Utilizing Patch-level Category Activation Patterns for Multiple Class Novelty Detection

Poojan Oza and Vishal M. Patel

Johns Hopkins University, 3400 N. Charles St, Baltimore, MD 21218, USA
{poza, vp36}@jhu.edu

Abstract. For any recognition system, the ability to identify novel class samples during inference is an important aspect of the system's robustness. This problem of detecting novel class samples during inference is commonly referred to as Multiple Class Novelty Detection. In this paper, we propose a novel method that makes deep convolutional neural networks robust to novel classes. Specifically, during training one branch performs traditional classification (referred to as global inference), and the other branch provides patch-level information to keep track of the class-specific activation patterns (referred to as local inference). Both global and local branch information are combined to train a novelty detection network, which is used during inference to identify novel classes. We evaluate the proposed method on four datasets (Caltech256, CUB-200, Stanford Dogs and FounderType-200) and show that the proposed method is able to identify novel class samples better compared to the other deep convolutional neural network-based methods.

Keywords: Multiple class novelty detection, class activation patterns.

1 Introduction

Improving the robustness of recognition models has been one of the primary research topics in computer vision and machine learning in recent years. Specifically, problems such as adversarial attacks [37,12,30,21,8,34], recognition bias [16,40,36], out-of-distribution detection [14,19,9], open-set recognition [2,23,26], outlier removal [42,43] and novelty/anomaly detection [28,5,20,32,1,25,3] have received tremendous interest. In this paper, we focus on one such aspect of robustness, referred to as multi-class novelty detection.

Typically, in a recognition problem, the goal is to learn a model that can identify the underlying features using data samples from a given set of classes (i.e., known classes). Later, these features can be used at inference stage to identify data samples into a given set of known classes. The problem arises when samples from novel classes (i.e. samples that do not belong to any of the known classes) are observed during inference. In this case, the network misidentifies the sample from a novel class as one of the known classes. Novelty detection was specifically introduced to address this issue [15,22,22,32,29,27]. Generally, a novelty detector attempts to identify whether a sample during inference is

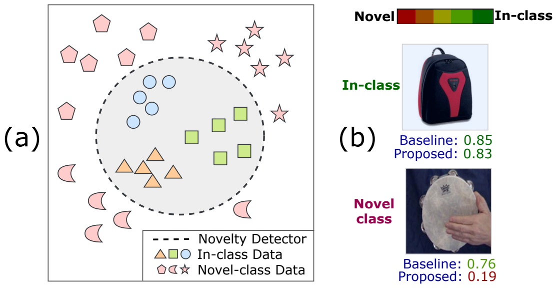

Fig. 1. (a) Typical example of a multiple class novelty detection scenario, where a novelty detector is used to differentiate between in-class and novel class data. (b) Baseline and the proposed method are able to produce high scores for in-class data. However, for novel class data the proposed approach does is better at assigning low scores compared to the baseline. Here, the "Baseline" refers to the novelty detection using traditional deep convolutional neural network with penultimate layer scores.

either from in-class (i.e, one of the known classes) or is from a novel class. When the number of known classes are more than one, the problem is also referred to as *multi-class novelty detection* [28,5,20,24]. When employed, the multi-class novelty detection module allows data samples only from known classes to pass through the recognition model, which results in increased robustness for the model. This is useful in many real-world vision applications. For example, in the case of autonomous navigation systems, it is important to stop and re-plan the navigation path by detecting a novel class as obstacle rather than misclassifying it and risking a potential crash.

The major challenge in developing multi-class novelty detector is the unavailability of novel class samples during training. Since the knowledge or data samples of novel classes are impossible to attain beforehand, the majority of novelty detection algorithms rely on how well they can encode the in-class data. There have been a few multi-class novelty detection methods proposed in recent years that try to overcome this challenge. Some of the earlier methods such as [5,20] use the feature encoding of in-class data to learn a subspace (referred to as null space of training data in [5,20]), and during inference the novelty score is calculated based on the distance of a test sample projected onto that subspace with the learned in-class data projections. However, these methods can not be integrated with deep convoluitonal neural networks (DCNN) to perform end-to-end training.

Recently, Perera and Patel [28] proposed a DCNN-based multi-class novelty detection method that can be trained in an end-to-end fashion. Specifically, to improve the novelty detection capability of a DCNN, they proposed a fine-tuning approach, where a *reference dataset* is used as a proxy for novel-class data. The authors argue that, since novelty detection methods often operate on features extracted from DCNN models which are pre-trained on the reference dataset, it would be beneficial, especially for novelty detection, to utilize the samples from a reference dataset as well. However, that argument does not always hold true. There are many cases where access to such reference datasets might not be possible. For example, consider a dataset having biometric information of users. Such datasets have high privacy risk associated with them and hence might not be available for public use. Additionally, for many private companies it is a competitive advantage to keep their datasets only for internal use, e.g., recently Google released state of the art neural network recognition models[1] trained on their internal datasets which are not publicly released. Also, in the case of Federated Learning [6] based applications, sharing dataset across devices is restricted to promote data privacy. However, in such scenario sharing a trained model parameters is possible as it contains very little risk on the privacy of the corresponding data. Hence, for the cases described above, it is not possible to access the reference dataset. Moreover, the reference dataset as described in [28] has to be fully labeled and hence can not be replaced by any randomly collected set of images. Ideally, we would want a novelty detection method that is flexible enough to work on scenarios where the reference dataset is not available, and when available it should be able to utilize the reference dataset to improve the novelty detection capability of the model.

