Few-Shot Open-Set Recognition using Meta-Learning

Bo Liu
UC, San Diego
boliu@ucsd.edu

Hao Kang
Wormpex AI Research
haokheseri@gmail.com

Haoxiang Li
Wormpex AI Research
lhxustcer@gmail.com

Gang Hua
Wormpex AI Research
ganghua@gmail.com

Nuno Vasconcelos
UC, San Diego
nuno@ece.ucsd.edu

Abstract

The problem of open-set recognition is considered. While previous approaches only consider this problem in the context of large-scale classifier training, we seek a unified solution for this and the low-shot classification setting. It is argued that the classic softmax classifier is a poor solution for open-set recognition, since it tends to overfit on the training classes. Randomization is then proposed as a solution to this problem. This suggests the use of meta-learning techniques, commonly used for few-shot classification, for the solution of open-set recognition. A new oPen sEt mEta LEaRning (PEELER) algorithm is then introduced. This combines the random selection of a set of novel classes per episode, a loss that maximizes the posterior entropy for examples of those classes, and a new metric learning formulation based on the Mahalanobis distance. Experimental results show that PEELER achieves state of the art open set recognition performance for both few-shot and large-scale recognition. On CIFAR and miniImageNet, it achieves substantial gains in seen/unseen class detection AUROC for a given seen-class classification accuracy.

1. Introduction

The introduction of deep convolutional neural networks (CNNs) has catalyzed large advances in computer vision. Most of these advances can be traced back to advances on object recognition, due to the introduction of large scale datasets, such as ImageNet [2], containing many classes and many examples per class. Many modern computer vision architectures are entirely or partially based on recognition networks. In the large scale setting, CNN-based classifiers trained by cross-entropy loss and mini-batch SGD have excellent recognition performance, achieving state of the art results on most recognition benchmarks.

However, variations of the recognition setting can lead to substantial performance degradation. Well known examples include few-shot learning [25, 3, 33, 29], where only a few training examples are available per class, domain adaptation [32], where a classifier trained on a source image domain (e.g. synthetic images) must be deployed to a target domain (e.g. natural images) whose statistics differ from those of the source, long tailed recognition [18], where the number of examples per class is highly unbalanced, or problems with label noise [20, 30]. All these *alternative recognition settings* stress the robustness, or ability to generalize, of the large-scale classifier .

Typical Recognition Paradigms	No. of Samples per Training Class	Supports unseen Class in Testing?
Closed-set [13]	Large	No
Few-shot [25, 3, 33, 29]	Small	No
Open-set [5, 1]	Large	Yes
Few-Shot Open-set	Small	Yes

Table 1. Comparison between different recognition tasks.

More recently, there has been interest in endowing CNNs with self-awareness capabilities. At the core, self-awareness implies that, like humans, CNNs should identify what they can do and refuse what they cannot. Several variants of this problem have been proposed, including out-of-distribution detection [9], where the CNN rejects examples outside the training distribution (e.g. images from other domains or adversarial attacks [11]), realistic classification [19], where it rejects examples that it deems too hard to classify, or open-set recognition [1], where the examples to reject are from novel classes, unseen during training. While several techniques have been proposed, a popular approach is to force the CNN to produce high entropy posterior distributions in the rejection regions. Rejections can then be identified as examples that give rise to such distributions.

In this work, we consider open-set recognition. This has been mostly addressed in the large-scale setting, using so-

lutions based on the large-scale classifier . These attempt to recognize the novel classes by post-processing the posterior class distribution [1], defining a "reject" class trained with examples that are either artificially generated [5] or sampled from the training classes [28], or a combination of the two. We seek to generalize open-set recognition beyond the large-scale setting, proposing that it should target a continuum of tasks that includes all the alternative recognition settings. Since training over many settings can be complex, we consider the two at the extremes of this continuum: large-scale and few-shot recognition.

There are three main reasons for this. First, open-set recognition is a challenge under all settings. A recognizer trained in the few-shot regime is not less likely to face unseen classes. An open-set recognition technique that also supports the few-shot setting is thus more useful than the one that does not. Second, few-shot open-set recognition is harder to solve than large-scale open-set recognition, due to the scarcity of labeled data. Hence, the few-shot setting poses a greater challenge to open-set recognition research. Third, like open-set recognition, the main challenge of few-shot recognition is to make accurate decisions for data unseen during training. Since this makes robustness the main trait of few-shot architectures, these architectures are likely to also excel at open-set recognition. Most notably, they are likely to beat the large-scale classifier, which tends not to score highly along the robustness dimension.

The main limitation of the large-scale classifier (trained by cross-entropy) is that it tends to overfit the training classes. Because the ideal embedding for classification maps all examples of each class into a unique point in feature space, the large-scale classifier tends to learn embeddings that are only locally accurate. While the metric structure of the embedding is reflected by the semantic distances in the neighborhood of class representatives (the parameter vectors of the softmax layer), these distances are meaningless away from the latter. In result, large-scale classifier embeddings underperform on tasks that require generalization beyond the training set, such as image retrieval [7], face identification [17], pose invariant recognition [10], person re-identification [38], or few-shot learning [33]. Embeddings derived from the metric learning literature [37] or explicitly designed for problems like few-shot recognition [36] tend to capture the metric structure of the data more broadly throughout the feature space and achieve better performance on these generalization critical tasks. It is thus expected that these embeddings will outperform the large-scale classifier on the open set recognition problem.

