Multiple Class Novelty Detection Under Data Distribution Shift

Poojan Oza[1], Hien V. Nguyen[2], and Vishal M. Patel[1]

[1] Johns Hopkins University, 3400 N. Charles St, Baltimore, MD 21218, USA
[2] University of Houston, Houston, TX 77004, USA
{poza, vp36}@jhu.edu, hienvnguyen@uh.edu

Abstract. The novelty detection models learn a decision boundary around multiple categories of a given dataset. This helps such models in detecting any novel classes encountered during testing. However, in many cases, the test data distribution can be different from that of the training data. For such cases, the novelty detection models risk detecting a known class as novel due to the dataset distribution shift. This scenario is often ignored while working with novelty detection. To this end, we consider the problem of multiple class novelty detection under dataset distribution shift to improve the novelty detection performance. Firstly, we discuss the problem setting in detail and show how it affects the performance of current novelty detection methods. Secondly, we show that one could improve those novelty detection methods with a simple integration of domain adversarial loss. Finally, we propose a method which brings together the techniques from novelty detection and domain adaptation to improve generalization of multiple class novelty detection on different domains. We evaluate the proposed method on digits and object recognition datasets and show that it provides improvements over the baseline methods.

Keywords: Dataset distribution shift, multiple class novelty detection

1 Introduction

In recent years, improving robustness of convolutional neural networks (CNNs) has received an increasing amount of attention [6,2]. Many problems such as countering adversarial/trojan/poison attacks [26,25,8,47], detecting novel categories [42,40,7,37,35,39] and out-of-distribution samples [13,24,10,54] etc. tackle different aspects of robustness of CNNs. One of the practical aspect related to model robustness is detection of samples belonging to novel categories during testing. Specifically, when the CNN models are tested in the real world environment, it is highly likely that the models will observe samples from categories that were not present during training. To tackle such cases, it would be better to first identify whether the given sample is from a novel category or not and only then should be passed through CNN for classification if it is identified as known. This problem is commonly referred to as novelty detection.