In this paper, we propose a multiple class novelty detection to address the above mentioned concerns. Specifically, we use two parallel DCNN branches, where one branch learns features for identifying what class is present in the image and the second branch learns class-wise activations in the image patches. The information from both branches are combined in proposed training strategy to train a novelty detection network, without requiring a reference dataset. Moreover, to increase the flexibility of the approach, we also extend the method for the cases where reference dataset is available to further improve the performance. The advantage of this approach is that, as opposed to previous methods [28], it does not rely heavily on the availability of a reference dataset. We show that this proposed approach performs well on the novelty detection task compared to the other methods in the literature.

In summary, this paper makes the following contributions:

- We propose multiple class novelty detection approach, trained using a novel training strategy which utilizes both image-level and patch-level information.
- The proposed approach does not rely heavily on the availability of reference dataset, but when reference dataset is accessible, it can be easily extended to further improve the novelty detection performance.
- The performance is evaluated on four benchmark datasets and is shown to achieve improvements over several recent novelty detection methods.

2 Related Work

Over the years many novelty detection methods have been proposed some the earliest methods include principle component analysis-based [39,15], support vector machine-based [31,38], sparse representation-based [41,44], nearest neighbors-based [17,13,11]. In some of the recent works, Bodesheim *et al.* [5] proposed a

[1] github.com/tensorflow/tpu/tree/master/models/official/efficientnet

kernel-based method that projects all in-class data onto a subspace (referred to as null-space of training data), where all in-class categories are forced to have zero intra-class variance. Specifically, they employ a special case of linear discriminant analysis formulation, called Null-space Foley-Shannon Transform (NFST), to achieve zero intra-class variance. The smallest distance between the test sample projection with the class projections is used to decide whether an input is from a known class or a novel class. Liu *et al.* [20] pointed out that NFST training does not scale well with the increase in dataset size due to its high computation cost. To counter that, they proposed an incremental addition of classes to learn NFST subspace, which results in improved scalability with increased dataset size. Bodesheim *et al.* [4] proposed another variant of NFST-based novelty detection method which rather than using all in-class data samples, learns the NFST model based on the k nearest neighbor samples. This selective sampling helps to locate the local manifold on the feature space and learn specific models for each test sample.

However, all of these methods provide a general framework for novelty detection and none of them are specifically designed for DCNNs. Schultheiss *et al.* [32] proposed a DCNN-based novelty detection method by examining the extreme signatures observed in the penultimate layer. More precisely, depending on the input data there are specific dimensions in the penultimate layer of DCNNs, which produce high activation values (referred to as extreme value signatures) if the input is from novel class. Recently, Perera *et al.* [28] proposed a DCNN-based training method using a reference dataset. Instead of just utilizing pre-trained models trained on some reference dataset, they propose to use samples from the reference dataset as well. They show that having access to these additional data samples acts as a novel class proxy and benefits the novelty detection aspect of DCNNs. The reference dataset used during training, enables learning of negative filters which forces low activations at penultimate layer, when the input data is not from a novel class.

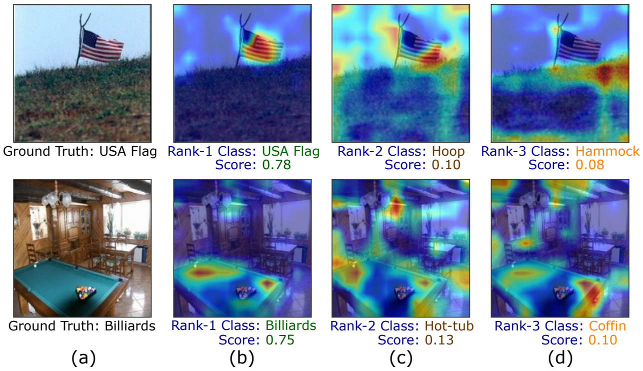

Fig. 2. (a) Original image with corresponding ground truth label. (b), (c) and (d) represent grad-cam visualizations for rank-1, rank-2 and rank-3 classes and predicted probability scores.

3 Proposed Approach

Deep convolutional neural networks have the ability to learn high-quality representations that are class-discriminative, making them the most successful tool for image recognition. These representations are learned by an end-to-end training and are computed by aggregating patch-level convolution responses (or activation maps) through non-linear activation functions and pooling process. Furthermore, these activation maps are aggregated depending on the strength of the activation to predict the probability scores for each class. The classes are ranked based on the predicted probability score and the class having the maximum score (i.e. rank-1 class) is predicted as the label. Fig. 2 illustrates this point with grad-cam [33] visualizations of top-3 classes. Here, the classes are ranked based on the predicted probability scores. The visualizations in Fig. 2 are not limited to top-3 classes and can be shown for all categories in the training set. This figure shows that given an image, a DCNN produces activation maps that has some contribution from all known classes.