In this work, we investigate the open-set performance of a class of popular solutions to the few-shot problem, known as meta-learning (ML) [33]. ML approaches replace the traditional mini-batch CNN training by episodic training. Episodes are generated by randomly sampling a subset of the classes and a subset of examples per class, to produce a support and query set to which a classification loss is applied. This randomizes the classification task for which the embedding is optimized at each ML step, producing a more robust embedding, less overfitted to any particular set of classes. We generalize the idea to open-set recognition, by randomly selecting a set of novel classes per episode, and introducing a loss that maximizes the posterior entropy for examples of these classes. This forces the embedding to better account for unseen classes, producing high-entropy posterior distributions beyond the regions of the target classes.

This solution has at least two strong benefits. First, because it draws on a state of the art approach to few shot learning, it immediately extends open set recognition to few-shot problems. Second, because ML embeddings are robust and open set recognition is about generalization, the performance of the latter improves *even in the large-scale setting*. This is shown by extensive experiments with various open-set recognition benchmarks, where we demonstrate significant improvements over the state of the art. We also investigate the role of the metric used in the feature space on open-set recognition performance, showing that a particular form of the Mahalanobis distance enables significant gains over the commonly used Euclidean distance.

Overall, the contributions of this work can be summarized as follows:

- A new ML-based formulation of open-set recognition. This generalizes open-set to the *few-shot* recognition setting.
- A new episodic training procedure, combining the cross-entropy loss and a novel *open-set loss* to improve open-set performance on both the large-scale and few-shot settings.
- A *Gaussian embedding* for ML-based open-set recognition.

2. Related Work

Open-Set Recognition: Open-set recognition addresses the classification setting where inference can face samples from classes unseen during training. The goal is to endow the open-set classifier with a mechanism to reject such samples. One of the first deep learning approaches was the work of Scheirer et al. [1], which proposed an extreme value parameter redistribution method for the logits generated by the classifier. Later works considered the problem in either discriminative or generative models. Schlachter et al. [28] proposed an intra-class splitting method, where a closed-set classifier is used to split data into typical and atypical subsets, reformulating open-set recognition as a traditional classification problem. G-OpenMax [5] utilizes a generator trained to synthesize examples from an extra class

that represents all unknown classes. Neal et al. [21] introduced counterfactual image generation, which aims to generate samples that cannot be classified into any of the seen classes, producing an extra class for classifier training.

All these methods reduce the set of unseen classes to one extra class. While open samples can be drawn from different categories and have significant visual differences, they assume that a feature extractor can map them all into a single feature space cluster. While theoretically possible, this is difficult. Instead, we allow a cluster per seen class and label samples that do not fall into these clusters as unseen. We believe this is a more natural way to detect unseen classes.

Out-of-Distribution: A similar problem to open-set recognition is to detect out-of-distribution (OOD) examples. Typically [9], this is formulated as the detection of samples from a different dataset, *i.e.* a different distribution from that used to train the model. While [9] solves this problem by directly using the softmax score, later works [16, 34] improve results by enhancing reliability. This problem differs from open-set recognition in that the OOD samples are not necessarily from unseen classes. For examples, they could be perturbed versions of samples from seen classes, as is common in the adversarial attack literature. The only constraint is that they do not belong to the training distribution. Frequently, these samples are easier to detect than samples from unseen classes. In the literature, they tend to be classes from other datasets or even images of noise. This is unlike open-set recognition, where unseen classes tend to come from the same dataset.

Few-Shot Learning: Extensive research on few-shot learning [25, 3, 33, 29, 31, 26, 6, 22, 15] has emerged in recent years. These methods can be broadly divided into two branches: optimization and metric based. Optimization based methods deal with the generalization problem by unrolling the back-propagation procedure. Specifically, Ravi et al. [25] proposed a learner module that is trained to update to novel tasks. MAML [3] and its variants [4] proposed a training procedure, where the parameters are updated based on the loss calculated by secondary gradients. Metric based approaches attempt to compare feature similarity between support and query samples. Vinyals et al. [33] introduced the concept of episode training, where the training procedure is designed to mimic the test phase, which was used with a cosine distance to train recurrent networks. The prototypical network [29] introduced class prototypes constructed from support set features, by combining metric learning and cross-entropy loss. The relation network [31] explored the relationship between a pair of support and query features implicitly with a neural network, instead of building a metric directly on feature space.

Since a feature space metric is also useful for open-set recognition, we mainly focus on metric based approaches to few-shot learning. Although several methods have demonstrated promising results for the few-shot task, it is still unclear whether the resulting classifiers can successfully reject unseen samples. In this work, we explore this problem under both the large-scale and few-shot scenarios.

Learning without Forgetting: Several works [6, 23, 24, 27] extend traditional few-shot to learning without forgetting. The model is trained to work on an extra few-shot problem and maintains its performance on the original recognition problem. Our work will focus on the traditional few-shot problem, this direction is beyond our scope, and can be discussed in the future.

3. Classification tasks and meta Learning

In this section, we discuss different classification settings (Sec. 3.1) and meta learning (Sect. 3.2), which motivates and provides the foundation for the proposed solution to open-set recognition in both the large-scale and few-shot settings.

