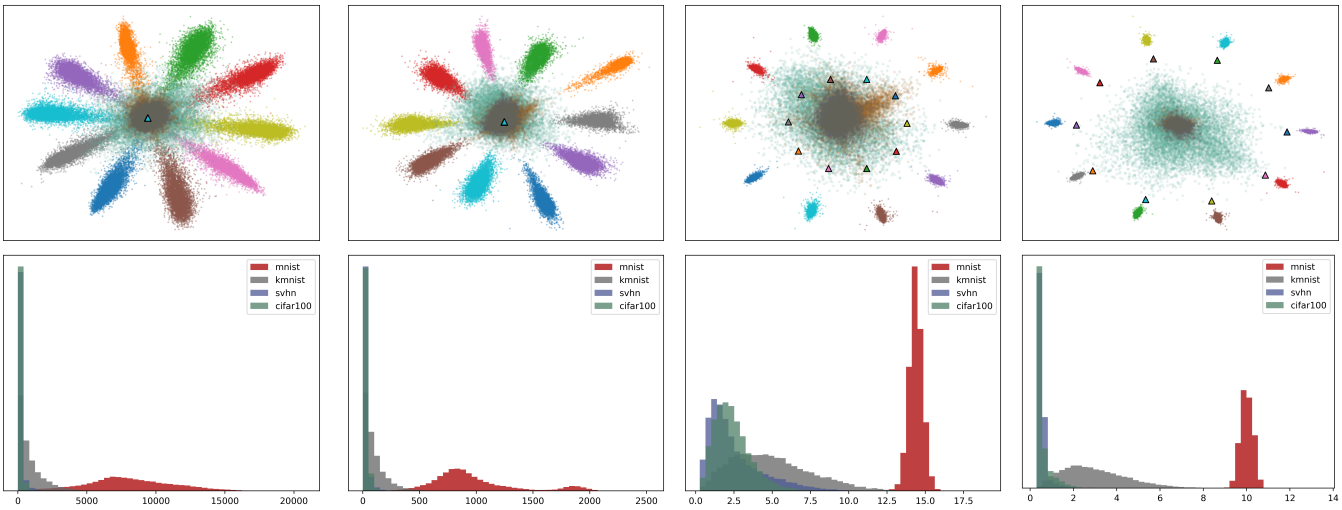
5.3 Further Analysis

5.3.1 *ARPL vs. Softmax.*

The classification loss term (the first part of Eq. (14)) defines classification principle of reciprocal points. Similar to softmax, the learned representation is still linear separable only with this classification loss. See Fig. 1(a) and 8(b), reciprocal points without $\mathcal{L}_o$ are learned to the origin, there is a significant overlap between the known and the unknown classes in low response part of entire feature space. Meanwhile, observing OSCR curve in Fig. 10, CCR is lower when FPR is lower for Softmax. This is the reason why the neural network learned with softmax detects unknown samples as some known classes with high confidence. In contrast, ARPL with $\mathcal{L}_o$ to conrtaint open space achieves better distribution (shown in Fig. 8(c)), where the whole feature space is contained in a limited range (1-13.5 in abscissa for unknowns and 14-17 in abscissa for knowns in Fig. 8(c)) to prevent high confidence for unknown classes. APRL with $\mathcal{L}_o$ can guarantee high accuracy even at the low FPR in Fig. 10, because of effective restrictions on open space by $\mathcal{L}_o$ and pushing known classes far away from global open space.

5.3.2 *ARPL vs. GCPL.*

As shown in Fig. 1(b), GCPL used the prototypes to reduce intra-class variance. However, without considering the unknown, GCPL extends unknown classes to the whole feature space, resulting in a significant overlap with known classes. In the initial stage of neural network training, the prototypes of GCPL are easily distributed in the unknown feature space. This also leads to some known categories are distributed in the lower response part of the feature space, so as to bring more open space risk. As shown in Fig. 10, GCPL achieves worse AUROC and OSCR performance than Softmax. In contrast, ARPL is not affected by initialization because each known class is far away from its corresponding unknown representation, reciprocal points. Under the interaction of the classification loss and the $\mathcal{L}_o$, different known categories are spread to the periphery of the space, while unknown categories are restricted to the interior. It can be observed that a clear gap is maintained between the two types of samples (known vs. unknown) in Fig. 8(c). ARPL improves the robustness of neural networks by preventing the misjudgment of the unknown class through the bounded restriction, thereby enhancing and stabilizing the classification of known categories.