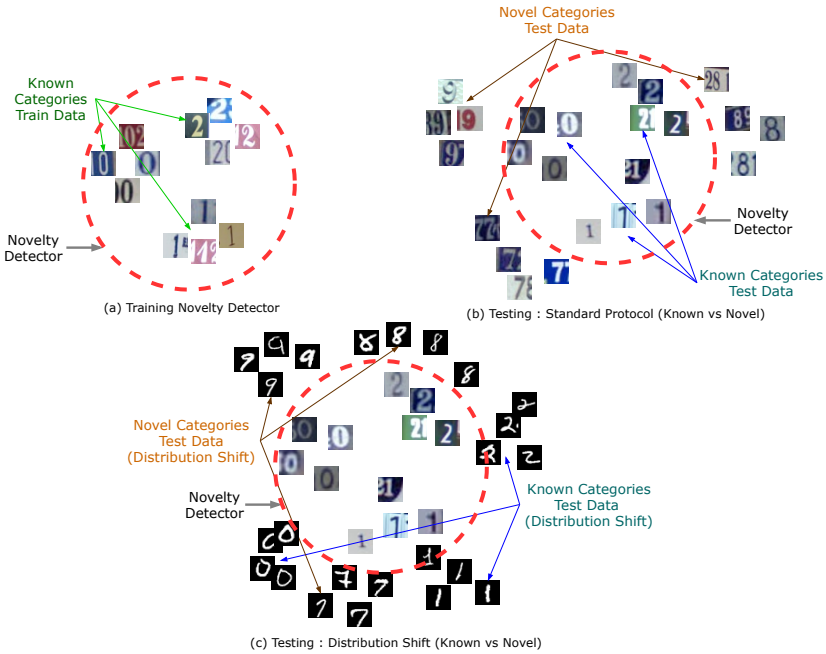


Fig. 1: An overview of the proposed problem setting. (a) We have a training data with samples from multiple known categories. Here, we have used the SVHN dataset with digits 0, 1 and 2 as known categories. These data samples are used to learn a novelty detector to enclose the known categories. (b) In a standard novelty detection testing protocol, the test data follows the same distribution as the training data. As shown in the figure, typically the novelty detector is able to distinguish between known categories and novel categories. Here, digits 7, 8 and 9 sampled from the SVHN dataset are used as novel categories. As illustrated in the figure, the learned novelty detector is able to differentiate between known and novel digits from the SVHN dataset correctly. (c) This figure illustrates the scenario where the test data does not follow the distribution of the training dataset. When tested with known (0, 1, 2) and novel (7, 8, 9) digits from the MNIST dataset, due to the distribution shift, the learned novelty detector performs poorly. This problem arises due to the fact that while training any novelty detector to enclose the known categories of a particular dataset, it also encloses the style/domain of that dataset. This creates a problem as shown in this figure, where the data from known categories, which follow a different distribution will have high risk of being detected as novel category.

There has been a lot of work done in the literature for the novelty detection task [42,40,7,34,3,4]. Typically, the novelty detection methods try to learn a decision boundary that encloses the known categories given in the dataset. However, while trying to enclose the known categories, these methods also enclose the style/domain of the dataset. As a result, samples from known categories

but having different style/domain, will have increased risk of false detection as a novel category. For example, a novelty detection method trained on SVHN digits dataset will be correctly able to detect known categories from novel, only if the test data follows the same distribution as SVHN. But, if the test data is from a digits dataset like MNIST, due to the domain shift, it is highly likely that the novelty detector will not be able to distinguish between novel and known categories accurately. This problem is also illustrated in Fig. 1. Most of the earlier novelty detection methods work on the assumption that the test data would follow a similar distribution as the training data.

A simple solution to this problem would be to create another dataset for the new domain. In the case of novelty detection, one could avoid labeling the dataset by considering the whole dataset as one class, and training any off-the-shelf novelty detection algorithm on it. However, most datasets contain multiple categories and ignoring this multi-class structure of the dataset could restrict the performance of the novelty detection algorithm. If we wish to exploit the multi-class structure to help novelty detection, it would require labeling efforts that are costly and time consuming. This problem can be solved to an extent by transferring the knowledge from a labeled dataset that has different style/domain to the dataset of interest. This type of problem setting has been widely studied as unsupervised domain adaptation [49,11,14], in the literature, and specifically deals with the dataset distribution shift issue. However, most of the work on this topic has been done for the task of classification [52,49,11,14], segmentation [50,55,15], detection [16,9,19] etc. and to the best of our knowledge no work is available in the literature that addresses the distribution shift problem for novelty detection.

To this end, we consider the problem of multiple-class novelty detection under dataset distribution shift. Since no prior work has been done for this specific problem, we first describe the problem statement in detail and provide trivial baselines for this task based on novelty detection and domain adaptation approaches. Furthermore, we propose a novelty detection method that can address the data distribution shift problem and help improve over the trivial baselines. Moreover, we discuss the differences between the closely related problem setting such as open-set domain adaptation [36] and also provide experimental analysis to show that their performance is sub-optimal in the problem setting considered in this paper.

To summarize, this paper makes the following contributions:

- We consider novelty detection under dataset distribution shift. To the best of our knowledge this is the first work to consider data distribution shift in the context of novelty detection.
- We show the effects of distribution shift on current novelty detection methods and provide a few baselines that combine novelty detection and domain adaptation techniques.
- We propose an algorithm to mitigate data distribution shift for novelty detection, and show that it can improve the detection performance over the trivial novelty detection baselines.