3.1. Classification settings

Softmax classification. The most popular deep learning architecture for object recognition and image classification is the softmax classifier. This consists of an embedding that maps images $\mathbf{x} \in \mathcal{X}$ into feature vectors $f_\phi(\mathbf{x}) \in \mathcal{F}$, implemented by multiple neural network layers, and a softmax layer with linear mapping that estimates class posterior probabilities with

$$p(y = k|\mathbf{x}; \phi, \mathbf{w}_k) = \frac{\exp(\mathbf{w}_k^T f_\phi(\mathbf{x}))}{\sum_{k'} \exp(\mathbf{w}_{k'}^T f_\phi(\mathbf{x}))} \quad (1)$$

where ϕ denotes all the embedding parameters and $\mathbf{w}_k$ is a set of classifier weight vectors. Recent works [29] have combined metric learning with softmax classification, implementing the softmax layer with a distance function

$$p_\phi(y = k|\mathbf{x}) = \frac{\exp(-d(f_\phi(\mathbf{x}), \mu_k))}{\sum_{k'} \exp(-d(f_\phi(\mathbf{x}), \mu_{k'}))} \quad (2)$$

where $\mu_k = E[f_\phi(\mathbf{x})|y = k]$ [29]. The softmax classifier is learned with a training set $\mathbb{S} = \{(x_i^s, y_i^s)\}_{i=1}^{n^s}$, where $y_i^s \in \mathbb{C}^s, \forall i$, $\mathbb{C}^s$ is a set of training image classes, and n^s the number of training examples. This consists of finding the classifier and embedding parameters that minimize the cross entropy loss

$$\mathcal{L}_{CE} = \sum_{(x_i^s, y_i^s)} -\log p(y_i^s|\mathbf{x}_i^s) \quad (3)$$

Recognition performance is evaluated on a test set $\mathbb{T} = \{(x_i^t, y_i^t)\}_{i=1}^{n^t}$, where $y_i^t \in \mathbb{C}^t, \forall i$, $\mathbb{C}^t$ is a set of test classes, and n^s the number of test examples.

Closed vs. open set classification. Under the traditional definition of classification, the sets of training and test

classes are identical, i.e. $\mathbb{C}^s = \mathbb{C}^t$. This is denoted as *closed set classification*. Recently, there has been interest in an alternative *open set* classification setting, where $\mathbb{C}^t = \mathbb{C}^s \cup \mathbb{C}^u$. In this case, the classes in $\mathbb{C}^s$ are denoted as *seen* (during training) classes and the classes in $\mathbb{C}^u$ as *unseen*.

Large-scale vs few-shot recognition. In the large-scale recognition setting, the number n^s of training examples is quite high, reaching the millions for datasets like ImageNet. On the contrary, for few-shot recognition, this number is quite low, usually less than twenty examples per class. A few-shot problem with K training examples per class is commonly known as K-shot recognition. Note that the number n^t of test examples plays no role in differentiating these two settings. Since these examples are only used for performance evaluation, the test set should have the same cardinality under the two settings. As shown in Table 1, different combinations of properties in the scale and coverage of the training data define different classification paradigms. The few-shot open-set setting is largely unexplored in the literature and the focus of this paper.

3.2. Meta-learning

Meta-learning (ML) addresses the problem of "learning to learn". In this case, a meta-learner learns a learning algorithm by inspection of many learning problems. For this, the meta-learner relies on a meta training set $\mathbb{MS} = \{(\mathbb{S}_i^s, \mathbb{T}_i^s)\}_{i=1}^{N^s}$, where $(\mathbb{S}_i^s, \mathbb{T}_i^s)$ are the training and test set of the i^{th} learning problem and N^s the number of learning problems used for training; and a meta test set $\mathbb{MT} = \{(\mathbb{S}_i^t, \mathbb{T}_i^t)\}_{i=1}^{N^t}$, where $(\mathbb{S}_i^t, \mathbb{T}_i^t)$ are the training and test set of the i^{th} test learning problem and N^t is the number of learning problems used for testing. Given $\mathbb{MS}$, the meta-learner learns how to map a pair $(\mathbb{S}, \mathbb{T})$ into an algorithm that leverages $\mathbb{S}$ to optimally solve $\mathbb{T}$.

The procedures are as follows. At meta iteration i, a meta-model h is initialized with the one produced by the previous meta-iteration. Two steps are then performed. First, the meta-learning algorithm performs the mapping

$$h' = \mathcal{M}(h, \mathbb{S}_i^s) \tag{4}$$

to produce an estimate h' of the optimal model for training set $\mathbb{S}_i^s$. The test set $\mathbb{T}_i^s$ is then used to find the model

$$h^* = \arg\min_h \sum_{(x_k, y_k) \in \mathbb{T}_i^s} L[y_k, h'(x_k)] \tag{5}$$

for a suitable loss function L, e.g. cross-entropy, using a suitable optimization procedure, e.g. back-propagation. The resulting h^* is finally returned as the optimal meta-model for meta-iteration i. During testing, the final meta-model h^* and a training set $\mathbb{S}_i^t$ from the meta test set $\mathbb{MT}$ are used by the meta-learner to produce a new model

$$h'' = \mathcal{M}(h^*, \mathbb{S}_i^t) \tag{6}$$

whose performance is evaluated with $\mathbb{T}_i^t$.

ML for few-shot recognition. While different approaches have been proposed to apply ML to few shot recognition, in this work we adopt the procedure introduced by the popular prototypical network architecture [29], which is the foundation of various other approaches. In this context, ML is mostly a randomization procedure. The pairs $(\mathbb{S}_i, \mathbb{T}_i)$ are denoted *episodes*, the training sets $\mathbb{S}_i$ are denoted *support sets* and the test sets $\mathbb{T}_i$ *query sets*. The meta-training set $\mathbb{MS}$ is generated by sampling train and test classes. In particular, the training set $\mathbb{S}_i^s$ of the i^{th} episode is obtained by sampling N classes from the set of classes of the low-shot problem, and K-examples per class. This defines a set of K-shot learning problems and is known as the *N-way K shot problem*. The model h is the softmax classifier of (2), the meta-learning mapping of (4) implements the Gaussian mean maximum likelihood estimator

$$\mu_k = \frac{1}{|\mathcal{P}_{i,k}|} \sum_{\mathbf{x}_j \in \mathcal{P}_{i,k}} f_\phi(\mathbf{x}_j) \tag{7}$$

where $\mathcal{P}_{i,k} = \{\mathbf{x}_j \in \mathbb{S}_i^s | y_j = k)$ is the set of support samples from class k, and back-propagation is used in (5) to update the embedding f_ϕ.