(a) RPL($\lambda = 0$) (b) ARPL($\lambda = 0$) (c) ARPL($\lambda = 0.1$) (d) ARPL+CS

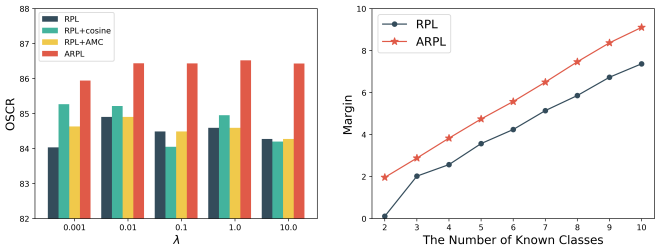
Fig. 8: (1) The first row is visualization in the learned feature space of MNIST as known and KMNIST, SVHN, CIFAR100 as unknown. Color shapes in the middle are data of unknown, and circles in color are data of known samples. Different colors represent different classes. Colored triangles represent the reciprocal points learned of different known categories. (2) The second row is the maximum distance distribution between features and reciprocal points.

5.3.3 ARPL vs. RPL.

As shown in Table 1, Table 2 and Table 3, ARPL has a great improvement than RPL [7]. Compared with RPL, ARPL improves similarity estimation and reduceing open space risk by a elastic bounded space. Firstly, for the distance between feature and reciprocal points in Eq. (6), the angle metric is added. Each known class is opposite to its reciprocal points in spatial position and angle direction. Compared Fig. 8(a) and Fig. 8(b), there are is a larger space between features of each class in ARPL(λ=0), and each class is more compact. Adding angle metric effectively reduces the intra-class distance to better performance with different λ as shown in Fig. 9(a). Secondly, we do not limits the distance between all known classes and corresponding reciprocal points to the same margin anymore, and use an adaptive regularization in Eq. (12). The stronger limitation in [7] will reduce the network's discriminability for all known classes. The performance of classification and unknown detection will be reduced if this restriction is too large. It also is sensitive to the setting of hyper-parameter λ in [7]. Compared with ARPL and RPL+AMC in Fig. 9(a), the performance of RPL and RPL+cosine are more affected by λ and their performances are more unstable. For the AMC, the learnable R is used as the anchor. By constantly adjusting the reciprocal points and the deep feature, all known classes are promoted to less than R, so as to focus on samples that are difficult to distinguish among unknown adaptively. Through this adversarial margin constraint in Eq. (12), the neural network could no longer focus on the samples that have met conditions in Eq. (11), and pay more attention to optimize bounded samples.

5.3.4 ARPL vs. ARPL + Confusing Samples.

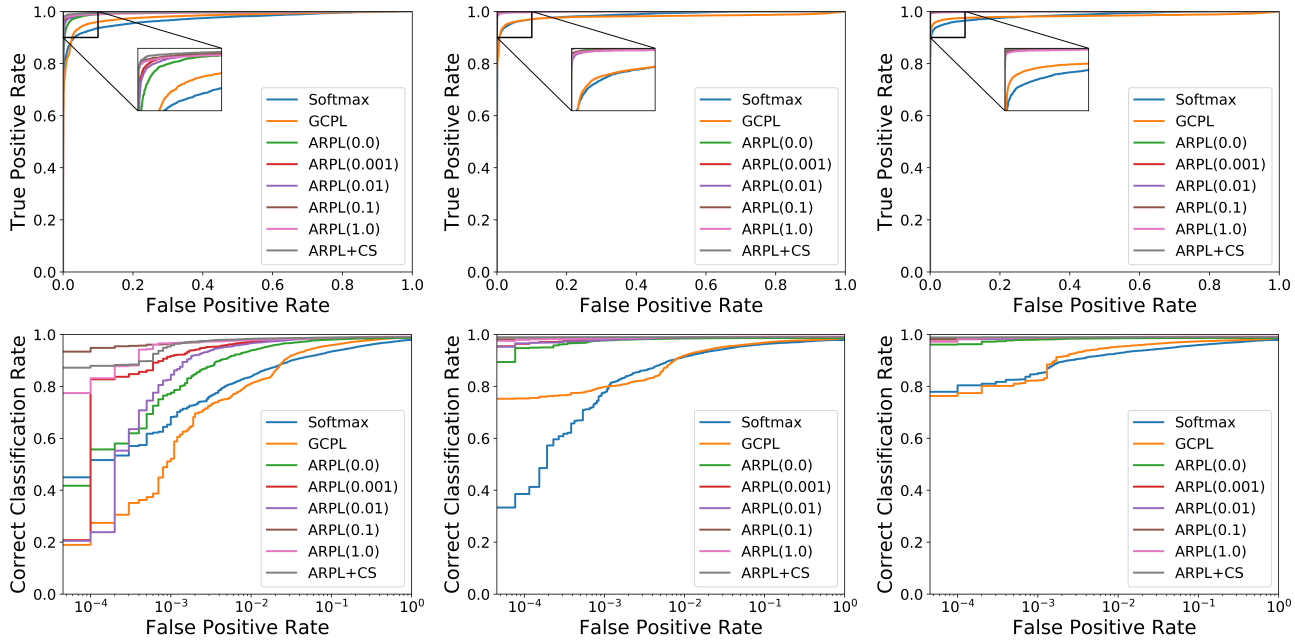
From the experiment for OOD detection, it could be observed that CS improves ARPL more for detecting far OOD.