For novel class test samples, none of the predictions would be correct, since the training set did not contain these classes. Furthermore, as shown in Fig. 3, often the rank-1 prediction scores for novel classes are very high, making it difficult for DCNNs to identify them as novel. However, looking at the examples shown in Fig. 3, one can notice that the patch-level activation patterns for both known class samples and novel class samples are different, even when both images are classified as the same class with high scores. The activation patterns for in-class (i.e. known class) samples are focused on the underlying object, whereas for novel class data the patterns are spread out across the image producing high activations at multiple image-patch locations. Given this information, we make an assumption that this type of discrepancy in the patch-level activation pattern exists across all novel class samples. Based on this assumption, we propose a nov-

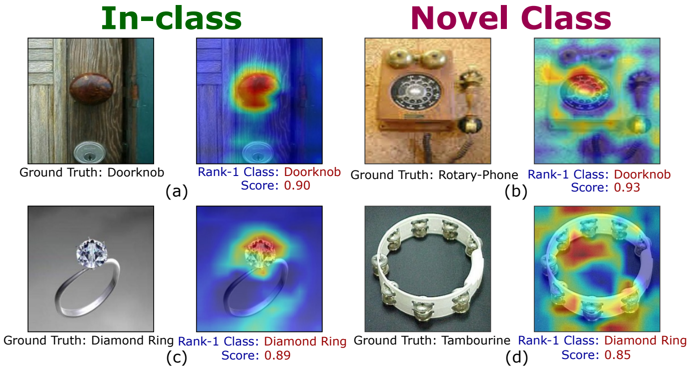

Fig. 3. (a)-(b) In-class samples from Doorknob and Diamond Ring classes with grad-cam visualizations and the predicted scores. (c)-(d) Novel class samples from Rotary-Phone and Tambourine are mis-classified as Doorknob and Diamond Ring as shown with grad-cam visualizations and predicted scores.

elty detection algorithm that learns to detect novel class samples by identifying discrepancy in the patch-level activation patterns.

Typically, for multi-class novelty detection we have access to only in-class data samples, $\{x_i, y_i\}_{i=1}^{i=n}$, where $y_i \in \{1, 2, ..., K\}$ is the class label corresponding to the data point x_i, n is the total number of data samples and K is the total number of classes. In the following subsections, we provide details of the proposed novelty detection method.

3.1 Global Inference Network

The global inference network can be decomposed in to two parts, feature extractor ($\mathcal{G}$) and classifier ($\mathcal{C}$). The feature extractor ($\mathcal{G}$), processes the image through stacked convolutional, pooling and activation layers to produce a global feature encoding of the object present in the image, as shown in Fig. 4(a). The classifier ($\mathcal{C}$), uses this global feature encoding to classify the image into one of K classes. The cross entropy loss used to train such network can be defined as follows

$$\mathcal{L}_{global} \ = \ \frac{1}{n}\sum_{i=1}^{n} \ L_{ce}(\mathcal{C}(\mathcal{G}(x_i)), \ y_i), \tag{1}$$

where y_i is the ground truth class label for the input x_i, n is total number of images from known classes and $\mathcal{C}(\mathcal{G}(x_i))$ is the predicted probability vector.

3.2 Local Inference Network

For local inference, the network needs to process individual image patches and provide predictions at patch-level as opposed to the global inference network where the predictions are provided at the image level. To achieve this, we utilize a recently proposed BagNet architecture [7] as local inference network. Specifically, BagNet processes the input image using a series of convolutional layers with 1×1 convolutions and 3×3 convolutions. The limiting of receptive field size restricts the network to perform patch-level processing and produce patch-level feature encodings. These patch-level encodings are used to produce patch-level prediction scores for all K classes, here referred to as local feature encodings. All these predictions are average pooled to produce the final prediction score, which is trained using the cross entropy loss in an end-to-end fashion. This process is illustrated in Fig. 4(b). The local feature encodings provide us with information regarding what each image-patch corresponds to and also the details regarding patch-level activation patterns for a particular class. This information is particularly useful in our approach and is utilized in the next section to train the novelty detection network. The local inference network is trained using the following loss function

$$\mathcal{L}_{local} \ = \ \frac{1}{n}\sum_{i=1}^{n} \ L_{ce}(gap(\mathcal{R}(x_i)), \ y_i), \tag{2}$$

where $\mathcal{R}$ denotes the local inference network, $\mathcal{R}(x_i)$ denotes the prediction map having all patch-level prediction scores corresponding to all K classes and *gap*

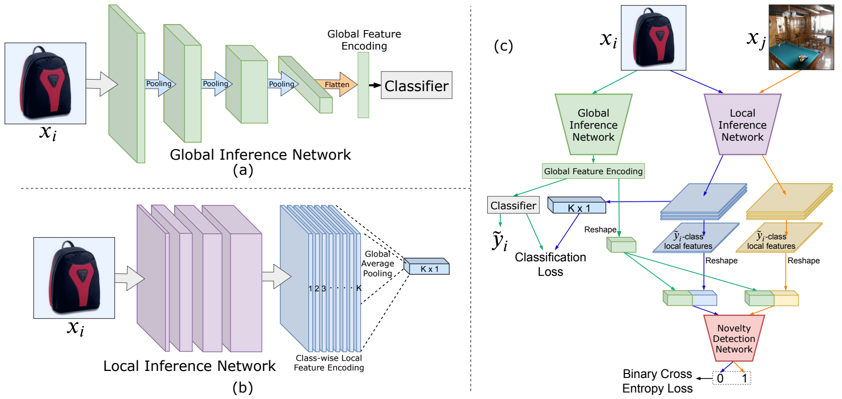

Fig. 4. (a) The global inference network processes the image to produce a global feature encoding which is used by the classifier to predict the class label. (b) The local inference network architecture provides patch-level features which are used to produce class-wise local feature encoding for all K classes, providing information regarding the presence of all classes at the patch-level. (c) Both global and local network information are combined in a novel training strategy for novelty detection, specifically to model mis-match between local activations and global predictions. For given any image x_i, the global and local features of the predicted class $\tilde{y}_i$ are concatenated to create a positive example. Local feature of the predicted class $\tilde{y}_i$ for another randomly sampled image x_j from a different class is combined with the same global feature to create a negative example. The novelty detection network is trained to distinguish between these positive and negative examples. The Global and Local inference networks are trained using the cross entropy classification loss on their respective predictions. *Note that, both x_i and x_j are sampled from in-class data.*

represents global average pooling operation along the height and width of the prediction map (shown in Fig. 4).