Benefits of ML for open-set classification. The ML procedure is quite similar to the standard mini-batch procedure for training softmax classifiers. Aside from the use of episode training instead of the more popular mini-batch training, there are two main differences. First, f_ϕ is updated by back-propagation from examples in $\mathbb{T}_i^s$ rather than $\mathbb{S}_i^s$. This is advantageous for few-shot learning due to the consistency with meta-testing, which is not the case in the large-scale regime. Second, mini-batches of random examples from all classes are replaced by examples from the subset of N classes contained in $\mathbb{S}_i^s$ and $\mathbb{T}_i^s$. Randomizing the classification task learned per episode forces the embedding f_ϕ to generalize better to unseen data. This property makes ML a good solution for few-shot learning, where the training data lacks a good coverage of intra-class variations. In this case, large subsets of test samples from the known classes are unseen during training. We propose that the same property makes ML a good candidate for open-set classification, where by definition the classifier must deal with unseen samples from unseen classes. This observation motivates us to design a unified ML-based solution to open-set classification that supports both large-scale and few-shot classification.

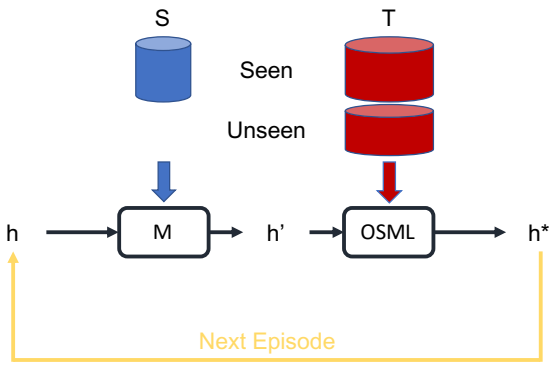


Figure 1. The proposed general framework for open-set meta-learning: a meta training set $\{\mathbb{S}, \mathbb{T}\}$ is sampled, where $\mathbb{T}$ contains classes not in $\mathbb{S}$ as "unseen" classes, and loss in (8) is minimized to obtain h^*.

4. Meta Learning-based open-set recognition

In this section, we introduce the proposed ML approach to open-set recognition. We first introduce the general procedure and then discuss the specific embedding metric used in our PEELER implementation.

4.1. Open-Set Meta-Learning

As shown in Figure 1, open-set meta-learning (PEELER) relies on a meta training set $\mathbb{MS} = \{(\mathbb{S}_i^s, \mathbb{T}_i^s)\}_{i=1}^{N^s}$, and a meta test set $\mathbb{MT} = \{(\mathbb{S}_i^t, \mathbb{T}_i^t)\}_{i=1}^{N^t}$. The only difference with respect to standard ML is that the episodes $(\mathbb{S}, \mathbb{T})$ are open set. While the training set $\mathbb{S}$ is identical to the one used in standard ML, the test set $\mathbb{T}$ is *augmented with unseen classes*. Hence, PEELER can be addressed with a solution similar to the ML procedure of Section 3.2. While the ML step remains as in (4), the optimization step of (5) becomes

$$h^* = \arg\min_h \left\{ \sum_{(x_k,y_k)\in\mathbb{T}_i^s | y_k\in\mathbb{C}_i^s} L_c[y_k, h'(x_k)] + \lambda \sum_{(x_k,y_k)\in\mathbb{T}_i^s | y_k\in\mathbb{C}_i^u} L_o[h'(x_k)] \right\} \quad (8)$$

where $\mathbb{C}_i^s$ ($\mathbb{C}_i^u$) is the set of seen (unseen) classes of $\mathbb{T}_i^s$, $L_c[.,.]$ is the classification loss to apply to the seen classes (typically the cross-entropy loss) and L_o is an open-set loss applied to those unseen in $\mathbb{S}_i^s$.

Few-shot open-set recognition. The few-shot setting requires the sampling of the classes that make up the support and query sets. Similarly to closed-set few-shot recognition, the support set $\mathbb{S}_i^s$ of the i^{th} episode is obtained by sampling N classes and K-examples per class. This defines the seen classes $\mathbb{C}_i^s$. However, the query set $\mathbb{T}_i^s$ is composed by the combination of these classes with M additional unseen classes $\mathbb{C}_i^u$. These support and query sets are used in (8).

Large-scale open-set recognition. In the large-scale setting, the seen classes have plenty of examples and are well trained without any need for resampling. However, resampling of the unseen classes is still advantageous, since it enables an embedding that generalizes better to these classes. In each episode, M classes are randomly sampled from the class label space to form the set of unseen classes $\mathbb{C}_i^u$, and the remaining classes are used as seen classes $\mathbb{C}_i^s$. Without a support set $\mathbb{S}_i^s$, the meta-learning step of (4) is no longer needed. Instead, we rely on the set of seen classes to adjust the model to only classify samples into those classes, *i.e.* a mapping

$$h' = \mathcal{M}(h, \mathbb{C}_i^s), \quad (9)$$

with the loss function of (8) is still applied.