(a) (b)

Fig. 9: (a) The ablation experiments of λ in CIFAR10 as known data and CIFAR100 as unknown data. (b) The variation trend of margin with the number of known classes.

Meanwhile, as shown in Fig. 8(c) and Fig. 8(d), confusing samples make the difference between MNIST, SVHN and CIFAR100 even greater. Note that SVHN and MNIST have the same class, numbers 0-9, but they are also accurately detected as unknown classes. The main reason for this difference is the direct difference of image domain, color image *vs.* black and white image. This also demonstrates that the proposed method also has the ability to reject data from different domains. For KMNIST, ARPL+CS does not seem to bring much improvement in feature visualization in Fig. 8(d). These samples from KMNIST are difficult to distinguish for the classifier, and have more similarities in the shape and structure with MNIST. However, ARPL+CS still improves the detection ability for these confusing samples, as shown in Table 3 and Fig. 10. In general, confusing samples effectively improve the ability of ARPL to detect various unknown categories under the premise of ensuring the accuracy of known class classification.



(a) known:MNIST, unknown: KMNIST (b) known:MNIST, unknown: SVHN (c) known:MNIST, unknown: CIFAR100

Fig. 10: (1) The first row is the Area Under the Receiver Operating Characteristic (AUROC) applied to the data from MNIST as known and KMNIST, SVHN, CIFAR100 as unknown. (2) The second row is open set Classification Rate curves provided for the same algorithms. Compared with AUROC, more significant differences could be observed through open set Classification Rate curves.

TABLE 4: Test accuracy of different methods on CIFAR10, CIFAR100 and Air-300. The best results are indicated in bold.

Method	CIFAR10	CIFAR100	Air-300
Softmax	93.1	70.8	92.4
GCPL	93.3	70.3	92.3
RPL	93.8	71.8	92.9
ARPL	**94.1**	72.1	94.5
ARPL+CS(w/o ABN)	94.0	71.8	93.2
ARPL+CS(w/o FN)	93.1	71.5	93.0
ARPL+CS	94.0	**72.8**	**94.7**

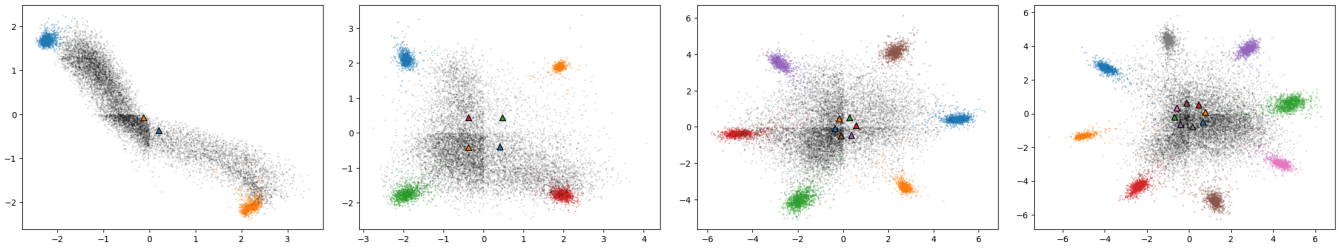
5.3.5 *Analysis of the Margin.*

Different datasets need different sizes for deep features space to ensure that known and unknown could be classified correctly. Fig. 9(b) proves that the margin increases with the number of known classes with fixed λ. Meanwhile, the distributions of features learned under different numbers of known classes still are discriminative for known classes and unknown classes as shown in Fig. 11. This phenomenon demonstrates the rationality of the spatial distribution learned for multiclass. ARPL could effectively control the interaction among different known classes, by learning the more appropriate embedding space size. As a result, the previous conclusion about ARPL still holds for different numbers of known classes.

