Towards Accurate Open-Set Recognition via Background-Class Regularization

Wonwoo Cho[1,2] and Jaegul Choo[1,2]

[1] KAIST AI
[2] Letsur Inc.
{wcho,jchoo}@kaist.ac.kr

Abstract. In open-set recognition (OSR), classifiers should be able to reject unknown-class samples while maintaining high closed-set classification accuracy. To effectively solve the OSR problem, previous studies attempted to limit latent feature space and reject data located outside the limited space via offline analyses, *e.g.*, distance-based feature analyses, or complicated network architectures. To conduct OSR via a simple inference process (without offline analyses) in standard classifier architectures, we use distance-based classifiers instead of conventional Softmax classifiers. Afterwards, we design a background-class regularization strategy, which uses background-class data as surrogates of unknown-class ones during training phase. Specifically, we formulate a novel regularization loss suitable for distance-based classifiers, which reserves sufficiently large class-wise latent feature spaces for known classes and forces background-class samples to be located far away from the limited spaces. Through our extensive experiments, we show that the proposed method provides robust OSR results, while maintaining high closed-set classification accuracy.

Keywords: Generalized Open-Set Recognition, Distance-Based Classifiers, Background-Class Regularization, Probability of Inclusion

1 Introduction

In machine learning (ML), classification algorithms have achieved great success. Through recent advances in convolutional neural networks, their classification performance already surpassed the human-level performance in image classification [8]. However, such algorithms have usually been developed under a *closed-set* assumption, *i.e.*, the class of each test sample is assumed to always belong to one of the pre-defined set of classes. Although this conventional assumption can be easily violated in real-world applications (classifiers can face unknown-class data), traditional classification algorithms are highly likely to force unknown-class samples to be classified into one of the known classes. To tackle this problem, the *open-set recognition (OSR)* problem [34] aims to properly classify unknown-class samples as "unknown" and known-class samples as one of the known classes.

According to the definition of OSR [34], it is required to properly limit the latent feature space of known-class data. To satisfy the requirement, various OSR

methods were developed based on traditional ML models. Previously, Scheirer *et al.* [33] calibrated the decision scores of support vector machines (SVMs). Based on the intuition that a large set of data samples of unknown classes can be rejected if those of known classes are accurately modeled, Jain *et al.* [13] proposed P_I-SVM, which utilized the statistical modeling of known-class samples located near the decision boundary of SVMs. Afterwards, it was attempted to solve the OSR problem based on the principle of the nearest neighbors [14]. Taking distribution information of data into account, Rudd *et al.* [29] proposed the extreme value machine which utilizes the concept of margin distributions.

Since deep neural networks (DNNs) have robust classification performance by learning high-level representations of data, OSR methods for DNNs have received great attention. Based on the theoretical foundations studied in traditional ML-based OSR methods, Bendale and Boult [1] proposed the first OSR strategy for DNNs called Openmax, which calibrates the output logits of pre-trained Softmax classifiers. To improve Openmax, Yoshihashi *et al.* [41] proposed the classification-reconstruction learning to make robust latent feature vectors. Afterwards, Oza and Patel [27] proposed to exploit a class-conditioned autoencoder and use its reconstruction error to assess each input sample. Sun *et al.* [37] employed several class-conditioned variational auto-encoders for generative modeling.

Although previous methods applied *offline analyses* to pre-trained Softmax classifiers or employed complicated DNN architectures, they have limited performance since the classifiers were trained solely based on known-class data. To mitigate the problem, this paper designs an simple and effective open-set classifier in the *generalized OSR setting*, which uses background-class regularization (BCR) at training time. Despite its effectiveness, BCR has received a little attention in OSR and previous BCR methods [5,11,21] are insufficient to properly solve the OSR problem. In this paper, we denote the infinite label space of all classes as $\mathcal{Y}$ and use the following class categories, whose definition is also provided in [5,7].

- **Known known classes** (KKCs; $\mathcal{K} = \{1, \cdots, C\} \subset \mathcal{Y}$) include distinctly labeled positive classes, where $\mathcal{U} = \mathcal{Y} \setminus \mathcal{K}$ is the entire unknown classes.
- **Known unknown classes** (KUCs; $\mathcal{B} \subset \mathcal{U}$) include background classes, *e.g.*, labeled classes which are not necessarily grouped into a set of KKCs $\mathcal{K}$.
- **Unknown unknown classes** (UUCs; $\mathcal{A} = \mathcal{U} \setminus \mathcal{B}$) represent the rest of $\mathcal{U}$, where UUCs are not available at training time, but occur at inference time.

Also, we denote $\mathcal{D}_t$ as a training set consisting of multiple pairs of a KKC data sample and the corresponding class label $y \in \{1, \cdots, C\}$. $\mathcal{D}_{test}^k$ and $\mathcal{D}_{test}^u$ are test sets of KKCs and UUCs, respectively. $\mathcal{D}_b$ is a background dataset of KUCs.

2 Preliminary Studies

2.1 The Open-Set Recognition Problem

The OSR problem addresses a classification setting that can face test samples from classes unseen during training (UUCs). In this setting, open-set classifiers aim to properly classify KKC samples while rejecting UUC ones simultaneously. A

(a) Latent feature space (b) Closed-set problem (c) Open-set problem
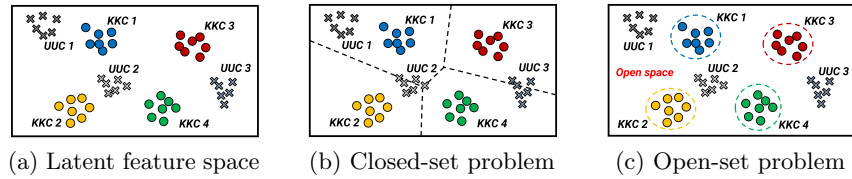
Fig. 1. Given (a) a latent feature space, we demonstrate (b) closed-set and (c) open-set problems, where KKCs and UUCs are known and unknown classes, respectively.

similar problem to OSR is out-of-distribution (OoD) detection [10], which typically aims to reject data items drawn far away from the training data distribution. Conventionally, previous studies such as [10,20,17,18] assumed that OoD samples are drawn from other datasets or can be even noise data. In this paper, we aim to reject test data whose classes are unknown but related to the training data, which narrows down the scope of conventional OoD detection tasks.

Previously, Scheirer *et al.* [34] introduced a formal definition of OSR based on the notion of open-space risk $R_\mathcal{O}$, which is a relative measure of a positively labeled union of balls $\mathcal{S}_V$ and open space $\mathcal{O}$ located far from $\mathcal{S}_V$. Since labeling any data item in $\mathcal{O}$ incurs open-space risk, it is straightforward that a classifier cannot be a solution for the OSR problem if the classifier accepts data in infinitely wide regions, *i.e.*, its open-space risk is unbounded ($R_\mathcal{O} = \infty$). The definition implies that essential requirements to solve the OSR problem are 1) *bounding open-space risk* and 2) *ideally balancing it with empirical risk*.