Open-set loss. During inference, when faced with samples from unseen classes, the model should not assign a large probability to any class. In this case, a sample can be rejected if the largest class probability $\max_k p_\phi(y = k|\mathbf{x})$ among seen classes is small. To enable this, the learning algorithm should minimize the probabilities on seen classes for samples from $\mathbb{C}_i^u$. This can be implemented by maximizing the entropy of seen class probabilities, i.e using the negative entropy

$$L_o[\mathbf{x}] = \sum_{k\in\mathbb{C}_i^s} p(y = k|\mathbf{x}) \log p(y = k|\mathbf{x}) \quad (10)$$

as loss function.

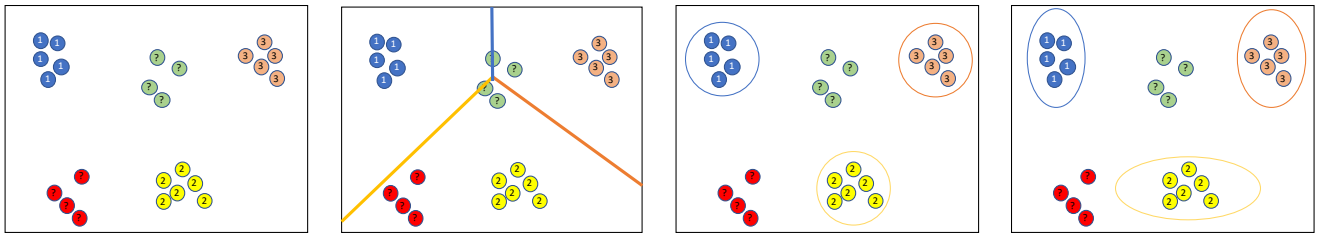
4.2. Gaussian Embedding

While PEELER is a general framework, in this work we propose an implementation based on the prototypical network architecture [29]. A set of class prototypes is first defined and samples with a large distance to the closest class prototype are assigned to the set of unseen classes. For low-shot classification, the class prototypes are the class means of (7). For large-scale classification, we assume fixed prototypes that are embedded in the network and learned by back-propagation.

Although several distances are discussed in [29], the prototypical network is usually implemented with the Euclidean distance, *i.e.*,

$$d(f_\phi(\mathbf{x}), \mu_k) = (f_\phi(\mathbf{x}) - \mu_k)^T (f_\phi(\mathbf{x}) - \mu_k). \quad (11)$$

This implies that the features from each class follow a Gaussian distribution with mean $\mathbf{c}_k$ and diagonal covariance $\sigma^2 I$, where σ is shared by all classes. While sensible for closed-set few-shot learning, where the embedding is learned to produced such feature distributions, this can be very sub-optimal when open-set samples are introduced. Figure 2 illustrates the problem for a setting with three seen classes and two unseen classes. Even though, as shown in Figure 2



(a) (b) (c) (d)

Figure 2. (a) Feature space of an open-set recognition problem with 3 seen and 2 unseen classes; (b) Optimal boundaries for closed-set classification; (c) Each class defines an euclidean distance of equal radius. While optimal for closed-set, this is sub-optimal for open set recognition; (d) a better set of open-set distances can be learned by allowing different Gaussian clusters of class-dependent variance along each dimension.

(a), the embedding is such that seen classes are spherically distributed and have the same covariance, the open-set samples, unseen during training, are still embedded onto the feature space at random locations. Hence, although the optimal boundaries of the closed-set classifier, shown in Figure 2 (b), match the contours of the distance of (11), shown in Figure 2 (c), they are not optimal for open set recognition. In fact, as shown in Figure 2 (d), the shape of the optimal boundary between seen and unseen classes can even vary from one class to another.

To account for this, we assume a Gaussian distribution of mean μ_k and covariance Σ_k for class k. The Euclidean distance of (11) is thus replaced by the Mahalanobis distance

$$d(f_\phi(\mathbf{x}), \mu_k) = [f_\phi(\mathbf{x}) - \mu_k]^T \Sigma_k^{-1} [f_\phi(\mathbf{x}) - \mu_k]. \quad (12)$$

To keep the number of parameters manageable, we assume that all covariance matrices are diagonal, *i.e.* $\Sigma_k = \text{diag}(\sigma_{k1}, \sigma_{k2}, \ldots, \sigma_{kM})$, and the precision matrix $A_k = \Sigma_k^{-1}$ is used to ease calculations. Similar to the class prototypes, the learning of precision matrices depends on the recognition setting. For large-scale open-set recognition, the A_k are network parameters learned directly by back-propagation. In the few-shot setting, where support samples are available, we introduce a new embedding function g_φ with learnable parameters φ, and define

$$A_k = \frac{1}{|S_k|} \sum_{(\mathbf{x}_i, y_i) \in S_k} g_\varphi(\mathbf{x}_i). \quad (13)$$

5. Experiments

The proposed method was compared to state-of-the-art (SOTA) approached to open-set recognition. Following Neal et al. [21], we evaluate both classification accuracy and open-set detection performance. To clarify the terminology, closed-set classes are the categories seen during training and open-set classes the novel categories only used for testing. All training is based on the training samples from closed-set classes. For testing, we use the testing samples from both the training and open-set classes. Classification *accuracy* is used to measure how well the model classifies closed-set samples, i.e., test samples from closed-set classes. The *AUROC* (Area Under ROC Curve) metric is used to measure how well the model detects open-set samples, i.e., test samples from open-set classes, within all test samples. To simplify the writing, we define the following acronyms: *our basic* represents prototypical networks with euclidean distance for open-set detection; *GaussianE* represents the Gaussian Embedding introduced in Sec. 4.2; *OpLoss* represents the proposed open-set loss.