5.3.6 *Analysis of Closed Set Recognition.*

We adopt ResNet with 34 layers [48] for closed set recognition on CIFAR10, CIFAR100, and **Air**craft **300** (Air-300) [7]. Air-300 contains 320,000 annotated color images from 300 different classes in total. Each category contains 100 images least, and a maximum of 10,000 images, which leads to the long tail distribution. All classes are divided into two parts with 180 known classes for training and 120 novel unknown classes for testing respectively. Compared with the existing benchmark datasets, the tailored Air-300 dataset maintains a long tail distribution to simulate the real visual world. Here, we focus on the closed set accuracy of the model in 180 known categories. The images of Air-300 are center-cropped and resized to 64x64 in this experiment.

As shown in Table 4, ARPL achieves comparable performance with traditional softmax and prototype learning method GCPL. The inter-class distance between the known classes is increased by using reciprocal points to push known classes away from the global open space, so that the neural network can learn more discriminative features for the known classes through ARPL. This demonstrates the effectiveness of ARPL for conventional closed set recognition tasks. Moreover, by combining CS, the closed set accuracy for ARPL+CS is not decreased and even better ARPL can even better than ARPL on CIFAR100 and Air-300. ARPL+CS is not affected by confusing samples, which largely depends on the proposed ABN and FN. Without ABN or FN, the closed set accuracy could be decreased because of the deviation from confusing samples. These results demonstrate that ARPL and ARPL+CS can improve the



(a) $K = 2$ (b) $K = 4$ (c) $K = 6$ (d) $K = 8$

Fig. 11: The learned representations of ARPL with different numbers of known classes. The data is from MNIST by randomly sampling K known classes and $10 - K$ unknown classes. Colored triangles represent the learned reciprocal points of different known classes.

TABLE 5: Performance of three methods using DomainNet. The known is the real-A subset. The type of distribution shift presents a trend of difficulty to the OOD detection problem: Semantic shift (S) > Non-Semantic Shift (NC) > Semantic + Non-Semantic shift.

OOD	Shift		AUROC	OSCR
	S	NS	Softmax / ARPL / ARPL+CS	
real-B	✓		72.3/74.2/**75.2**	41.7/60.8/**61.9**
clipart-A		✓	66.4/70.9/**72.7**	41.4/58.0/**59.4**
clipart-B	✓	✓	77.0/81.1/**82.9**	45.3/65.3/**66.6**
quickdraw-A		✓	77.9/86.1/**86.7**	44.6/68.5/**69.0**
quickdraw-B	✓	✓	79.5/87.4/**87.5**	45.5/69.4/**69.5**

ability of neural networks to distinguish known classes and the discriminability of judging various unknown classes.

5.3.7 *Semantic Shift versus Non-semantic Shift*

More complex open set scenarios is explored through a large-scale data set DomainNet [49]. DomainNet has high-resolution images in 345 classes from six different domains. There are three domains in the dataset with class labels available when the experiments are conveyed. They are real, clipart, and quickdraw, resulting in different types of distribution shifts. To create subsets with semantic shift, all classes is separated into two splits. Split A has class indices from 0 to 172, while split B has 173 to 334. Our experiments uses real-A for in-distribution and has the other subsets for out-of-distribution. With the definition given in [50], real-B has a semantic shift from real-A, while clipart-A has a non semantic shift. Clipart-B therefore has both types of distribution shift. We train the ResNet with 34 layers [48] for 100 epochs with batch size 128 and SGD optimizer with momentum 0.9. The learning rate starts at 0.01 and is dropped by a factor of 0.1 in the training progress every 30 epochs. The images are center-cropped and resized to 80x80 in this experiment.

The results in Table 5 reveal some trends. The first one is that the OOD datasets with both types of distribution shifts are easier to detect, followed by non-semantic shift. The second observation is ARPL could effectively detect all distributions shift compared with Softmax. Especially for OSCR, ARPL has achieved good performance improvement.

TABLE 6: Open set recognition performance of different methods on the larger and more difficult datasets, where ImageNet-1K as the known dataset and ImageNet-O as the unknown dataset.

Method	ACC	AUROC	OSCR
Softmax	69.6	48.2	42.4
ARPL	**70.2**	**60.0**	**48.9**

Finally, confusing samples plays a important role for different domains and can improve the detection performance of ARPL. Near domain could get larger improvement from confusing samples.