Unlike traditional classifier models, open-set classifiers are required to limit the latent feature space of KKC data to bound their open-space risk. To ensure open-space risk to be bounded, Scheirer *et al.* [33] introduced compact abating probability (CAP) models. The principle of CAP models is that if the support region of a classifier decays in all directions from the training data, thresholding the region will bound the classifier's open-space risk [2]. As depicted in Figure 1, which compares traditional closed-set and open-set classification problems [7], building proper *class-wise* CAP models is an effective strategy for OSR.

2.2 Post-Classification Analysis for Pre-Trained Softmax Classifier

This paper aims to solve the OSR problem solely based on a standard DNN-based classifier architecture f as a latent feature extractor. Applying a fully-connected layer to f, a conventional Softmax classifier computes the posterior probability of an input $\mathbf{x}$ belonging to the c-th known class by

$$P_s(y = c|\mathbf{x}) = \frac{\exp(\mathbf{w}_c^T f(\mathbf{x}) + b_c)}{\sum_{i=1}^{C} \exp(\mathbf{w}_i^T f(\mathbf{x}) + b_i)}, \tag{1}$$

where $c \in \{1, \cdots, C\}$, $f(\mathbf{x}) \in \mathbb{R}^n$ is the latent feature vector of $\mathbf{x}$, and $\mathbf{w}_c$ and b_c are the weight and bias for the c-th class, respectively. For pre-trained Softmax

classifiers, Hendrycks and Gimpel [10] proposed a baseline technique to detect anomalous samples, which imposes a threshold on the predictive confidence of Eq. (1). When using the baseline approach to solve the OSR problem, one can estimate the class of each KKC sample and recognize UUC data by

$$
\widehat{y} = \begin{cases} \arg\max_{c \in \{1, \cdots, C\}} P_s(y = c | \mathbf{x}), & \text{if } \max_{c \in \{1, \cdots, C\}} P_s(y = c | \mathbf{x}) \geq \tau, \\ C + 1 \text{ (unknown class)}, & \text{otherwise.} \end{cases} \tag{2}
$$

However, Eq. (2) cannot formally bound open-space risk and formulate class-wise CAP models since it only rejects test data near the decision boundary of classifiers, thus having infinitely wide regions of acceptance [2]. Therefore, *post-classification analysis* methods using an auxiliary measure other than the Softmax probability are necessary to build auxiliary CAP models in the latent feature space of f, where *distance measures* have been widely employed in previous studies [1,18].

To build class-wise CAP models, Openmax [1] defined radial-basis decaying functions $\{s(\mathbf{x}, i)\}_{i=1}^{C}$, each of which measures the class-belongingness of $\mathbf{x}$ for the c-th class, in the latent feature space of f. For each $s(\mathbf{x}, c)$, the authors employed distance measures between $f(\mathbf{x})$ and an empirical class mean vector $\boldsymbol{\mu}_c$, *e.g.*, $s(\mathbf{x}, c) = D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c) = (f(\mathbf{x}) - \boldsymbol{\mu}_c)^T (f(\mathbf{x}) - \boldsymbol{\mu}_c)$. To formulate more effective CAP models, they statistically analyzed the distribution of $s(\mathbf{x}, c)$ based on the extreme value theory (EVT) [32], which provides a theoretical foundation that the Weibull distribution is suitable for modeling KKC samples located far from the class mean vectors (extreme samples). To be specific, Openmax fits a Weibull distribution on extreme samples of the c-th class having the highest $D_E(f(\mathbf{x}), \boldsymbol{\mu}_c)$ values, where its cumulative distribution function (CDF) formulates the *probability of inclusion* $P_I(\mathbf{x}, c)$ [13,29], *i.e.*, $P_I(\mathbf{x}, c) = 1 - \texttt{WeibullCDF}$, which rapidly decays near the extreme samples. Based on $P_I(\mathbf{x}, c)$, the decision rule of Eq. (2) can be calibrated to conduct OSR with Softmax classifiers.

2.3   Background-Class Regularization

Although they need additional inference procedures (*e.g.*, EVT modeling), previous offline analyses may have limited OSR performance since the classifiers were trained solely based on known-class data. To obtain robust empirical results without complicated analyses, one can use the strategy of BCR at the training phase, which exploits background-class (KUC) samples as surrogates of UUC data. Geng *et al.* [7] argued that the generalized OSR setting that utilizes KUC samples is still less-explored and an important research direction for robust OSR.

Conventionally, a loss function for training classifiers with BCR can be

$$
\mathcal{L} = \mathcal{L}_{cf} + \lambda \mathcal{L}_{bg} = \mathbb{E}_{(\mathbf{x}^k, y) \sim \mathcal{D}_t} \left[-\log P_s(y | \mathbf{x}^k) + \lambda \mathbb{E}_{\mathbf{x}^b \sim \mathcal{D}_b} \left[f_{reg} \left(\mathbf{x}^k, y, \mathbf{x}^b \right) \right] \right], \tag{3}
$$

where $\mathcal{L}_{cf}$ and $\mathcal{L}_{bg}$ are the loss terms for closed-set classification and BCR, respectively, and λ is a hyperparameter. For $\mathcal{L}_{bg}$, previous studies designed their own f_{reg}, where [5] proposed the objectosphere loss for OSR, and [11] and [21] employed the uniformity and the energy losses for OoD detection, respectively.

In this paper, we tackle the following limitations of the previous BCR methods.

- In the previous BCR methods, $\mathcal{L}_{bg}$ were designed to make normal data and anomalies more distinguishable in terms of the corresponding anomaly scores. Since they categorized normal data into a single group (did not consider the classes) in $\mathcal{L}_{bg}$, the previous methods may have limited performance in rejecting UUC data and maintaining robust closed-set classification results.
- The previous methods using the decision rule of Eq. (2) (*e.g.*, objectosphere [5] and uniformity [11]) cannot bound open-space risk. Although one can use post-classification analyses to bound open-space risk, trained latent feature space can be inappropriate for using another metric such as distance measures.
- To increase the gap between KKC and KUC data in terms of latent feature magnitude and energy in the objectosphere [5] and the energy [21] losses, respectively, it is necessary to find proper margin parameters for each dataset.