3.3 Novelty Detection Network

The proposed novelty detection method utilizing global and local inference is illustrated in Fig. 4(c). As discussed earlier, the proposed approach relies on two assumptions, *1)* the activation patterns for a particular global predictions are different in the case of in-class sample and novel class sample, and *2)* for each image from in-class data belonging to a particular class (y_i), DCNN produces activation maps that has some contribution from all known classes.

Based on these assumptions, we train the novelty detection network to model the probability of mis-match (discrepancy) between the predicted label by the global inference and corresponding patch-level activation patterns predicted by the local inference. This modeling should help during testing to detect novel samples by detecting the mis-match between the activation patterns and the prediction. Specifically, consider two randomly sampled images x_i and x_j having corresponding labels y_i and y_j, such that $y_i \neq y_j$. The predicted label and global feature encoding for image x_i is denoted as $\tilde{y}_i = \arg\max_i \mathcal{C}(\mathcal{G}(x_i))$ and $g_i = \mathcal{G}(x_i)$, respectively. The local feature encoding belonging to the predicted class $\tilde{y}_i$ for both images x_i and x_j are denoted as $r_i = \mathcal{R}(x_i)_{\tilde{y}_i}$ and $r_j = \mathcal{R}(x_j)_{\tilde{y}_i}$, respectively. This process is illustrated in Fig. 4(c). The following loss is used for training the novelty detection network

$$\begin{aligned}\mathcal{L}_{novelty} = \frac{1}{n} \sum_{\substack{i=1, y_i \neq y_j \\ j \sim \{1,..,n\}}}^{n} & L_{ce}(\mathcal{N}(cat(g_i, r_i)),\ 0) \\ & + L_{ce}(\mathcal{N}(cat(g_i, r_j)),\ 1),\end{aligned} \tag{3}$$

where $\mathcal{N}$ denotes the novelty detection network and *cat* represents reshape and concatenation operations. Also, $j \sim \{1,..,n\}$ and $y_i \neq y_j$ denote that for every training image x_i an index j is randomly sampled from the given in-class data, such that both x_j and x_i have different labels. During, testing the novel samples are identified by using predictions from network $\mathcal{N}$. The overall objective for the proposed approach can be written by combining Eq. (1)-(3) as follows

$$\underset{\mathcal{N},\ \mathcal{G},\ \mathcal{R},\ \mathcal{C}}{min} \quad \mathcal{L}_{global} + \mathcal{L}_{local} + \mathcal{L}_{novelty}. \tag{4}$$

Details regarding the network architectures and training procedures are provided in supplementary material.

3.4 Leveraging a Reference Dataset

The proposed method can be easily extended in the case where the reference dataset is available. We apply regularization on penultimate activations of the global inference network, similar to the loss function proposed in [10]. Such regularization of the final layer activations penalizes the high activations of any input from the reference dataset. Let us denote the reference dataset as $\mathcal{D}_{ref}$ having m number of images, then the regularization loss can be expressed as follows

$$\mathcal{L}_{reg} \ = \ \frac{1}{m} \sum_{x \in \mathcal{D}_{ref}} \quad \|\mathcal{C}(\mathcal{G}(x))\|_2. \tag{5}$$

The final objective function in this case is updated by adding $\mathcal{L}_{reg}$, in Eq. 4 as,

$$\underset{\mathcal{N},\ \mathcal{G},\ \mathcal{R},\ \mathcal{C}}{min} \quad \mathcal{L}_{global} + \mathcal{L}_{local} + \mathcal{L}_{novelty} + \ \lambda\mathcal{L}_{reg}. \tag{6}$$

Here, the parameter λ controls the effect of regularization on the final activations, and is chosen using the validation accuracy of the dataset. In experiments, we set parameter λ equal to 0.001.

4 Experiments and Results

4.1 Novelty Detection Datasets

Caltech-256. The Caltech-256 dataset contains 256 object classes and a total of 30607 images. The dataset has a minimum of 80 images to a maximum of

Table 1. Novelty detection performance measured using the Area Under the receiver operating characteristic Curve evaluation metric (AUC). The best performing method for each dataset is shown in bold. The second best method is shown in italics. Here, symbol † indicate that reference dataset was used during training for that method.