2 Related Work

Novelty Detection. Earlier works in novelty detection were based on Principle Component Analysis (PCA) [51], Mixture Models [30], Support Vector Machines [46] etc. Typically, these methods work on features extracted from the image and learn a decision boundary to enclose the extracted features from the dataset. However, most of the methods for novelty detection have shifted to CNNs in recent years due to their outstanding representation learning capability. Especially, unsupervised learning strategies such as auto-encoders [1] and generative adversarial networks [12] are among the most popular algorithms for novelty detection. Some approaches use auto-encoders [1] for novelty detection. However, such approaches are not optimal since auto-encoder often suffer from blurry reconstructions. Sabokrou *et al.* [42] proposed a novelty detection algorithm using a de-noising auto-encoder based generative adversarial network. Specifically, during training, input is injected with gaussian noise and auto-encoders are tasked to provide clean reconstructions. The reconstructions are supervised with a combination of adversarial loss and reconstruction loss. Finally, discriminator prediction probability of the reconstructed image is used as the novelty detection score. Pidhorskyi *et al.* [40] proposed another method based on adversarial auto-encoders [29]. Specifically, the encoder is trained to learn a feature embedding that are Gaussian distributed and the overall network is designed to reconstruct the original image. Both of these approaches are shown to work reasonably well when there are multiple categories present in the dataset and both show a marginal drop in the performance with increased number of categories. Recent works such as OC-GAN [38] and non-adversarial generative method [7] consider a specific case where it is assumed that there is only one category available in the dataset. With that assumption, they learn a one-class novelty detector to enclose a particular given category. The authors of these approaches have not evaluated the performance of their methods in the case when there are multiple categories present in the dataset. Moreover, when the dataset contains only one category, it is better to just train the novelty detector on the data from the new domain. The problem of distribution shift is much more relevant when datasets contain multiple categories, which is a more realistic scenario. However, all of these approaches do not consider the scenario of distribution shift in the dataset.

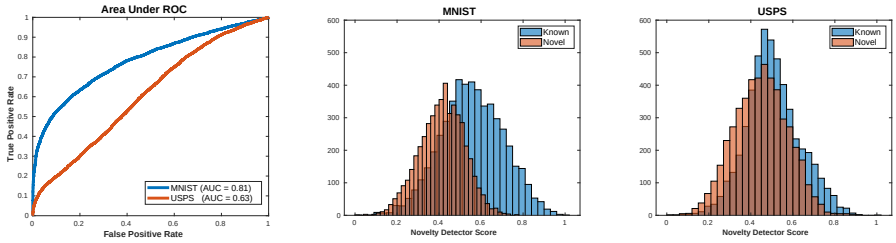
Domain Adaptation. Unsupervised domain adaptation problem has been well-studied in the literature for image classification task. It is defined as aligning domains having distinct distributions, namely source and target containing same categories. In unsupervised domain adaptation, it is assumed that images in the source dataset are available with category labels, while no label information is provided for the target images. The most popular approaches for this task are based on CNNs. Some of these approaches include feature distribution alignment [52], [11], [48], [44], similarity learning [41], residual transfer [27], [28], and generative adversarial network-based methods [17], [31], [14], [45]. These methods mostly consider a setting where both source and target datasets have equal number of categories. Recently, some works have started to consider different settings

where the number of categories in source and target are not the same. These extensions include partial domain adaptation [5], universal domain adaptation [53] and open-set domain adaptation [36]. Partial domain adaptation assumes that target domain categories are a subset of the source domain categories and hence only a part of the source dataset is useful during adaptation. Whereas open-set domain adaptation assumes that the source domain categories are a subset of the target domain categories and hence only a part of the target data is useful for the adaptation. Universal domain adaptation brings both open-set and partial settings together into a single framework. All of these modifications to the original domain adaptation problem setting are designed to improve the domain adaptation performance on more practical scenarios.

The most related problem to the proposed scenario available in the literature is open-set domain adaptation proposed by Busto and Gall *et al.* [36]. However, we would like to point out that there are some key differences between open-set domain adaptation and the proposed approach. Specifically, in open-set domain adaptation, the target categories are a superset of the source categories, i.e., there are some unknown categories available in the target dataset. Since, no labels are provided for the target domain, the challenge for open-set domain adaption method is to separate out the samples belonging to known and unknown categories in the available target dataset. This extends the domain adaptation capability to a real-world scenario where the target category set will be a superset of the source. In the proposed problem, we do not modify the domain adaptation setting like the open-set domain adaptation, but on the contrary, utilize the domain adaptation techniques to improve generalization of novelty detection methods on different data domains. Specifically, in the proposed problem we have labeled data from the source domain and unlabeled data from the target domain and both of these domains share the same category set. Also, unlike open-set domain adaptation, where unknown category data samples are accessible during training, in the proposed problem setting, unknown category data samples are only observed during testing. The end goal for the proposed problem is to utilize the source domain information to create a better novelty detection model for the target domain data. Since both methods follow different problem settings, either of the methods would not be optimal for the other problem setting. We provide an experiment and discuss this point in more detail in the supplementary material.