3 Proposed Method

3.1 Overview

Using a standard classifier f, this paper aims to design open-set classifiers having simple yet effective inference steps. In the following, we summarize our method.
- Instead of applying fully-connected layers to feature extractors f, we use the principle of linear discriminant analysis (LDA) [24] to classify images based on a distance measure. By simply imposing a threshold on the distance as in Eq. (2), our classifiers can easily build class-wise CAP models. (Section 3.2)
- Afterwards, we propose a novel BCR strategy suitable for the distance-based classifiers. Following the convention of Eq. (3), we design our own $\mathcal{L}_{bg}$ called *class-inclusion loss*, where our total loss is function defined by

$$\mathcal{L} = \mathcal{L}_{cf} + \lambda\mathcal{L}_{bg} = \mathcal{L}_{cf} + \lambda(\mathcal{L}_{bg,k} + \mathcal{L}_{bg,u}) \tag{4}$$

  The class-inclusion loss first limits the feature space of KKC data by formulating *explicit* class-wise boundaries, and then forces KUC data to be located outside the boundaries at each training iteration. Our loss is designed to increase the distance gaps between KKC and KUC samples while maintaining robust closed-set classification performance. (Sections 3.3 and 3.4)

For a better understanding of the training and inference processes of our method, we provide their detailed algorithm in our supplementary materials.

3.2 Distance-Based Classification Models

Distance-based classifiers. To train a robust open-set classifier, we formulate a *distance-based classifier* as an alternative of Eq. (1):

$$P_d(y = c|\mathbf{x}) = \frac{P_c \cdot \mathcal{N}(f(\mathbf{x})|\boldsymbol{\mu}_c, \mathbf{I})}{\sum_{i=1}^{C} P_i \cdot \mathcal{N}(f(\mathbf{x})|\boldsymbol{\mu}_i, \mathbf{I})} = \frac{P_c \cdot \exp\left(-D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c)\right)}{\sum_{i=1}^{C} P_i \cdot \exp\left(-D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_i)\right)} \tag{5}$$

where Eq. (5) uses the principle of LDA and $\mathcal{L}_{cf} = \mathbb{E}_{(\mathbf{x}^k, y) \sim \mathcal{D}_t}[-\log P_d(y|\mathbf{x}^k)]$. In Eq. (5), we exploit an identity covariance matrix $\mathbf{I}$ and $P_c = P(y = c) = C^{-1}$ for

all c for KKCs. The classifier estimates the class of each $\mathbf{x}$ via $D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c) = (f(\mathbf{x}) - \boldsymbol{\mu}_c)^T(f(\mathbf{x}) - \boldsymbol{\mu}_c)$, the Euclidean distance between $f(\mathbf{x}) \in \mathbb{R}^n$ and $\boldsymbol{\mu}_c \in \mathbb{R}^n$, where we call $\boldsymbol{\mu}_c$ a *class-wise anchor*. To ensure sufficiently large distance gaps between the pairs of initial class-wise anchors, we randomly sample each $\boldsymbol{\mu}_c$ from the standard Gaussian distribution and then set each $\boldsymbol{\mu}_c$ as a trainable vector. For distance analysis results of such randomly sampled vectors, see [12].

Decision rule. At inference time, each KKC sample $\mathbf{x}$ can be classified via $\widehat{y} = \arg\min_{c \in \{1, \cdots, C\}} D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c)$. Furthermore, applying a threshold to $D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c)$ can bound open-space risk by formulating class-wise CAP models as follows:

$$\widehat{y} = \begin{cases} \arg\min_{c \in \{1, \cdots, C\}} D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c), & \text{if } \max_{c \in \{1, \cdots, C\}} -D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c) \geq \tau, \\ C + 1 \text{ (unknown class)}, & \text{otherwise.} \end{cases}$$
(6)

As Eq. (6) employs the same metric D_E for classification and UUC rejection, our method may support more accurate latent feature space analysis for OSR than the previous OSR methods using post-classification analyses.

The concept of distance-based classification was also employed in prototypical networks [36], nearest class mean classifiers [23], and the previous studies of the center loss function [38] and convolutional prototype classifiers [40]. In addition, polyhedral conic classifiers [3] used the idea of returning compact class regions for KKC samples based on distance-based feature analyses. It is noteworthy that our main contribution is a novel BCR method that can effectively utilize KUC samples in a distance-based classification scheme (described in Sections 3.3 and 3.4), not the distance-based classifier method itself. To the best of our knowledge, we are the first to discuss the necessity of distance-based BCR methods for OSR and propose a reasonable regularization method for distance-based classifiers.

3.3 Background Class Regularization for Distance-based Classifiers

Intuition and hypersphere classifiers. To obtain robust OSR performance via Eq. (6), we aim to design a BCR method suitable for distance-based classifiers, which uses $\mathcal{D}_t$ and $\mathcal{D}_b$ as surrogates of $\mathcal{D}_{test}^k$ and $\mathcal{D}_{test}^u$ at training time, respectively. Although it cannot provide any information of $\mathcal{D}_{test}^u$, $\mathcal{D}_b$ can be effective to limit the latent feature space of KKCs, while reserving space for UUCs. With $\mathcal{D}_b$, it is intuitive that the primary objective of BCR for Eq. (6) is to make KUC samples located far away from $\boldsymbol{\mu}_i$ for all classes $i \in \{1, \cdots, C\}$.

Before we illustrate our BCR method, we first introduce hypersphere classifiers (HSCs) [30]. An HSC conducts anomaly detection by using a feature extractor g, where its anomaly score for an input $\mathbf{x}$ is the Euclidean distance between a single center vector $\boldsymbol{\mu}$ and $g(\mathbf{x})$. When training the HSC model, the authors used normal and background data, $\mathcal{D}_t$ and $\mathcal{D}_b$, respectively, and a loss function

$$\mathbb{E}_{\mathbf{x}^k \sim \mathcal{D}_t} \left[h \left(D_E^2 \left(g(\mathbf{x}^k), \boldsymbol{\mu} \right) \right) \right] - \mathbb{E}_{\mathbf{x}^b \sim \mathcal{D}_b} \left[\log \left(1 - \exp \left(-h \left(D_E^2 \left(g(\mathbf{x}^b), \boldsymbol{\mu} \right) \right) \right) \right) \right].$$
(7)

The loss function is designed to decrease the Euclidean distances between normal samples $\mathbf{x}^k$ and $\boldsymbol{\mu}$ while increasing the distances for background samples $\mathbf{x}^b$. In Eq. (7), $h(x) = \sqrt{x+1}-1$, which implies that the Euclidean distance $D_E^2(g(\mathbf{x}), \boldsymbol{\mu})$ is scaled into the range of $(0, 1]$ via $\exp(-h(D_E^2(g(\mathbf{x}), \boldsymbol{\mu})))$.