Method	**Caltech**		**CUB**		**Stanford Dogs**		**FounderType**		**Overall**
	VGG16	AlexNet	VGG16	AlexNet	VGG16	AlexNet	VGG16	AlexNet	**Performance**
Fine-tune	0.827	0.785	0.931	0.909	0.766	0.702	0.841	0.650	0.801
K-extremes [32]	0.546	0.521	0.520	0.514	0.610	0.592	0.557	0.512	0.546
OC-SVM [31]	0.576	0.561	0.554	0.532	0.542	0.520	0.627	0.612	0.567
KNFST [5]	0.743	0.688	0.891	0.748	0.633	0.602	0.870	0.678	0.732
Local KNFST [4]	0.712	0.628	0.820	0.690	0.626	0.600	0.673	0.633	0.673
OpenMax [2]	0.831	0.787	0.935	0.915	0.776	0.711	0.852	0.667	0.809
Fine-tune† [28]	0.848	0.788	0.921	0.899	0.780	0.692	0.754	0.723	0.800
DTMND† [28]	*0.869*	0.807	0.958	0.947	0.825	0.748	*0.893*	0.741	0.848
Proposed	0.859	*0.826*	*0.972*	*0.952*	*0.827*	*0.751*	0.876	*0.798*	*0.857*
Proposed†	**0.870**	**0.847**	**0.979**	**0.965**	**0.873**	**0.812**	**0.898**	**0.801**	**0.879**

827 images per category. Based on the protocol defined in [28], we first sort all classes into the alphabetical order according to their class name. The first 128 classes and the last 128 classes are considered as in-class and novel categories, respectively. The in-class categories are further divided into 50-50 splits to create training and test sets.

Caltech-UCSD Birds-200. The Caltech-UCSD Birds (CUB-200) is a fine-grained bird classification dataset. It contains 200 distinct bird categories and 6033 images in total. Similar to the protocol used before, the first 100 classes in the alphabetical order are picked as in-class categories and the last 100 classes in the alphabetical order are considered as the novel classes. The in-class categories are further divided into 50-50 splits to create training and test sets. As before, we make sure that both novel and in-class categories have equal number of images.

Stanford Dogs. This is another fine-grained classification dataset, containing 120 distinct dog breeds and a total of 20580 images. After sorting the dog breed classes in the alphabetical order, we pick the first and the last 60 breed categories as in-class and novel class, respectively. The in-class categories are further divided into 50-50 splits to create training and test sets. The number of images are the same for both in-class and novel classes during testing.

FounderType-200. The FounderType-200 dataset contains 200 different font types corresponding to the Chinese characters. Each font type category contains 6763 images. Similar to the other datasets, the first 100 font types are used as in-class categories and the last 100 font types are used as the novel class categories. We keep 50% of the image samples per category as the training set and the remaining 50% are used for testing. The number of images are the same for both in-class and novel classes during testing.

4.2 Quantitative Analysis

Novelty Detection Performance We evaluate the performance of our method and compare it with several recent novelty detection methods. Each method provides a score to quantify the novelty of a test image. The lower the score, the higher the probability of input being from a novel class and vice versa. Following the protocol proposed in [28], we compare all methods using AlexNet [18] and VGG16 [35] as the global inference network architectures. In our approach, BagNet-33 [7] is used as the local inference network. Below is the list of methods used for comparison:

- **Fine-tune:** In this baseline, the pre-trained DCNN models are fine-tuned on the in-class data samples. The scores from penultimate layer of the models are used to evaluate novelty detection performance.
- **OC-SVM:** One-class SVM [31] is trained on the fine-tuned features and the SVM scores are used to evaluate the novelty detection performance.
- **KNFST:** KNFST as proposed in [4]. It uses fine-tuned deep features to learn a subspace for in-class data. The distance from the subspace is used to evaluate the performance.
- **Local KNFST:** Local KNFST [4] is an extension of the previous baseline, where a local region of in-class data are used to compute the score for performance evaluation.
- **OpenMax:** OpenMax [2] uses penultimate layer scores of a fine-tuned DCNN and distance from class-wise mean vectors combined with extreme value modeling for performance evaluation.
- **K-extremes:** This baseline focuses on the penultimate activations where top 10% of the sorted activations are binarized to find extreme signatures, which are later used to compute the normalized scores for performance evaluation.
- **Fine-tune†:** This is another fine-tuning baseline proposed in [28]. Here, during fine-tuning DCNN on any given novelty detection dataset, a *reference dataset* is used to improve the quality of the features. During testing, the maximum score from the penultimate layer of a DCNN, extracted from the in-class categories (excluding the reference dataset) is used for performance evaluation.
- **DTMND:** Recently proposed novelty detection method, where a *reference dataset* is utilized in a novel training strategy to learn better model that can respond negatively to the novel classes. Maximum activation from the penultimate layer of the model is used for evaluating the novelty detection performance.

The evaluation protocol proposed by [28] considered two more baselines, namely KNFST-*pre* and Local KNFST-*pre*. However, we excluded these from comparison here as they do not observe any improvement over the KNFST and Local KNFST baselines. More details regarding these baselines are provided in [28]. For the proposed method, we use addition of scores from the global inference and the novelty detection networks to evaluate the performance.