3 Novelty Detection vs Distribution Shift

We provide a preliminary experimental analysis to show the effect of dataset distribution shift on the performance of novelty detection. For this experiment, we consider a novelty detector [42], referred to as Adversarially learned One-Class Classifier (ALOCC). The ALOCC method is trained on the MNIST dataset. For training, we consider digits 0 to 4 as known categories and the remaining digits as novel categories. Fig. 2(a) shows the ROC curve illustrating the performance of the novelty detector when evaluated on the MNIST data (Blue curve). The



(a) (b) (c)

Fig. 2: (a) Area under the ROC plot when a novelty detector is evaluated on the MNIST and USPS datasets. (b) Histogram of scores corresponding to the MNIST dataset. (c) Histogram of scores corresponding to the USPS dataset.

novelty detector achieves area under the curve of 0.81. In order to simulate the data distribution shift, we evaluate the novelty detector on the USPS dataset, again considering 0 to 4 digits as known categories and the remaining digits as novel categories. As we can see from Fig. 2(a), the performance on the USPS dataset (red curve) drops by ∼20% compared to the MNIST dataset. Also, by looking at the histogram of score predictions in Fig. 2(b) and Fig. 2(c), it is clear that compared to MNIST, USPS scores for both known and novel categories on average are shifted towards the left. This shows that the novelty detector trained on MNIST has high risk of detecting USPS known categories as novel. This is due to the shift in the distribution between MNIST and USPS datasets.

4 Robust Novelty Detection Under Distribution Shift

In this section, we first formulate the problem and then discuss some baseline methods. Finally, we present the proposed method in detail.

4.1 Problem Setting

Typically, a novelty detection model is developed using a training dataset having multiple categories which we refer to as the source dataset. This trained model is then tested in the real-world where the goal is to detect any test input samples belonging to novel categories. However, as discussed in Sec. 1, these models have high risk of detecting any test samples belonging to known categories as unknown, when the test samples are from a different distribution than that of the training dataset. The goal of the proposed problem setting is to generalize the novelty detection models on a dataset having different distribution, which we refer to as the target dataset. The terminology of referring labeled dataset as source and unlabeled dataset as target is borrowed from the domain adaptation literature. Formally, in the proposed problem setting, we have access to the source dataset, $\mathcal{D}_s = \{X_{si}\}_{i=1}^{N_s}$ and their corresponding label set $\mathcal{Y}_s = \{y_{si}\}_{i=1}^{N_s}$. There are in total C categories and each y_{si} takes a value from the label set $\{1, 2, ..., C\}$. Similarly, we have access to the target dataset, $\mathcal{D}_t^k = \{X_{ti}\}_{i=1}^{N_t}$,

having different distribution than the source dataset. Both source ($\mathcal{D}_s$) and target ($\mathcal{D}_t^k$) datasets share the same C categories. However, for $\mathcal{D}_t^k$ we do not have access to the corresponding labels. Here, the superscript k denotes that the dataset contains only the known categories, i.e., all data samples in the $\mathcal{D}_t^k$ belong to one of the categories from the label set $\{1, 2, ..., C\}$. During training, the goal is to learn a novelty detector that generalizes well on the target dataset with the help of the information available in the source dataset, i.e., $\mathcal{D}_s$ and $\mathcal{Y}_s$. The learned novelty detector is evaluated using a test set from the target dataset ($\mathcal{D}_t^{k:test}$) having known categories and a target set containing data from unknown categories ($\mathcal{D}_t^u$). Here, superscript u denotes that the dataset contains only novel categories. Note that data from $\mathcal{D}_t^u$ is not utilized during training but only used while evaluating the novelty detection performance on the target set.

4.2 Simple Approaches

As discussed in Sec. 1 and shown by preliminary experiment in Sec. 3 the dataset distribution shift is one of the unexplored problems in novelty detection. Following the problem setting and notations described in previous section, in this section, we explore some potential solutions for tackling this problem. Since there are no prior works available in the literature on this problem, we develop a few baselines by considering similar works from the literature. The block diagrams of these methods are illustrated in Fig. 3(a)-(d). In what follows, we describe these baseline approaches in detail.

Softmax. The most simple baseline would be to utilize the labeled source data to train a feature extractor and classifier network to perform multi-class classification. However, classification networks are prone to novel classes even in the source domain, hence would not translate well for the target domain novelty detection.