Background-class regularization strategy. It is straightforward that the decision rule of Eq. (6) employs the principle of HSCs in a class-wise manner. In other words, the class-wise HSC for the c-th class determines whether a test sample belongs to the c-th class by computing $D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c)$, where the input is determined as UUC if the entire class-wise HSCs reject the data item. Thus, a proper BCR strategy for distance-based classifiers should force each KUC sample $\mathbf{x}^b$ to be rejected by the entire class-wise HSCs (increase $D_E^2(f(\mathbf{x}^b), \boldsymbol{\mu}_i)$ for all i). Since it is inefficient to consider the entire KKCs to regularize f with $\mathbf{x}^b$ at each iteration, we approximate the process by only taking the *closest* class-wise HSC into account (increase $\min_{i \in \{1, \cdots, C\}} D_E^2(f(\mathbf{x}^b), \boldsymbol{\mu}_i)$).

Although one can adopt Eq. (7) to formulate $\mathcal{L}_{bg}$ for distance-based classifiers, scaling $D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c)$ into $(0, 1]$ via $\exp(-h(D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c)))$, which rapidly decays near $\boldsymbol{\mu}_c$, can be insufficient to move KUC data far away from class-wise anchors. Therefore, we design $\mathcal{L}_{bg}$ that can guarantee sufficient spaces for KKC data and simultaneously force KUC samples located outside the limited class-wise spaces.

3.4   Probability of Inclusion and Class-Inclusion Loss

As we described in Section 2.2, the probability of inclusion builds effective CAP models, since it is designed to rapidly decay near extreme data, *i.e.*, $P_I(\mathbf{x}, c) \approx 1$ in the region that a majority of class-c KKC samples are located. In the following, we introduce a novel regularization method for distance-based classifiers based on the principle of the probability of inclusion, and then design a loss function.

Probability of inclusion for distance-based classifiers. For pre-trained Softmax classifiers, Openmax [1] formulated the probability of inclusion via EVT modeling at inference time, where the strategy is to find *implicit* class-wise boundaries that distinguish KKCs from UUCs. However, such EVT-based analysis can be intractable at each training iteration, since it requires computationally-expensive and parameter-sensitive processes. In addition, it is inappropriate to make boundaries by analyzing features which are not properly trained yet.

Thus, we build *explicit* class-wise boundaries by formulating $P_I(\mathbf{x}, c)$ based on the underlying assumption of LDA, and then use the boundaries for regularization without additional analysis of latent feature distribution. Under the assumption of LDA that each class-c latent feature vector is drawn from a unimodal Gaussian distribution $\mathcal{N}(f(\mathbf{x})|\boldsymbol{\mu}_c, \mathbf{I})$, the Euclidean distance $D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c)$, a simplified version of the Mahalanobis distance, can be assumed to follow the Chi-square distribution having the degree of freedom n. Then, we have

$$P\left(D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c) = t\right) = \frac{t^{\frac{n}{2}-1}}{2^{\frac{n}{2}} \cdot \Gamma(n/2)} \cdot \exp\left(-\frac{t}{2}\right), \tag{8}$$

where $t \geq 0$, $\Gamma(\cdot)$ is the Gamma function, and n is the dimension of $f(\mathbf{x})$.

As previous studies [13,1,29] formulated the probability of inclusion by computing the CDF of the Weibull distribution, *i.e.*, $P_I(\mathbf{x}, c) = 1 - \texttt{WeibullCDF}$, we define our $P_I(\mathbf{x}, c)$ by using the CDF of Eq. (8) as follows:

$$P_I(\mathbf{x}, c) = 1 - \int_0^{D_E^2(\mathbf{x},c)/2} \frac{t^{n/2-1}}{\Gamma(n/2)} \cdot \exp{(-t)} \, dt = \frac{\Gamma(n/2, D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c)/2)}{\Gamma(n/2)}, \quad (9)$$

where $\Gamma(\cdot, \cdot)$ is the upper incomplete Gamma function. It is noteworthy that Eq. (9) can be easily computable via $\texttt{igammac}$ function in PyTorch [28].

Class-inclusion loss function. Based on $\mathcal{D}_t$, $\mathcal{D}_b$, and our $P_I(\mathbf{x}, c)$ of Eq. (9), the primary objective of the proposed BCR strategy, which aims to force each KUC data sample to be located far away from the closest class-wise HSC, can be achieved by employing a loss function $\mathcal{L}_{bg,u} = \mathbb{E}_{\mathbf{x}^b \sim \mathcal{D}_b}[-\log(1 - \max_{i \in \{1,\cdots,C\}} P_I(\mathbf{x}^b, i))]$. To compare $P_I(\mathbf{x}, c)$ and $P_H(\mathbf{x}, c) = \exp(-h(D_E^2(f(\mathbf{x}), \boldsymbol{\mu}_c)))$, which was used in Eq. (7), we plot $P_I(\mathbf{x}, c)$ and $P_H(\mathbf{x}, c)$ in
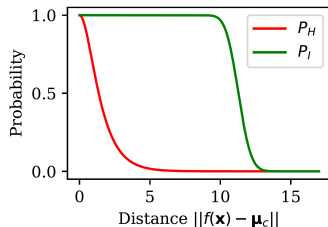
Fig. 2. P_H and P_I (Ours)

Figure 2 with respect to $||f(\mathbf{x}) - \boldsymbol{\mu}_c||$ by assuming $n = 128$. The figure implies that unlike $P_H(\mathbf{x}, c)$, our $P_I(\mathbf{x}, c)$ can assign sufficiently large space for KKC data and force KUC samples to be located outside the space. Also, it is noteworthy that our regularization method based on $P_I(\mathbf{x}, c)$ does not require any margin parameters dependent on datasets or the dimension of latent features.

At training time, $P_I(\mathbf{x}, c) = 0.5$ constructs an auxiliary decision boundary between the c-th class KKC data and the other data items, where $\mathcal{L}_{bg,u}$ makes a majority of KUC data to be located outside the entire class-wise boundaries. However, $\mathcal{L}_{bg,u}$ can be insufficient to achieve robust UUC rejection and closed-set classification results, since it does not control correctly classified KKC samples to be located inside the corresponding class-wise boundaries. Therefore, in addition to $\mathcal{L}_{cf} = \mathbb{E}_{(\mathbf{x}^k, y) \sim \mathcal{D}_t}[-\log P_d(y|\mathbf{x}^k)]$, we apply another loss $\mathcal{L}_{bg,k}$ to KKC data to *maintain* high closed-set classification accuracy and enhance the gap between KKC and KUC samples in terms of the Euclidean distance. By formulating $\mathcal{L}_{bg,k} = \mathbb{E}_{(\mathbf{x}^k, y) \sim \mathcal{D}_t}[-\mathbb{1}(y = \hat{c}) \log(P_I(\mathbf{x}^k, \hat{c}))]$, where $\hat{c} = \arg\max_{i \in \{1,\cdots,C\}} P_I(\mathbf{x}^k, i)$, we define our $\mathcal{L}_{bg}$ as $\mathcal{L}_{bg,k} + \mathcal{L}_{bg,u}$ and call $\mathcal{L}_{bg}$ the *class-inclusion loss*.