5.1. Large-Scale Open-Set Recognition

Most prior works on open-set recognition are designed for the large-scale setting. Here we evaluate PEELER on CIFAR10 [13] and extended miniImageNet [6]. CIFAR10 consists of $60,000$ images of 10 classes. Following [21], 6 classes are first randomly selected as closed-set classes, and the other 4 kept as open-set classes. Results are averaged over 5 random partitions of closed/open-set classes. Extended miniImageNet is designed for few-shot learning. We use the 64 training categories and 600 images per category as closed set training data, while the 300 extra images of each category are used for closed set testing. Images from 20 test categories are used for open set testing.

Training. On CIFAR10, we randomly sample 2 from the 6 closed-set classes per episode to apply the open-set loss, the remaining 4 classes are used to train classification. On miniImageNet, the closed/open-set partition is fixed. We use Adam [12] with an initial learning rate of 0.001, $\lambda = 0.5$ in (8) and $10,000$ training episodes. The learning rate is decayed by a factor of 0.1 after $6,000$ and $8,000$ episodes.

Results. We compare the proposed method to several open-set recognition SOTA methods, including OpenMax [1], G-OpenMax [5], and Counterfactual [21], and an out-of-distribution SOTA method, Confidence [14]. All models use the same CNN backbone for fair comparison. We tested both a CNN, denoted as ConvNet, proposed by [21], and ResNet18 [8].

Table 2 shows that, for both backbones, PEELER outperforms all previous approaches by a significant margin. For a similar classification accuracy on seen classes, it achieves

Model	Accuracy(%)	AUROC(%)
ConvNet on CIFAR10		
Softmax	80.1	67.7
OpenMax	80.1	69.5
G-OpenMax	81.6	67.9
Counter	82.1	69.9
Confidence	82.4	73.19
Our basic	82.4	74.62
Our basic + GaussianE	82.3	75.65
Our basic + GaussianE + OpLoss	82.3	**77.22**
ResNet18 on CIFAR10		
Softmax	94.2	78.90
OpenMax	94.2	79.02
Confidence	94.0	80.90
Our basic	94.7	82.94
Our basic + GaussianE	94.3	83.12
Our basic + GaussianE + OpLoss	94.4	**83.99**
ResNet18 on miniImageNet		
Softmax	76.1	76.65
OpenMax	76.1	77.80
Confidence	76.5	80.67
Our basic	76.4	80.59
Our basic + GaussianE	76.1	81.06
Our basic + GaussianE + OpLoss	76.3	**82.12**

Table 2. Comparison to SOTAs on large-scale open-set recognition: PEELER outperforms all others in terms of open-set sample detection *AUROC*, for a comparable classification accuracy.

much higher AUROCs for the detection of unseen classes. The open-set detection performance is enhanced by both the proposed Gaussian embedding and open-set loss.

5.2. Few-Shot Open-Set Recognition

Dataset. Few-Shot Open-Set performance is evaluated on mini-Imagenet [33], using the splits of [25]. 64 classes are used for training, 16 for validation and another 20 for test.

Training. The open-set problem follows the 5-way few-shot recognition setting. During training, 10 classes are randomly selected from the training set per episode, 5 are used as closed-set classes and the other 5 as unseen. All support set samples are from the closed-set classes. The query set contains samples from closed-set classes, the closed-query set, and samples from open-set classes, the open-query set. The evaluation set is sampled from the test set with the same strategy. The evaluation is repeated 600 times to minimize uncertainty. The total number of training episodes is $30,000$. The learning rate is decreased by a factor of 0.1 after $10,000$ and $20,000$ episodes.

Results. Since not all prior open-set methods support the few-shot setting, some modifications were required. For example, generative methods [5, 21] do not support few-shot samples. Instead, we train the model on the pre-training set and fine-tune it on the support set. The closed-set classifier is as above, and a two-class classifier is further trained to detect open-set samples. OpenMax [1] is easier to apply under the few-shot setting. We apply OpenMax on the activations of the pre-softmax layer, *i.e.* the negative of the distance

Model	Accuracy(%)	AUROC(%)
5-way 1-shot		
GaussianE + OpenMax	57.89±0.59	58.92±0.59
GaussianE + Counterfactual	57.89±0.59	52.20±0.61
Our basic	56.31±0.57	58.94±0.60
Our basic + OpLoss	56.34±0.57	60.94±0.61
Our basic + GaussianE	57.89±0.59	58.66±0.60
Our basic + GaussianE + OpLoss	**58.31±0.58**	**61.66±0.62**
5-way 5-shot		
GaussianE + OpenMax	75.31±0.76	67.54±0.67
GaussianE + Counterfactual	75.31±0.76	53.25±0.59
Our basic	74.19±0.75	66.00±0.67
Our basic + OpLoss	74.14±0.74	67.92±0.68
Our basic + GaussianE	**75.31±0.76**	66.50±0.67
Our basic + GaussianE + OpLoss	75.08±0.72	**69.85±0.70**

Table 3. Few-shot open-set recognition results. Comparison to several baselines and prior open-set methods.

Model	Accuracy(%)	AUROC(%)
Our basic	39.61±0.40	71.32±0.70
Proto+Entropy	39.41±0.40	72.23±0.72
Our basic + GaussianE	40.18±0.40	71.31±0.70
Our basic + GaussianE + OpLoss	**41.90±0.39**	**74.97±0.74**

Table 4. 10-way 1-shot openset recognition results. The AUROC is higher than those in 5-way.