ALOCC. Another approach would be to disregard the source domain information and only use the target domain unlabeled data to train any off-the-shelf novelty detector algorithm. For this baseline, we utilize ALOCC method for novelty detection proposed in [42]. Specifically, ALOCC trains an auto-encoder which aims to reconstruct a clean image from the input image using Gaussian noise. This auto-encoder network is trained in generative adversarial framework and the score from the discriminator of the reconstructed image is used for novelty detection. The dataset will have multiple categories, however ALOCC remains agnostic to that by considering multiple categories as one.

GRL. Gradient reversal layer [11] has been widely used to reduce the domain gap between two datasets having different distributions for the classification task. GRL baseline can be considered as an extension to the Softmax baseline such that the domain gap issue between source and target is addressed by the gradient reversal layer.

ALOCC+GRL. This is the final baseline which combines the gradient reversal training to reduce the domain gap between source and target, together with the novelty detection training specified in the ALOCC. This ad-hoc combination provides a strong baseline for the proposed setting, since GRL is able to take

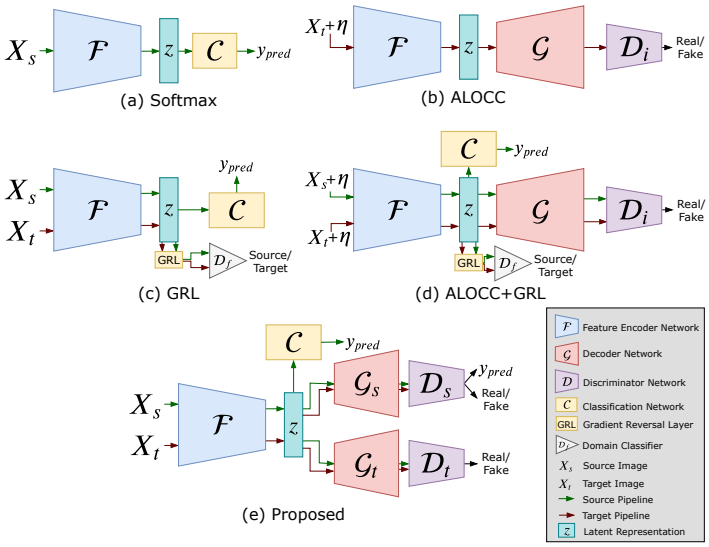


Fig. 3: Illustration of multiple potential solutions to address the distribution shift problem for novelty detection. (a) Softmax: Simplest approach which utilizes the labeled source data to train a classification network. The maximum softmax probability can be used as the novelty score. (b) ALOCC: Another approach which directly utilizes the unlabeled target data to train an off-the-shelf novelty detector. We utilize, a novelty detection algorithm proposed in [42]. Here, η denotes the Gaussian noise added to the input image. (c) GRL: Uses labeled source and unlabeled target data to learn a domain invariant feature space using a gradient reversal layer [11]. The maximum softmax prediction probability can be used as the novelty score. (d) ALOCC+GRL: A combination of both novelty detector [42] and domain invariant feature learning [11] in an ad-hoc manner. (e) Proposed method: A shared feature space is learned through cross-domain mappings. The corss-domain mappings helps to learn a better feature space which is especially useful for novelty detection.

care of the domain gap and with the help of domain invariant feature space, the ALOCC is able to learn a more general novelty detector which is likely to perform better on the target domain.

4.3 Proposed Method

ALOCC+GRL is the most related method out of all the methods described above. Also, it is able to exploit both novelty detection training and domain adversarial loss to learn a domain invariant feature space. This should help the novelty detector mitigate the effects of distribution shift and perform reasonably well on the target domain. However, such method is an ad-hoc combination of the domain adaptation and novelty detection algorithms. To get the best out of the information available in the proposed problem setting, we need a unified

approach where novelty detection training inherently mitigates the distribution shift. Fig. 3(e) gives an overview of the proposed approach, where the cross-domain decoders trained for novelty detection task guides the shared feature extractor to learn a common feature space. As opposed to the method with ad-hoc combination, the proposed way of learning can benefit from the unified training strategy, since the novelty detection task guides the feature space learning. Here, we discuss the training methodology used for proposed approach.