In our total loss (Eq. (4)), $\mathcal{L}_{cf}$ makes KKC samples be correctly classified, $\mathcal{L}_{bg,u}$ makes KUC samples located outside the explicit class-wise boundaries, and $\mathcal{L}_{bg,k}$ additionally regularizes correctly classified KKC samples. It is noteworthy that we use an additional loss for KKC samples after they are correctly classified, to prevent obstructions in training closed-set classifiers at early iterations.

4   Experiments

Through extensive experiments, we compared our class-inclusion loss for distance-based classifiers to the objectosphere [5], the uniformity (also widely known as

OE) [11], and the energy [21] losses for conventional Softmax classifiers. This section aims to show that whether our approach provides competitive UUC rejection results, while keeping high closed-set classification accuracy. Furthermore, we conducted additional experiments and provided the corresponding discussions.

4.1   Experimental Settings

For evaluation, we first measured the closed-set classification accuracy. To quantify the accuracy of UUC data rejection, we also measured the area under the receiver operating characteristic curve (AUROC). Also, we used the open-set classification rate (OSCR) as additional OSR accuracy measure by quantifying the correct closed-set classification rate when the false positive rate for UUC rejection is 10^{-1}. For in-depth details of OSCR, see [5]. As $\mathcal{D}_b$, we used ImageNet [31], which was also employed in [19]. To ensure that the classes of $\mathcal{D}_b$ and our test sets are disjoint, we used only the remaining classes of ImageNet, which are not included in the test sets. In our experiments, we considered the following two settings.

Setting 1. In Setting 1, a single dataset was split into KKCs and UUCs, where we used the KKCs in the training set as $\mathcal{D}_t$, and the KKCs and UUCs in the test set as $\mathcal{D}_{test}^k$ and $\mathcal{D}_{test}^u$, respectively. Following the protocol in [25], which were also employed in [27,37], we conducted experiments by using the following standard datasets: SVHN [26], CIFAR10 & CIFAR100 [15], and TinyImageNet [16].

SVHN, CIFAR10   For SVHN and CIFAR10, each of which consists of images of 10 classes, each dataset was randomly partitioned into 6 KKCs and 4 UUCs.

CIFAR+10, CIFAR+50   For CIFAR+M, we employed randomly selected 4 classes of CIFAR10 as KKCs and M classes of CIFAR100 as UUCs.

TinyImageNet   For a larger number of classes, we randomly selected 20 classes of TinyImageNet as KKCs and then used the remaining 180 classes as UUCs.

Setting 2. By using the training and the test sets of a single dataset as $\mathcal{D}_t$ and $\mathcal{D}_{test}^k$, respectively, we employed the test set of another dataset relatively close to $\mathcal{D}_t$ as $\mathcal{D}_{test}^u$ in Setting 2. Adopting the experiment settings in [41] and [20], we used the entire classes of a dataset as KKCs for CIFAR10 & CIFAR100. For UUC dataset, TinyImageNet, LSUN [42], and iSUN [39] were selected. TinyImageNet and LSUN consists of 10,000 test samples each, where the samples in each dataset were resized (R) or cropped (C) into the size 32×32. The iSUN dataset has 8,925 test samples and they were also resized into the size of 32×32. The modified datasets can be obtained in the Github repository of [20].

4.2   Training Details

Network selection. For f, we employed the Wide-ResNet (WRN) [43] and then used its penultimate layer $f(\mathbf{x}) \in \mathbb{R}^n$ for the latent feature vector of each

Table 1. Comparison with the previous BCR methods in the first setting.

Experiments	Accuracy (↑)			AUROC (↑)			OSCR (↑)		
	Objectosphere / Uniformity / Energy / Class-inclusion (Ours)								
SVHN	0.968 / 0.966 / 0.972 / <u>0.974</u>			0.935 / 0.927 / 0.911 / <u>0.956</u>			0.813 / 0.793 / 0.774 / <u>0.854</u>		
CIFAR10	0.964 / 0.964 / 0.956 / <u>0.973</u>			0.942 / 0.923 / 0.933 / <u>0.948</u>			0.851 / 0.814 / 0.807 / <u>0.870</u>		
CIFAR+10	0.958 / 0.969 / 0.949 / <u>0.976</u>			0.945 / 0.950 / 0.936 / <u>0.961</u>			0.839 / 0.867 / 0.808 / <u>0.881</u>		
CIFAR+50				0.944 / 0.942 / 0.937 / <u>0.957</u>			0.837 / 0.837 / 0.808 / <u>0.865</u>		
TinyImageNet	0.778 / 0.779 / 0.715 / <u>0.802</u>			0.755 / 0.771 / 0.727 / <u>0.785</u>			0.484 / 0.488 / 0.357 / <u>0.493</u>		

input sample $\mathbf{x}$. For CIFAR10 and TinyImageNet, we used WRN 40-2 with a dropout rate of 0.3, where WRN 28-10 was employed for CIFAR100 with the same dropout rate. For SVHN, we used WRN 16-4 with a dropout rate of 0.4. Such network selection was determined by referring the experiments in [11,43].

Parameters. For the entire BCR methods, we set the mini-batch sizes of KKC training samples and KUC samples to 128. We kept λ as a constant during training, *i.e.*, each f was trained with the BCR method *from scratch*. To select hyperparameters and margin parameters of the previous regularization methods, we followed the official implementations[3,4,5]. For SVHN, CIFAR10, CIFAR100, and TinyImageNet, we trained the corresponding classifiers for 80, 100, 200, and 200 epochs, respectively, where we used the stochastic gradient descent for optimization. For SVHN and the other datasets, we used initial learning rates of 0.01 and 0.1, respectively, and a cosine learning rate decay [22]. We also used the learning rate warm-up strategy for the first 5 epochs of each training process.

4.3   Results

The OSR results of our proposed approach and the previous methods are reported in Tables 1 and 2. All the reported values were averaged over five randomized trials, by randomly sampling seeds, data splits of KKCs and UUCs, and class-wise anchors. In the tables, ↑ and ↓ indicate higher-better and lower-better measures, respectively, where underlined values present the best scores.

Setting 1. For the first setting, Table 1 compares our proposed BCR methods for distance-based classifiers with the previous approachs designed for Softmax classifiers. The results demonstrate that our proposed method obtained robust UUC rejection results, which were superior to the results of the previous approaches. It is noteworthy that our method achieved higher classification accuracy values, which were critical in acquiring better OSR results in terms of the OSCR measure, than the previous methods. Such results imply that the proposed framework effectively satisfies the two essential requirements described in Section 2.1, bounding open-space risk and ideally balancing it with empirical risk.