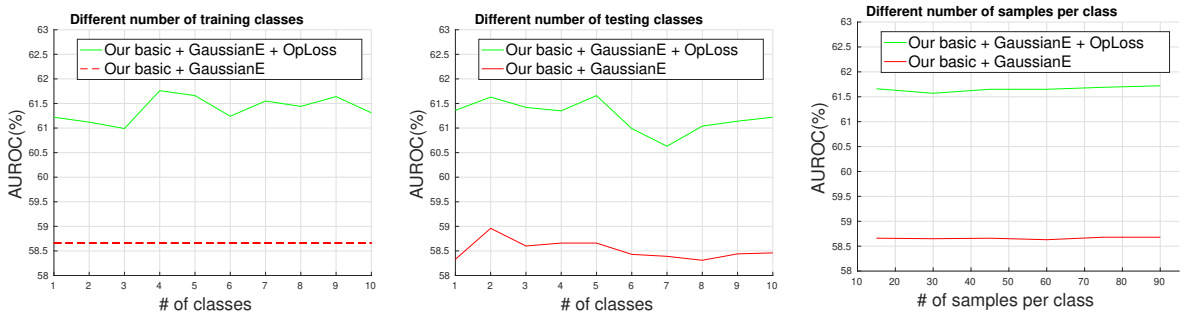
of our Gaussian setting. All methods are implemented with the ResNet18 [8] for fair comparison.

As shown in Table 3, both OpenMax and Counterfactual have weaker performance than the proposed approach. Note that a *AUROC* of 50% corresponds to chance performance. The proposed open-set loss, Gaussian embedding, and their combination all provide gains.

5.3. Ablation Studies

Sampling Strategy in Training. We study how the training sampling strategy affects open-set detection results. The number of unseen classes used to produce open-set samples by training episode is varied, for a fixed total number of open-set samples. The corresponding open-set detection results are shown in Figure 3 (a). The performance variation is minor, when compared to the average gain over the baseline method. We hypothesize that this robustness is due to task randomness. When the total number of training episodes is large, the model converges well no matter how many open-set classes are included in a single episode.

Factors in Open-set Testing. For large-scale open-set recognition, the number of open-set samples follows from the number of open-set classes. The difficulty of the open-set problem is determined by the latter. We try to discover factors that determine the difficulty of the few-shot open-set problem. A first factor is the number of classes in the open-query set. We vary this number from 1 to 10, while keeping the training procedure and the total number of samples in the open-query set unchanged. Results are shown in Figure 3 (b). A second factor is the number of samples per class, which is changed while the number of classes remains 5. Results are shown in Figure 3 (c). The figures



(a) (b) (c)

Figure 3. Ablation Study: the proposed open-set loss produces consistent gain with different (a) number of training classes, (b) number of testing classes, or (c) number of testing samples per class.

Category	VODC [35]			Ours		
	I	II	III	I	II	III
airplane	20	10	**0**	6	6	**0**
balloon	13	4	3	11	7	**2**
bear	3	3	**2**	**2**	**2**	**2**
cat	4	5	5	5	**3**	**3**
eagle	23	12	8	5	2	**0**
ferrari	11	7	6	11	**3**	**3**
figure skating	**0**	**0**	**0**	**0**	**0**	**0**
horse	5	**1**	**1**	5	4	3
parachute	14	10	2	10	2	**0**
single diving	18	13	5	4	2	**1**
Avg.	11.1	6.5	3.2	5.9	3.1	**1.4**

Table 5. The number of misclassified frames (lower is better) by varying the number of annotated frames (I, II, III in the second row).

show that variations cause small changes in AUROC performance. This means that the factors have little impact in the difficulty of the problem.

Factors in Few-Shot Testing. We compare 5-way to 10-way classification, for a fixed open-set component of both training and testing sets. Results are shown in Table 4. Although, as expected, 10-way underperforms 5-way classification, the open-set performance improves significantly. This shows that few-shot classification tasks with more class diversity are more robust to open-set samples.

5.4. Weakly Supervised Object Discovery

Finally, we investigate an application of few-shot open-set recognition. A weakly supervised object discovery problem is considered in [35]. A video of an object is given, but some frames are irrelevant for the presence of the object. The task is to find these irrelevant frames when the number of annotated frames (object presented or not) is limited. For the open-set problem, only relevant frame labels are given, *i.e.* frames containing the object. The irrelevant frames are detected as open-set samples.

The XJTU-Stevens Dataset [35] is adopted for evaluation. It has 101 videos from 10 categories with different instances, and some frames in those videos do not have the object. During training, a 5-way 1-shot few-shot open-set model is trained as described in Sec. 5.2. Instead of category level classification, we perform instance level classification of each frame. During testing, only one video is considered, using the frames annotated as relevant as the support set. This implies that only one Gaussian class center is provided. The unlabeled frames are labeled as seen or unseen by their Mahalanobis distance to the center. Frames with distance higher than a threshold are detected as irrelevant. The number of annotated relevant frames varies from 1 to 3. Results are shown as the number of misclassified frames. The best VODC method in [35] is listed for comparison. Note that VODC requires the same number of annotated irrelevant frames, which PEELER does not. The proposed method largely outperforms VODC, halving the number of misclassified frames in all three settings. This shows that its feature embedding is a better solution for open-set detection.

6. Conclusion

In this work, we have revisited the open-set recognition problem in the context of few-shot learning. We proposed an extension of meta-learning that includes an open-set loss, and a better metric learning design. The resulting classifiers provide a new stat-of-the-art for few-shot open-set recognition, on mini-Imagenet. It was shown that, with few modifications, the approach can also be applied to large-scale recognition, where it outperforms state of the art methods for open-set recognition. Finally, the XJTU-Stevens Dataset was used to demonstrante the effectiveness of the proposed model on a weakly supervised object discovery task.