The performance of different methods are evaluated using the area under the receiver operating characteristic curve (AUC) metric. The results are reported in Table 1. As can be seen from this table, OC-SVM and K-extremes methods have the lowest performances. Local KNFST performs better than both OC-SVM and K-extremes for all four datasets. KNFST provides better performance compared to Local KNFST on average, and has consistently better performance on all datasets. On average Fine-tune and Fine-tune† have similar performances. However, their performances are inconsistent across datasets and network architectures. For the Caltech-256 dataset, Fine-tune† performs better than Fine-tune for both AlexNet and VGG16, while for CUB-200 the trend is reversed. For both the Stanford Dogs and the FounderType-200 datasets, Fine-tune† performs better when the VGG16 architecture is used and the reverse trend is observed when the AlexNet architecture is used. The performance obtained by Fine-tune† baseline shows that simple fine-tuning is not an efficient way to utilize a reference dataset for novelty detection. OpenMax performs better than both Fine-tune and Fine-tune† baselines, resulting in 1% overall improvement. Except for the FounderType-200 dataset using the VGG16 architecture, OpenMax consistently performs better than OC-SVM, K-extremes, Local KNFST, KNFST, Fine-tune and Fine-tune† baselines. Out of all the baselines, DTMND yields the best performance. DTMND on average performs 3% better than the next best performing baseline and performs approximately 5% better than Fine-tune† on average. Even-though both of these baselines have access to a reference dataset, DTMND utilizes this additional data more efficiently, resulting in the better performance. The performance of DTMND is largely attributed to their approach for fine-tuning using the reference dataset.

In the absence of reference dataset, the best method in the literature DTMND would become similar to that of fine-tune baseline and the performance will drop by ∼5% to 0.80. Whereas the proposed approach without the reference dataset during training provides approximately 6% improvement over the DTMND without reference dataset. This is due to the fact that the performance gain for DTMND is mainly due to the fact that it uses an external reference dataset for training the network. When the reference dataset is utilized during the training of the proposed approach (described in Eq. 5), the proposed approach consistently performs better than DTMND for all datasets and network architectures. Overall, when the proposed approach is trained with the help of reference dataset it improves by ∼2% and provides ∼4% improvement over the DTMND. The performance improvement with the proposed† method shows that our approach can be easily extended to a scenario where a reference dataset is available to further enhance the novelty detection performance. On the other hand, DTMND becomes sub-optimal for the cases where a reference dataset is not available. Especially in such cases the proposed approach is a better alternative for DCNN-based multi-class novelty detection compared to DTMND.

Ablation Analysis In this section, we provide an ablation analysis showing the significance of combining patch-level information with global in our approach. For ablation experiments, we consider all four novelty detection datasets and the corresponding protocol proposed in Sec. 4.1. For all experiments, VGG16 is used as the global inference network. The following ablation baselines are considered:

- **Global Only:** This baseline is similar to Fine-tune as described in Sec. 4.2. The in-class data samples are used to fine-tune the VGG16 network. The maximum activation score from the penultimate layer of VGG16 is used to evaluate the novelty detection performance.
- **Local Only:** Fine-tuning only the local inference network using the given

Table 2. Ablation analysis using AUC. The best performing method is shown in bold.

Method	Caltech	CUB	Stanford Dogs	FounerType	Overall Performance
Global Only	0.827	0.931	0.766	0.841	0.841
Local Only	0.799	0.785	0.598	0.773	0.739
Global+Local	0.831	0.943	0.741	0.835	0.837
Proposed	**0.859**	**0.972**	**0.827**	**0.876**	**0.883**

in-class data. The maximum activation score from the penultimate layer of the local inference network is used to evaluate the novelty detection performance.

- **Global+Local:** Here, we perform a straight forward concatenation of information from the global and local inference networks. The novelty detection performance is evaluated based on the addition of scores from both networks.
- **Proposed:** This is the method proposed in the paper, where instead of a straight-forward fusion we utilize novel training strategy proposed in Sec. 3, to train a novelty detector network, which can better identify the mismatch of local activity patterns for global feature of a given category.

The performance of all three ablation baselines are reported in Table. 2. The lowest performance is obtained by local only baseline. The local inference network processes image patches and classifies images based on the local image features. This leads to relatively poor classification of in-class samples, which in turn hurts the novelty detection performance. On the other hand, the global inference network processes the entire image with a cascade of convolutional, pooling and fully connected layers to get a feature encoding for the entire image. This helps the global only baseline perform better classification and generates high prediction scores for the in-class samples. However, the problem with the global only baseline is that it ends up providing high prediction scores for the novel class samples as well, hurting the novelty detection performance. In the proposed approach, the novelty detection network is trained using both local and global inference networks. The combined information and novel training strategy helps the trained novelty detection network to perform better in identifying novel classes. Specifically, the local inference network provides patch-level activation information corresponding to the prediction provided by the global inference network. The novelty detection network identifies the mismatch between the patch-level activation patterns and global feature encoding to predict whether the input image belongs to either in-class or novel class. As a result, the proposed method performs approximately 14% and 4% better than the local and the global baselines, respectively. We also compare the performance of our method with a *naive* fusion baseline, i.e. Global+Local, where the information from global and local networks are directly concatenated and the performance evaluation is done using the added scores. From Table. 2, it can be observed that the proposed approach is able to perform better than the Global+Local baseline.

4.3 Qualitative Analysis

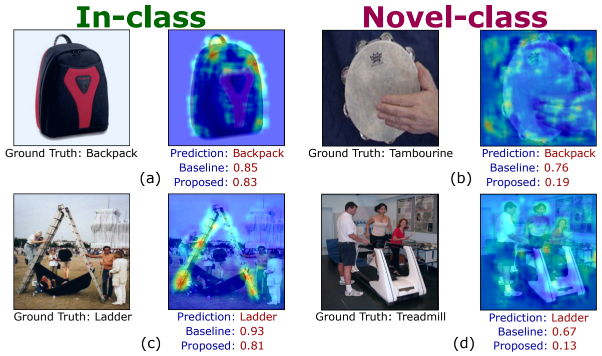

Fig. 5. Image examples of in-class (a) & (c) and novel class (b) & (d) data with corresponding class activation heat-maps as predicted by local inference network and scores assigned using both baseline and proposed.