---

[3] `https://github.com/Vastlab/Reducing-Network-Agnostophobia`

[4] `https://github.com/hendrycks/outlier-exposure`

[5] `https://github.com/wetliu/energy_ood`

Table 2. Comparison with the previous methods in the second setting. The corresponding classification accuracy values are reported in the first column.

$\mathcal{D}_t/\mathcal{D}_{test}^k$	$\mathcal{D}_{test}^u$	AUROC (↑)	OSCR (↑)
		Objectosphere / Uniformity / Energy / Class-inclusion (Ours)	
CIFAR10 0.940 / 0.939 / 0.925 / <u>0.947</u>	ImageNet-C	0.988 / 0.986 / 0.981 / <u>0.989</u>	0.929 / 0.928 / 0.894 / <u>0.932</u>
	ImageNet-R	0.979 / 0.984 / 0.972 / <u>0.984</u>	0.923 / 0.926 / 0.886 / <u>0.927</u>
	LSUN-C	<u>0.994</u> / 0.990 / 0.989 / 0.993	0.938 / 0.931 / 0.904 / <u>0.940</u>
	LSUN-R	0.985 / 0.988 / 0.984 / <u>0.990</u>	0.928 / 0.931 / 0.897 / <u>0.935</u>
	iSUN	0.985 / 0.989 / 0.984 / <u>0.991</u>	0.928 / 0.932 / 0.896 / <u>0.936</u>
	Average	0.986 / 0.987 / 0.982 / <u>0.989</u>	0.929 / 0.930 / 0.895 / <u>0.934</u>
CIFAR100 0.727 / 0.735 / 0.705 / <u>0.779</u>	ImageNet-C	0.886 / 0.929 / 0.925 / <u>0.930</u>	0.641 / 0.686 / 0.652 / <u>0.696</u>
	ImageNet-R	0.815 / 0.910 / <u>0.934</u> / 0.920	0.572 / 0.674 / 0.658 / <u>0.687</u>
	LSUN-C	<u>0.967</u> / 0.931 / 0.901 / 0.965	0.685 / 0.680 / 0.643 / <u>0.751</u>
	LSUN-R	0.844 / 0.930 / <u>0.959</u> / 0.945	0.608 / 0.695 / 0.684 / <u>0.731</u>
	iSUN	0.842 / 0.923 / 0.954 / <u>0.955</u>	0.603 / 0.689 / 0.680 / <u>0.734</u>
	Average	0.871 / 0.925 / 0.935 / <u>0.943</u>	0.621 / 0.685 / 0.663 / <u>0.720</u>

Setting 2. In Table 2, we present our experiment results of the second setting. When using the CIFAR10 and CIFAR100 datasets as KKC data, our approach achieved the highest closed-set classification accuracy, which is consistent with the experiment results of Setting 1. Furthermore, by averaging the AUROC and the OSCR values over the various UUC datasets, the table shows that our model outperformed the previous methods in the second setting.

Average runtime. We conducted all the experiments with PyTorch and two GeForce RTX 3090 GPUs. At each trial in the CIFAR10 experiment of Setting 1, the running time of each training epoch took 28 seconds for our method, where its OSR evaluation required approximately 6.5 seconds. We observed that the other methods take similar running time at their training and inference phases.

4.4   Additional Experiments and Discussions

We further analyzed our BCR method by using various λ in our loss function. Furthermore, we compared our method by formulating another baseline using the triplet loss [35]. Using the CIFAR10 and TinyImageNet experiments in Setting 1, we present the corresponding OSR results. We also conducted various additional experiments that can show the effectiveness of our proposed method.

Selecting λ. Conducting additional OSR experiments with $\lambda \in \{0.1, 0.5, 1, 5, 10\}$ in our loss function $\mathcal{L} = \mathcal{L}_{cf} + \lambda\mathcal{L}_{bg}$, Table 3 presents that our method provides robust OSR accuracy across a *wide range* of λ, *e.g.*, $\lambda \in [1, 5]$, which implies that users can flexibly select λ in our method. Although such range may depend on datasets, users are not required to carefully adjust the λ parameter. In additional experiments, $\lambda = 5$ yielded the best OSR results in the SVHN and CIFAR + M experiments of Setting 1 and the CIFAR10 experiments of Setting 2, where

Table 3. OSR results with various λ in our class-inclusion and the triplet losses. In each cell, the results are presented in the form of (Accuracy / AUROC / OSCR).

Parameter λ	CIFAR10		TinyImageNet	
	Class inclusion	Triplet	Class inclusion	Triplet
0 (Vanilla)	0.963 / 0.759 / 0.472	–	0.785 / 0.631 / 0.308	–
0.1	0.966 / 0.899 / 0.742	0.963 / 0.820 / 0.537	0.790 / 0.765 / 0.442	0.787 / 0.746 / 0.431
0.5	0.967 / 0.928 / 0.810	0.968 / 0.842 / 0.572	0.794 / 0.775 / 0.464	0.785 / 0.729 / 0.426
1	0.973 / 0.936 / 0.840	0.966 / 0.860 / 0.628	0.802 / 0.785 / 0.493	0.793 / 0.714 / 0.423
5	0.973 / 0.948 / 0.870	0.965 / 0.872 / 0.628	0.798 / 0.783 / 0.480	0.785 / 0.707 / 0.360
10	0.968 / 0.947 / 0.863	0.958 / 0.856 / 0.597	0.787 / 0.701 / 0.361	0.738 / 0.637 / 0.220

$\lambda = 0.5$ showed the best results in the CIFAR100 experiments. Such empirical results implies that a lower λ value can be better when handling more KKCs.

Triplet loss. We propose a distance-based BCR method suitable for the OSR problem, where the proposed method defines explicit class-wise boundaries and then increases the distance gap between KKC and KUC samples based on the boundaries. Another loss function that can separate KKC and KUC data in terms of such distance measure is the triplet loss, where the loss function has been widely employed to control the distances between latent feature vectors effectively. Therefore, we formulated a baseline distance-based BCR method by following the conventional definition of the triplet loss $\mathcal{L}_{tri}$, where we set class-wise anchors, KKC training data, and KUC data as anchors, positive samples, and negative samples, respectively. Since we observed that training classifiers solely based on the triplet loss $\mathcal{L} = \mathcal{L}_{tri}$ yields significantly worse OSR results in comparison with the regularization method $\mathcal{L} = \mathcal{L}_{cf} + \lambda\mathcal{L}_{tri}$, we employed $\mathcal{L}_{tri}$ as a regularization loss function for BCR. In Table 3, we reported experiment results by using the triplet loss as $\mathcal{L}_{bg}$. The results show that our proposed method (class-inclusion loss) outperforms the regularization method based on the triplet loss.