Let's consider images X_s and X_t sampled from the source and target domain, respectively. The feature encoder network ($\mathcal{F}$), takes these samples and generates latent representations z_s and z_t. Since, for the source domain, we have access to the class labels, the classifier ($\mathcal{C}$) is trained to classify latent representations of source domain in to respective categories. As discussed earlier, the feature extractor network $\mathcal{F}$ is learned with the help of two generator networks $\mathcal{G}_s$ and $\mathcal{G}_t$ for source and target domain, respectively.

For the source domain discriminator D_s, a conditional GAN [33] based approach is used. This specifically helps the generator networks when datasets contain multiple categories. Following the conditional GAN formulation proposed by [33], the discriminator network D_s has two parts. The first part referred to as, D_s^b, identifies whether the samples generated by $\mathcal{G}_s$ are real or fake by a binary classification. On the other hand, the second part referred to as, D_s^a, classifies the generated images into one of the known categories. $\mathcal{G}_s$ takes in the latent representations z_s and z_t to generate images $\hat{X}_{s2s}$ and $\hat{X}_{t2s}$, respectively. This process can be described as follows,

$$\begin{aligned} z_s = \mathcal{F}(X_s), \quad z_t = \mathcal{F}(X_s) \\ \hat{X}_{t2s} = \mathcal{G}_s(z_t), \quad \hat{X}_{s2s} = \mathcal{G}_s(z_s). \end{aligned} \tag{1}$$

For the target domain discriminator D_t, a binary classifier based on the cross entropy loss is used. The generator network $\mathcal{G}_t$ generates the image samples from the source and the target domain, using latent representations z_s and z_t, respectively. This process can be described as follows,

$$\hat{X}_{s2t} \;=\; \mathcal{G}_t(z_s), \quad \hat{X}_{t2t} \;=\; \mathcal{G}_t(z_t). \tag{2}$$

The classifier loss function can be defined as follows

$$\mathcal{L}_{ce} = \; \mathbb{E}_{\{X,y\}\sim\{\mathcal{D}_s,\mathcal{Y}_s\}}[\ell_{ce}(\mathcal{C}(\mathcal{F}(X)), y)], \tag{3}$$

where, $\mathcal{L}_{ce}$ is the overall classification loss computed on the labeled source data and ℓ_{ce} is the categorical cross entropy loss. Considering $\hat{y} = \mathcal{C}(z_s)$ as the predicted probability vector, ℓ_{ce} can be expressed as follows

$$\ell_{ce}(\hat{y}, y) \;=\; -\sum_{j=1}^{C} \; y_j \;\; \log[\hat{y}_j]. \tag{4}$$

To train the source discriminator in the conditional GAN framework, we need to perform real/fake classification and categorical classification, which can be

expressed as

$$\mathcal{L}_{cGAN}^{D_s} = \mathbb{E}_{X\sim\mathcal{D}_s}[\log(1 - D_s^b(X))] + \mathbb{E}_{X\sim\mathcal{D}_s}[\log(D_s^b(\hat{X}_{t2s}))] + \mathbb{E}_{X\sim\mathcal{D}_t^k}[\log(D_s^b(\hat{X}_{s2s}))] + \mathbb{E}_{X\sim\mathcal{D}_s, y\sim\mathcal{Y}_s}[\ell_{ce}(D_s^a(\hat{X}_{s2s}), y)], \quad (5)$$

where, the first term in the equation trains the discriminator D_s^b to identify data sampled from the source dataset $\mathcal{D}_s$ as real images. The second and third term train the discriminator to identify images generated by $\mathbb{G}_s$, i.e., $\hat{X}_{t2s}$ and $\hat{X}_{s2s}$, as fake. The fourth term is a classification loss similar to Eq. 3, where the generated images $\hat{X}_{s2s}$ are classified in to the category corresponding to the source input images using D_s^a.