Fine-tune Baseline vs Proposed Method To show the effectiveness of the proposed approach, we provide a qualitative comparison with the Fine-tune baseline (i.e. traditional DCNN) in Fig. 5. Specifically, we provide image examples, prediction from the global inference network, their corresponding local class-activation heat-maps and scores assigned by both baseline and the proposed method. The heat-maps are generated by normalizing the local feature encodings of the class predicted by the global inference network. The images presented here are from two novel classes, namely, 'Tambourine' and 'Treadmill', as shown in Fig. 5(b), Fig. 5(d), respectively. These images are wrongly identified by the baseline as in-class data, and assigned the category 'Backpack', and 'Ladder' with high scores. Additionally, we show the images from the corresponding in-class categories 'Backpack' and 'Ladder' and their corresponding class activation heat-maps in Fig. 5(a) and Fig. 5(c), respectively. This figure shows the difference in class activation heat-maps for the case where the image samples are from in-class data and the case where the image samples are from novel classes. For example, in Fig. 5(a), the image sample is from a known class with category label 'Backpack' and the network is able to correctly identify it by assigning a high score. The patch-level class activation patterns shown in heat-map focuses on highly discriminative patch locations providing strong presence of the given class. On the other hand, in Fig. 5(b), the image sample is from a novel class, but the network wrongly identifies it as 'Backpack' with a high score. However, if we look at the class activation patterns, there are moderate to high activations all over the image, as opposed to in-class image in Fig. 5(a). The novelty detector of the proposed method is specifically trained to identify this mis-match in activation patterns and predicted label. This helps the proposed approach correctly predict a high score for the image sample of 'Backpack' and a low score for the image sample of a novel class, 'Tambourine'. Similar observations can be made for the other example provided for 'Ladder' in Fig. 5(c) and Fig. 5(d).

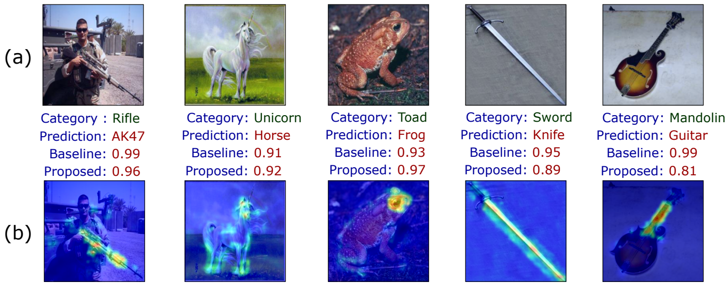

Fig. 6. Examples of images from novel classes that are wrongly identified as in-class samples with high scores.

Examples of Wrong Prediction Though the proposed approach exhibits reasonable novelty detection performance, there are some cases where it fails to predict low scores when the samples are from novel classes. Some of these examples are illustrated in Fig. 6 with their corresponding class activation heat-maps and the predicted scores using the Fine-tune baseline (i.e. traditional DCNN) and the proposed method. The image sample from novel category 'Toad' is identified as in-class category 'Frog'. In this case, the novelty detector network fails to detect any mis-match between the local patch-wise activation patterns and the predicted label. Similarly, the novel categories 'Unicorn', 'Rifle' and 'Mandolin' are identified as in-class categories 'Horse', 'AK47' and 'Guitar', respectively. For all of these examples presented here, the reason for failure can be due to very subtle differences between these novel categories with their respective misclassified in-class categories.

5 Conclusion

We proposed a novel DCNN-based multi-class novelty detection method, that is end-to-end trainable. Unlike recent methods, the proposed approach does not rely on the availability of a reference dataset and is flexible enough to work on both scenarios, when the reference dataset is available and when it is not. We discussed assumptions regarding patch-level activation patterns of DCNNs when the test image is from novel classes. Based on these assumptions, we proposed a novel training methodology which utilizes both global level predictions from the traditional DCNNs and a local inference network, which processes image at patch level. Furthermore, we show how the proposed approach can be extended when a reference dataset is accessible by regularizing the reference data penultimate activations. Experimental results, evaluated on four multi-class novelty detection datasets, show that the proposed method is able to identify novel class samples better compared to the other DCNN-based methods.

Acknowledgement

This work was supported by the NSF grant 1910141.