Vanilla distance-based classifiers. To show the effectiveness of our method, we assessed the OSR performance of vanilla distance-based classifiers (trained solely based on $\mathcal{L}_{cf}$), where we present the results in the form of (Accuracy / AUROC / OSCR). In the CIFAR10 and TinyImageNet experiments of Setting 1, we obtained (0.962 / 0.757 / 0.470) and (0.785 / 0.629 / 0.315), respectively. In the CIFAR10 and CIFAR100 experiments of Setting 2, the OSR results averaged over the five UUC datasets in vanilla distance-based classifiers were (0.936 / 0.838 / 0.709) and (0.766 / 0.807 / 0.549), respectively. Comparing these results to the results in Tables 1 and 2, we show that our regularization strategy can significantly improve the OSR performance of distance-based classifiers.

Ablation study on loss terms. Recall our loss function $\mathcal{L}_{cf} + \lambda(\mathcal{L}_{bg,k} + \mathcal{L}_{bg,u})$. As $\mathcal{L}_{bg,u}$ is essential for BCR by making KUC samples located outside explicit class-wise boundaries, we conducted an ablation study to investigate the necessity of $\mathcal{L}_{bg,k}$. In the absence of $\mathcal{L}_{bg,k}$, which additionally regularizes correctly classified KKC data, we obtained the result of (0.963 / 0.821 / 0.509) for the (Accuracy /

Table 4. Comparison with the previous OSR methods (Macro-averaged F1 score).

Experiments	Setting 1				Setting 2			
	SVHN	CIFAR10	CIFAR+10	CIFAR+50	ImageNet-C	ImageNet-R	LSUN-C	LSUN-R
Softmax [10]	0.725	0.600	0.701	0.637	0.639	0.653	0.642	0.647
Openmax [1]	0.737	0.623	0.731	0.676	0.600	0.684	0.657	0.668
CROSR [41]	0.753	0.668	0.769	0.684	0.721	0.735	0.720	0.749
CGDL [37]	0.776	0.655	0.760	0.695	0.840	0.832	0.806	0.812
AOSR [6]	0.842	0.705	0.773	0.706	0.798	0.795	0.839	0.838
Ours	0.854	0.761	0.805	0.732	0.876	0.869	0.880	0.877

AUROC / OSCR) measures in the CIFAR10 experiment of Setting 1, which is worse than our original result (0.973 / 0.948 / 0.870). This result implies that $\mathcal{L}_{bg,k}$ is necessary to increase the distance gap between KKC and KUC data.

Also, by designing $\mathcal{L}_{bg}$ based on the original HSC loss function (Eq. (7)), we obtained the result of (0.950 / 0.634 / 0.338) for (Accuracy / AUROC / OSCR) in the CIFAR10 experiment of Setting 1, which supports our hypothesis.

Previous OSR approaches. We additionally compared our proposed approach to previous OSR methods, whose OSR results are already reported in [41,6]. For fair comparison, the entire methods presented in Table 4 (including ours) were implemented by using a VGG backbone and tested based on the codebase of (`https://github.com/Anjin-Liu/Openset_Learning_AOSR`). The table, which presents OSR results based on the macro-averaged F1 score measure, shows that our distance-based BCR approach can achieve robust OSR results via a simple inference process in standard classifier architectures.

ResNet-18 architecture. In our main experiments, we used the WRN architectures as feature extractors. To further investigate the effectiveness of our method, we used another standard classifier architecture, ResNet-18 [9]. In the first setting, we obtained the quantitative results of (0.963 / 0.945 / 0.844), (0.966 / 0.950 / 0.845), (0.967 / 0.945 / 0.848), and (0.971 / 0.947 / 0.850) for the regularization methods using the objectosphere [5], the uniformity [11], the energy [21], and our class-inclusion losses, respectively. In addition to the results, Table 5 shows quantitative results in the second setting, where the results imply that our method can outperform the previous BCR methods with ResNet-18, as we observed in the experiments using the WRN architectures.

Text classification. To show that our proposed BCR method can be applicable in another domain, we compared our class-inclusion loss to the uniformity loss in text classification applications. For text classification, we used 20 Newsgroups and WikiText103 for KKCs and KUCs, respectively, and trained a simple GRU model [4] for f as in [11]. As UUC sets, we used Multi30K, WMT16, and IMDB. Since the margin parameters of the objectosphere and the energy losses selected for image classification cannot be suitable for the text classification tasks, we only tested the uniformity loss for comparison. In Table 6, we present the results, where we additionally reported the area under the precision-recall curve (AUPR)

Table 5. Comparison with the previous methods in the second setting by using ResNet-18. The corresponding classification accuracy values are reported in the first column.

$\mathcal{D}_t/\mathcal{D}_{test}^k$	$\mathcal{D}_{test}^u$	AUROC ($\uparrow$)	OSCR ($\uparrow$)
		Objectosphere / Uniformity / Energy / Class-inclusion (Ours)	
CIFAR10 0.937 / 0.949 / 0.933 / <u>0.951</u>	ImageNet-CR	0.982 / 0.979 / 0.983 / <u>0.987</u>	0.932 / 0.928 / 0.917 / <u>0.941</u>
	ImageNet-RE	0.977 / 0.982 / 0.975 / <u>0.988</u>	0.918 / 0.934 / 0.909 / <u>0.945</u>
	LSUN-CR	0.991 / 0.984 / 0.987 / <u>0.993</u>	0.934 / 0.935 / 0.919 / <u>0.946</u>
	LSUN-RE	0.987 / 0.987 / 0.986 / <u>0.990</u>	0.932 / 0.938 / 0.918 / <u>0.942</u>
	iSUN	0.987 / 0.986 / 0.987 / <u>0.994</u>	0.932 / 0.938 / 0.919 / <u>0.946</u>
	Average	0.985 / 0.984 / 0.984 / <u>0.991</u>	0.930 / 0.935 / 0.916 / <u>0.944</u>

Table 6. Comparison with the previous BCR method in text classification experiments.

$\mathcal{D}_t/\mathcal{D}_{test}^k$	$\mathcal{D}_{test}^u$	AUROC ($\uparrow$)	AUPR ($\uparrow$)	FPR95 ($\downarrow$)	OSCR ($\uparrow$)
		Uniformity / Class-inclusion (Ours)			
20 Newsgroups 0.719 / <u>0.749</u>	Multi30k	0.997 / <u>0.997</u>	<u>0.998</u> / 0.997	<u>0.002</u> / 0.010	0.715 / <u>0.745</u>
	WMT16	<u>0.997</u> / 0.996	<u>0.997</u> / 0.995	<u>0.010</u> / 0.016	0.715 / <u>0.742</u>
	IMDB	0.805 / <u>0.999</u>	0.692 / <u>0.999</u>	0.367 / <u>0.003</u>	0.585 / <u>0.747</u>
	Average	0.933 / <u>0.997</u>	0.896 / <u>0.997</u>	0.126 / <u>0.010</u>	0.672 / <u>0.745</u>

and the false-positive rate at 95% true-positive rate (FPR95) measures. As it outperformed the uniformity loss in image classification tasks, our method also showed significantly better OSR accuracy in text classification. We provide more training details of our text classification models in our supplementary materials.