5.3.8 *Experiments on ImageNet.*

To better compare our method with traditional softmax, we conduct experiments on the larger and more difficult ImageNet-1K dataset [51]. ImageNet-1K includes 1000 classes with more than 1,200,000 training images and 50K validation images. Moreover, ImageNet-O [52] is adopted as the out-of-distribution dataset for ImageNet-1K. ImageNet-O includes 2K examples from ImageNet-22K [51] excluding ImageNet-1K. The ResNet 18 [48] is trained on ImageNet-1K and tested on both ImageNet-1K and ImageNet-O.

As shown in Table 6, ARPL performs better than traditional softmax even on the large and difficult dataset, in both the close-set accuracy (ACC) and unknown detection (AUROC). Especially for the unknown detection, ARPL achieves about 12% improvement. Due to the constraints on the global open space by reciprocal points, our method can ensure the better separation of known and unknown classes under the condition of accurate recognition of known classes. These results show the excellent scalability of ARPL in larger scale datasets.

6 Implementation Details

γ is set as 1.0, λ and β are set to 0.1, in all training phases. Reciprocal points are initialized by the random normal distribution and each margin is initialized with one. For open set recognition, a global average pooling is added after the final convolution layer of the encoder. In experiments for out-of-distribution, the output after global average pooling

(GAP) of the ResNet is utilized as the feature. The dimension of the reciprocal point is consistent with the output of the GAP. Each known class is assigned one reciprocal point for training. In addition to MNIST, random center-cropped and random horizon flip are used as data augmentation. The images in TinyImageNet are resized to 64x64 in the experiments.

7 Conclusion

This paper formulates the open space risk from the perspective of multi-class integration, by introducing a novel concept, *Reciprocal Point*, to model the extra-class space corresponding to each known category. We introduce a novel learning framework, Adversarial Reciprocal Point Learning, towards reliable open set neural network. Specifically, a classification framework with the adversarial margin constraint is introduced to reduce the empirical classification risk and the open space risk. The rationality of the adversarial margin constraint is theoretically guaranteed by Theorem 1. Furthermore, to estimate the unknown distribution from the open space, an instantiated adversarial enhancement is designed to generate more diverse confusing training samples from the confrontation between the known data and reciprocal points. Our methods breaks the closed-world assumption in traditional neural networks and adopts the open-world reciprocal points for discrimination between known and unknown. Extensive experiments conducted on multiple datasets demonstrate that our method outperforms previous state-of-the-art open set classifiers in all cases.

Meanwhile, this paper also reveals that the recognition of unknown classes by the neural network is mostly based on known priors, so the distribution of unknown classes is more aggregated in the low response of deep feature space, while the known classes are distributed in the high response space. This is very similar to that the neocortical areas get structured knowledge from the hippocampus through interleaved learning [53]. In the future, we will explore more details about the neural mechanism of few shot learning, and then utilize it to improve the ability of the neural network to detect and learn unknown categories.

References

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1026–1034, 2015.

[2] 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.

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

[4] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *Proceedings of European Conference on Computer Vision*, pages 499–515. Springer, 2016.

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

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[7] Guangyao Chen, Limeng Qiao, Yemin Shi, Peixi Peng, Jia Li, Tiejun Huang, Shiliang Pu, and Yonghong Tian. Learning open set network with discriminative reciprocal points. In *Proceedings of European Conference on Computer Vision*, pages 507–522. Springer, 2020.

[8] Akshay Raj Dhamija, Manuel Günther, and Terrance Boult. Reducing network agnostophobia. In *Advances in Neural Information Processing Systems*, pages 9157–9168, 2018.

[9] Peter L Bartlett and Marten H Wegkamp. Classification with a reject option using a hinge loss. *Journal of Machine Learning Research*, 9(8), 2008.

[10] Qing Da, Yang Yu, and Zhi-Hua Zhou. Learning with augmented class by exploiting unlabeled data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28, 2014.

[11] Ming Yuan and Marten Wegkamp. Classification methods with reject option based on convex risk minimization. *Journal of Machine Learning Research*, 11(1), 2010.

[12] 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.

[13] Lalit P Jain, Walter J Scheirer, and Terrance E Boult. Multi-class open set recognition using probability of inclusion. In *Proceedings of European Conference on Computer Vision*, pages 393–409. Springer, 2014.