Acknowledgement. Gang Hua was supported partly by National Key R&D Program of China Grant 2018AAA0101400 and NSFC Grant 61629301. Bo Liu and Nuno Vasconcelos were partially supported by NSF awards IIS-1637941, IIS-1924937, and NVIDIA GPU donations.

References

[1] Abhijit Bendale and Terrance E. Boult. Towards open set deep networks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem:1563–1572, 2016. 1, 2, 6, 7

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1

[3] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. 2017. 1, 3

[4] Chelsea Finn, Kelvin Xu, and Sergey Levine. Probabilistic Model-Agnostic Meta-Learning. (NeurIPS), 2018. 3

[5] Zongyuan Ge, Sergey Demyanov, and Rahil Garnavi. Generative OpenMax for Multi-Class Open Set Classification. 2019. 1, 2, 6, 7

[6] Spyros Gidaris and Nikos Komodakis. Dynamic Few-Shot Visual Learning Without Forgetting. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 4367–4375, 2018. 3, 6

[7] Albert Gordo, Jon Almazán, Jerome Revaud, and Diane Larlus. Deep image retrieval: Learning global representations for image search. In *European conference on computer vision*, pages 241–257. Springer, 2016. 2

[8] 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. 6, 7

[9] Dan Hendrycks and Kevin Gimpel. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. pages 1–12, 2016. 1, 3

[10] Chih-Hui Ho, Pedro Morgado, Amir Persekian, and Nuno Vasconcelos. Pies: Pose invariant embeddings. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12377–12386, 2019. 2

[11] Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, and Pieter Abbeel. Adversarial attacks on neural network policies. *arXiv preprint arXiv:1702.02284*, 2017. 1

[12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[13] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The cifar-10 dataset. *online: http://www. cs. toronto. edu/kriz/cifar. html*, 55, 2014. 1, 6

[14] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. *arXiv preprint arXiv:1711.09325*, 2017. 6

[15] Hongyang Li, David Eigen, Samuel Dodge, Matthew Zeiler, and Xiaogang Wang. Finding Task-Relevant Features for Few-Shot Learning by Category Traversal. 1, 2019. 3

[16] Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. (2017), 2017. 3

[17] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 212–220, 2017. 2

[18] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-Scale Long-Tailed Recognition in an Open World. pages 2537–2546, 2019. 1

[19] Jiajun Lu, Theerasit Issaranon, and David Forsyth. Safetynet: Detecting and rejecting adversarial examples robustly. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 446–454, 2017. 1

[20] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *Advances in neural information processing systems*, pages 1196–1204, 2013. 1

[21] Lawrence Neal, Matthew Olson, Xiaoli Fern, Weng Keen Wong, and Fuxin Li. Open set learning with counterfactual images. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11210 LNCS(1):620–635, 2018. 3, 6, 7

[22] Boris N Oreshkin, Pau Rodriguez, and Alexandre Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. In *Advances in Neural Information Processing Systems*, volume 2018-Decem, pages 721–731, 2018. 3

[23] Hang Qi, Matthew Brown, and David G. Lowe. Low-Shot Learning with Imprinted Weights. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 5822–5830, 2018. 3

[24] Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan Yuille. Few-Shot Image Recognition by Predicting Parameters from Activations. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 7229–7238, 2018. 3

[25] Sachin Ravi and Hugo Larochelle. Optimization as a Model For Few-Shot Learning. pages 1–11, 2017. 1, 3, 7

[26] Avinash Ravichandran, Rahul Bhotika, and Stefano Soatto. Few-Shot Learning with Embedded Class Models and Shot-Free Meta Training. 2019. 3

[27] Mengye Ren, Renjie Liao, Ethan Fetaya, and Richard S. Zemel. Incremental Few-Shot Learning with Attention Attractor Networks. 2018. 3

[28] Patrick Schlachter, Yiwen Liao, and Bin Yang. Open-Set Recognition Using Intra-Class Splitting. 2019. 2

[29] Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical Networks for Few-shot Learning. 2017. 1, 3, 4, 5

[30] Sainbayar Sukhbaatar and Rob Fergus. Learning from noisy labels with deep neural networks. *arXiv preprint arXiv:1406.2080*, 2(3):4, 2014. 1

[31] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H.S. Torr, and Timothy M. Hospedales. Learning to Compare: Relation Network for Few-Shot Learning. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1199–1208, 2018. 3

[32] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176, 2017. 1

[33] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching Networks for One Shot Learning. 2016. 1, 2, 3, 7

[34] Apoorv Vyas, Nataraj Jammalamadaka, Xia Zhu, Dipankar Das, Bharat Kaul, and Theodore L. Willke. Out-of-distribution detection using an ensemble of self supervised leave-out classifiers. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11212 LNCS:560–574, 2018. 3

[35] Le Wang, Gang Hua, Rahul Sukthankar, Jianru Xue, Zhenxing Niu, and Nanning Zheng. Video object discovery and co-segmentation with extremely weak supervision. *IEEE transactions on pattern analysis and machine intelligence*, 39(10):2074–2088, 2016. 8

[36] Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7278–7286, 2018. 2

[37] Lin Dahua Zhirong Wu, Yuanjun Xiong, Stella X. Yu. Unsupervised Feature Learning via Non-Parametric Instance-level Discrimination. (CVPR), 2018. 2

[38] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1318–1327, 2017. 2