5   Concluding Remarks

In this paper, we propose a novel BCR method to train open-set classifiers that can provide robust OSR results with a simple inference process. By employing distance-based classifiers with the principle of LDA, we designed a novel class-inclusion loss based on the principle of probability of inclusion, which effectively limits the feature space of KKC data in a class-wise manner and then regularizes KUC samples to be located far away from the limited class-wise spaces. Through our extensive experiments, we present that our method can achieve robust UUC rejection performance, while maintaining high closed-set classification accuracy. As this paper aims to improve the reliability of modern DNN-based classifiers, we hope our work to enhance reliability and robustness in various classification applications by providing a novel methodology of handling UUC samples.

References

1. Bendale, A., Boult, T.E.: Towards open set deep networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1563–1572 (2016)
2. Boult, T.E., Cruz, S., Dhamija, A.R., Gunther, M., Henrydoss, J., Scheirer, W.J.: Learning and the unknown: Surveying steps toward open world recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 9801–9807 (2019)
3. Cevikalp, H., Uzun, B., Köpüklü, O., Ozturk, G.: Deep compact polyhedral conic classifier for open and closed set recognition. Pattern Recognition **119**, 108080 (2021)
4. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. In: The 2014 Conference on Empirical Methods in Natural Language Processing (2014)
5. Dhamija, A.R., Günther, M., Boult, T.: Reducing network agnostophobia. Advances in Neural Information Processing Systems **31**, 9157–9168 (2018)
6. Fang, Z., Lu, J., Liu, A., Liu, F., Zhang, G.: Learning bounds for open-set learning. In: International Conference on Machine Learning. pp. 3122–3132. PMLR (2021)
7. Geng, C., Huang, S.j., Chen, S.: Recent advances in open set recognition: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020)
8. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1026–1034 (2015)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
10. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. In: International Conference on Learning Representations (2017)
11. Hendrycks, D., Mazeika, M., Dietterich, T.: Deep anomaly detection with outlier exposure. In: International Conference on Learning Representations (2019)
12. Izmailov, P., Kirichenko, P., Finzi, M., Wilson, A.G.: Semi-supervised learning with normalizing flows. In: International Conference on Machine Learning. pp. 4615–4630 (2020)
13. Jain, L.P., Scheirer, W.J., Boult, T.E.: Multi-class open set recognition using probability of inclusion. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 393–409. Springer (2014)
14. Júnior, P.R.M., Boult, T.E., Wainer, J., Rocha, A.: Specialized support vector machines for open-set recognition. arXiv preprint arXiv:1606.03802 (2016)
15. Krizhevsky, A.: Learning multiple layers of features from tiny images (2009)
16. Le, Y., Yang, X.: Tiny imagenet visual recognition challenge (2015)
17. Lee, K., Lee, H., Lee, K., Shin, J.: Training confidence-calibrated classifiers for detecting out-of-distribution samples. In: International Conference on Learning Representations (2018)
18. Lee, K., Lee, K., Lee, H., Shin, J.: A simple unified framework for detecting out-of-distribution samples and adversarial attacks. Advances in Neural Information Processing Systems **31**, 7167–7177 (2018)

19. Li, Y., Vasconcelos, N.: Background data resampling for outlier-aware classification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13218–13227 (2020)
20. Liang, S., Li, Y., Srikant, R.: Enhancing the reliability of out-of-distribution image detection in neural networks. In: International Conference on Learning Representations (2018)
21. Liu, W., Wang, X., Owens, J., Li, Y.: Energy-based out-of-distribution detection. Advances in Neural Information Processing Systems **33** (2020)
22. Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983 (2016)
23. Mensink, T., Verbeek, J., Perronnin, F., Csurka, G.: Metric learning for large scale image classification: Generalizing to new classes at near-zero cost. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 488–501. Springer (2012)
24. Murphy, K.P.: Machine learning: a probabilistic perspective. MIT press (2012)
25. Neal, L., Olson, M., Fern, X., Wong, W.K., Li, F.: Open set learning with counter-factual images. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 613–628 (2018)
26. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Reading digits in natural images with unsupervised feature learning (2011)
27. Oza, P., Patel, V.M.: C2ae: Class conditioned auto-encoder for open-set recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2307–2316 (2019)
28. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. (2019)
29. Rudd, E.M., Jain, L.P., Scheirer, W.J., Boult, T.E.: The extreme value machine. IEEE Transactions on Pattern Analysis and Machine Intelligence **40**(3), 762–768 (2018)
30. Ruff, L., Vandermeulen, R.A., Franks, B.J., Müller, K.R., Kloft, M.: Rethinking assumptions in deep anomaly detection. arXiv preprint arXiv:2006.00339 (2020)
31. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV) **115**(3), 211–252 (2015). https://doi.org/10.1007/s11263-015-0816-y
32. Scheirer, W.J.: Extreme value theory-based methods for visual recognition. Synthesis Lectures on Computer Vision **7**(1), 1–131 (2017)
33. Scheirer, W.J., Jain, L.P., Boult, T.E.: Probability models for open set recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**(11), 2317–2324 (2014)
34. Scheirer, W.J., de Rezende Rocha, A., Sapkota, A., Boult, T.E.: Toward open set recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(7), 1757–1772 (2013)
35. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 815–823 (2015)

36. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In: Advances in Neural Information Processing Systems. vol. 30, pp. 4080–4090 (2017)
37. Sun, X., Yang, Z., Zhang, C., Ling, K.V., Peng, G.: Conditional gaussian distribution learning for open set recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13480–13489 (2020)
38. Wen, Y., Zhang, K., Li, Z., Qiao, Y.: A comprehensive study on center loss for deep face recognition. International Journal of Computer Vision 127(6), 668–683 (2019)
39. Xu, P., Ehinger, K.A., Zhang, Y., Finkelstein, A., Kulkarni, S.R., Xiao, J.: Turkergaze: Crowdsourcing saliency with webcam based eye tracking. arXiv preprint arXiv:1504.06755 (2015)
40. Yang, H.M., Zhang, X.Y., Yin, F., Yang, Q., Liu, C.L.: Convolutional prototype network for open set recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020)
41. Yoshihashi, R., Shao, W., Kawakami, R., You, S., Iida, M., Naemura, T.: Classification-reconstruction learning for open-set recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4016–4025 (2019)
42. Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., Xiao, J.: Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. arXiv preprint arXiv:1506.03365 (2015)
43. Zagoruyko, S., Komodakis, N.: Wide residual networks. arXiv preprint arXiv:1605.07146 (2016)