[14] Ethan M Rudd, Lalit P Jain, Walter J Scheirer, and Terrance E Boult. The extreme value machine. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(3):762–768, 2018.

[15] Pedro R Mendes Júnior, Roberto M de Souza, Rafael de O Werneck, Bernardo V Stein, Daniel V Pazinato, Waldir R de Almeida, Otávio AB Penatti, Ricardo da S Torres, and Anderson Rocha. Nearest neighbors distance ratio open-set classifier. *Machine Learning*, 106(3):359–386, 2017.

[16] He Zhang and Vishal M Patel. Sparse representation-based open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(8):1690–1696, 2017.

[17] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1563–1572, 2016.

[18] Lei Shu, Hu Xu, and Bing Liu. Doc: Deep open classification of text documents. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 2911–2916, 2017.

[19] Stephen Roberts and Lionel Tarassenko. A probabilistic resource allocating network for novelty detection. *Neural Computation*, 6(2):270–284, 1994.

[20] Ryota Yoshihashi, Wen Shao, Rei Kawakami, Shaodi You, Makoto Iida, and Takeshi Naemura. Classification-reconstruction learning for open-set recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4016–4025, 2019.

[21] Xin Sun, Zhenning Yang, Chi Zhang, Keck-Voon Ling, and Guohao Peng. Conditional gaussian distribution learning for open set recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 13480–13489, 2020.

[22] Hongjie Zhang, Ang Li, Jie Guo, and Yanwen Guo. Hybrid models for open set recognition. In *Proceedings of European Conference on Computer Vision*, pages 102–117. Springer, 2020.

[23] 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*, 2017.

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

[25] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. 2017.

[26] Andras Rozsa, Manuel Günther, and Terrance E Boult. Adversarial robustness: Softmax versus openmax. In *Proceedings of the British Machine Vision Conference*, 2017.

[27] 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*, pages 2898–2906, 2019.

[28] Shiyu Liang, Yixuan Li, and R Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *Proceedings of International Conference on Learning Representations*, 2018.

[29] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *Proceedings of International Conference on Learning Representations*, 2018.

[30] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *Proceedings of International Conference on Learning Representations*, 2018.

[31] Ludmila I Kuncheva and James C Bezdek. Nearest prototype classification: Clustering, genetic algorithms, or random search? *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 28(1):160–164, 1998.

[32] Cheng-Lin Liu and Masaki Nakagawa. Evaluation of prototype learning algorithms for nearest-neighbor classifier in application to handwritten character recognition. *Pattern Recognition*, 34(3):601–615, 2001.

[33] Atsushi Sato and Keiji Yamada. A formulation of learning vector quantization using a new misclassification measure. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 1, pages 322–325, 1998.

[34] Edwin V Bonilla and Antonio Robles-Kelly. Discriminative probabilistic prototype learning. In *Proceedings of International Conference on Machine Learning*, pages 1187–1194, 2012.

[35] Claude E Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948.

[36] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of International Conference on Machine Learning*, pages 448–456, 2015.

[37] Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan L Yuille, and Quoc V Le. Adversarial examples improve image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 819–828, 2020.

[38] Zihang Dai, Zhilin Yang, Fan Yang, William W Cohen, and Russ R Salakhutdinov. Good semi-supervised learning that requires a bad gan. In *Advances in Neural Information Processing Systems*, pages 6510–6520, 2017.

[39] Lawrence Neal, Matthew Olson, Xiaoli Fern, Weng-Keen Wong, and Fuxin Li. Open set learning with counterfactual images. In *Proceedings of European Conference on Computer Vision*, pages 613–628. Springer, 2018.

[40] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[41] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. In *Neural Information Processing Systems Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[42] Alex Krizhevsky et al. Learning multiple layers of features from tiny images. Technical report, 2009.

[43] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[44] Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of International Conference on Machine Learning*, pages 233–240, 2006.

[45] Tom Fawcett. An introduction to roc analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

[46] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of International Conference on Learning Representations*, 2015.

[47] Ning Qian. On the momentum term in gradient descent learning algorithms. *Neural Networks*, 12(1):145–151, 1999.

[48] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[49] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1406–1415, 2019.

[50] 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 Conference on Computer Vision and Pattern Recognition*, pages 10951–10960, 2020.

[51] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[52] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *arXiv preprint arXiv:1907.07174*, 2019.

[53] Dharshan Kumaran, Demis Hassabis, and James L McClelland. What learning systems do intelligent agents need? complementary learning systems theory updated. *Trends in Cognitive Sciences*, 20(7):512–534, 2016.