References

1. Baweja, Y., Oza, P., Perera, P., Patel, V.M.: Anomaly detection-based unknown face presentation attack detection. International Joint Conference on Biometrics (IJCB), Houston, TX (2020) 1
2. 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) 1, 9, 10
3. Bhattacharjee, S., Mandal, D., Biswas, S.: Multi-class novelty detection using mix-up technique. In: The IEEE Winter Conference on Applications of Computer Vision. pp. 1400–1409 (2020) 1
4. Bodesheim, P., Freytag, A., Rodner, E., Denzler, J.: Local novelty detection in multi-class recognition problems. In: 2015 IEEE Winter Conference on Applications of Computer Vision. pp. 813–820. IEEE (2015) 4, 9, 10
5. Bodesheim, P., Freytag, A., Rodner, E., Kemmler, M., Denzler, J.: Kernel null space methods for novelty detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3374–3381 (2013) 1, 2, 3, 9
6. Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H.B., Patel, S., Ramage, D., Segal, A., Seth, K.: Practical secure aggregation for privacy-preserving machine learning. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. pp. 1175–1191 (2017) 3
7. Brendel, W., Bethge, M.: Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. arXiv preprint arXiv:1904.00760 (2019) 6, 10
8. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: 2017 IEEE Symposium on Security and Privacy (SP). pp. 39–57. IEEE (2017) 1
9. DeVries, T., Taylor, G.W.: Learning confidence for out-of-distribution detection in neural networks. arXiv preprint arXiv:1802.04865 (2018) 1
10. Dhamija, A.R., Günther, M., Boult, T.E.: Improving deep network robustness to unknown inputs with objectosphere 8
11. Eskin, E., Arnold, A., Prerau, M., Portnoy, L., Stolfo, S.: A geometric framework for unsupervised anomaly detection. In: Applications of data mining in computer security, pp. 77–101. Springer (2002) 3
12. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2014) 1
13. Hautamaki, V., Karkkainen, I., Franti, P.: Outlier detection using k-nearest neighbour graph. In: Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. vol. 3, pp. 430–433. IEEE (2004) 3
14. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. arXiv preprint arXiv:1610.02136 (2016) 1
15. Hoffmann, H.: Kernel pca for novelty detection. Pattern recognition **40**(3), 863–874 (2007) 1, 3
16. Kim, B., Kim, H., Kim, K., Kim, S., Kim, J.: Learning not to learn: Training deep neural networks with biased data. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019) 1
17. Knorr, E.M., Ng, R.T., Tucakov, V.: Distance-based outliers: algorithms and applications. The VLDB Journal—The International Journal on Very Large Data Bases **8**(3-4), 237–253 (2000) 3
18. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012) 10

19. Liang, S., Li, Y., Srikant, R.: Enhancing the reliability of out-of-distribution image detection in neural networks. arXiv preprint arXiv:1706.02690 (2017) 1
20. Liu, J., Lian, Z., Wang, Y., Xiao, J.: Incremental kernel null space discriminant analysis for novelty detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 792–800 (2017) 1, 2, 4
21. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083 (2017) 1
22. Markou, M., Singh, S.: Novelty detection: a review—part 1: statistical approaches. Signal processing **83**(12), 2481–2497 (2003) 1
23. Neal, L., Olson, M., Fern, X., Wong, W.K., Li, F.: Open set learning with counterfactual images. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 613–628 (2018) 1
24. Oza, P., Nguyen, H.V.N., Patel, V.M.: Multiple class novelty detection under data distribution shift. In: European Conference on Computer Vision. Springer (2020) 2
25. Oza, P., Patel, V.M.: One-class convolutional neural network. IEEE Signal Processing Letters **26**(2), 277–281 (2018) 1
26. Perera, P., Morariu, V.I., Jain, R., Manjunatha, V., Wigington, C., Ordonez, V., Patel, V.M.: Generative-discriminative feature representations for open-set recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11814–11823 (2020) 1
27. Perera, P., Nallapati, R., Xiang, B.: Ocgan: One-class novelty detection using gans with constrained latent representations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2898–2906 (2019) 1
28. Perera, P., Patel, V.M.: Deep transfer learning for multiple class novelty detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 11544–11552 (2019) 1, 2, 3, 4, 9, 10
29. Perera, P., Patel, V.M.: Learning deep features for one-class classification. IEEE Transactions on Image Processing **28**(11), 5450–5463 (2019) 1
30. Samangouei, P., Kabkab, M., Chellappa, R.: Defense-gan: Protecting classifiers against adversarial attacks using generative models. arXiv preprint arXiv:1805.06605 (2018) 1
31. Schölkopf, B., Smola, A.J., Bach, F., et al.: Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT press (2002) 3, 9, 10
32. Schultheiss, A., Käding, C., Freytag, A., Denzler, J.: Finding the unknown: Novelty detection with extreme value signatures of deep neural activations. In: German Conference on Pattern Recognition. pp. 226–238. Springer (2017) 1, 4, 9
33. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 618–626 (2017) 5
34. Shao, R., Perera, P., Yuen, P.C., Patel, V.M.: Open-set adversarial defense. In: European Conference on Computer Vision. Springer (2020) 1
35. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014) 10
36. Srinivas, N., Ricanek, K., Michalski, D., Bolme, D.S., King, M.: Face recognition algorithm bias: Performance differences on images of children and adults. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 0–0 (2019) 1

37. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199 (2013) 1
38. Tax, D.M., Duin, R.P.: Support vector data description. Machine learning **54**(1), 45–66 (2004) 3
39. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of cognitive neuroscience **3**(1), 71–86 (1991) 3
40. Vera-Rodriguez, R., Blazquez, M., Morales, A., Gonzalez-Sosa, E., Neves, J.C., Proenca, H.: Facegenderid: Exploiting gender information in dcnns face recognition systems. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 0–0 (2019) 1
41. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. IEEE transactions on pattern analysis and machine intelligence **31**(2), 210–227 (2008) 3
42. Xia, Y., Cao, X., Wen, F., Hua, G., Sun, J.: Learning discriminative reconstructions for unsupervised outlier removal. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1511–1519 (2015) 1
43. You, C., Robinson, D.P., Vidal, R.: Provable self-representation based outlier detection in a union of subspaces. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017) 1
44. Zhang, H., Patel, V.M.: Sparse representation-based open set recognition. IEEE transactions on pattern analysis and machine intelligence **39**(8), 1690–1696 (2016) 3