After the discriminator update, the source generator is trained to generate images such that the discriminator network is fooled into identifying the generated images, $\hat{X}_{s2s}$ and $\hat{X}_{t2s}$ as real source images. To further improve the image generation quality, we add L1 reconstruction loss, denoted as ℓ_r, on the generated source images, $\hat{X}_{s2s}$. The loss functions described above can be mathematically formulated as

$$\mathcal{L}_{cGAN}^{\mathcal{G}_s} = \mathbb{E}_{X\sim\mathcal{D}_s}[\log(1 - D_s^b(\mathcal{G}_s(X)))] + \mathbb{E}_{X\sim\mathcal{D}_t^k}[\log(1 - D_s^b(\mathcal{G}_s(X)))], \quad (6)$$

$$\mathcal{L}_{rs}^{\mathcal{G}_s} = \mathbb{E}_{X\sim\mathcal{D}_s}[\ell_r(\hat{X}_{s2s}, X)], \quad (7)$$

where

$$\ell_r(\hat{X}, X) = \|X - \hat{X}\|_1. \quad (8)$$

Similar to the source domain discriminator and generator, we apply the same GAN losses for the target domain discriminator D_t, and generator $\mathcal{G}_t$. Since, the target domain labels are not available, a traditional GAN formulation is used [12], instead of the conditional GAN formulation [33] used for source domain. Additionally, similar to the source domain, we add L1 reconstruction loss on the generated target images, $\hat{X}_{t2t}$, to further improve the image generation quality in the target domain. These losses can be written as follows

$$\mathcal{L}_{GAN}^{D_t} = \mathbb{E}_{X\sim\mathcal{D}_t}[\log(1 - D_t(X))] + \mathbb{E}_{X\sim\mathcal{D}_s}[\log(D_t(\hat{X}_{s2t}))] + \mathbb{E}_{X\sim\mathcal{D}_t^k}[\log(D_t(\hat{X}_{t2t}))], \quad (9)$$

$$\mathcal{L}_{GAN}^{\mathcal{G}_t} = \mathbb{E}_{X\sim\mathcal{D}_t^k}[\log(1 - D_t(\mathcal{G}_t(X)))] + \mathbb{E}_{X\sim\mathcal{D}_s}[\log(1 - D_t(\mathcal{G}_t(X)))], \quad (10)$$

$$\mathcal{L}_{rt}^{\mathcal{G}_t} = \mathbb{E}_{X\sim\mathcal{D}_t^k}[\ell_r(\hat{X}_{t2t}, X)]. \quad (11)$$

Finally, the loss function for the feature encoder network consists of both the classification loss on the source and the adaptation loss from the conditional GAN module. The final loss for the network $\mathcal{F}$ can be expressed as

$$\mathcal{L}_{total}^{\mathcal{F}} = \mathcal{L}_{ce} + \lambda_1 \mathcal{L}_{cGAN}^{\mathcal{G}_s} + \lambda_2 \mathcal{L}_{GAN}^{\mathcal{G}_t}, \quad (12)$$

Fig. 4: Sample images from the datasets used for conducting experiments. (a) Digits (b) Office-31.

where λ_1 and λ_2 are parameters. The loss functions defined above, $\mathcal{L}_{cGAN}^{\mathcal{G}_s}$, $\mathcal{L}_{cGAN}^{D_s}$, $\mathcal{L}_{GAN}^{\mathcal{G}_t}$, $\mathcal{L}_{GAN}^{D_t}$, $\mathcal{L}_{ce}^{\mathcal{C}}$, $\mathcal{L}_{total}^{\mathcal{F}}$, $\mathcal{L}_{rt}^{\mathcal{G}_t}$ and $\mathcal{L}_{rs}^{\mathcal{G}_s}$, are minimized iteratively to update the parameters of their respective networks. The overall training procedure for the proposed method is summarized in Algorithm 1.

5 Experiments and Results

For experiments, we consider all the baseline methods discussed in Sec. 4.2 and the proposed method described in Sec. 4.3. We use SVHN [32], MNIST [23] and USPS [18] digit recognition datasets, as well as the Office-31 [43] object recognition datasets to conduct experiments (see Fig. 4). We evaluate the performance of different methods using the Area Under the ROC (AUROC) Curve metric, which is the most commonly used evaluation metric for novelty detection. Each datasets are divided into known and novel categories for novelty detection.

Algorithm 1 Pseudocode for Training Proposed Method

Require: Network models $\mathcal{F}$, $\mathcal{C}$, $\mathcal{G}_s$, $\mathcal{D}_s$, $\mathcal{G}_t$, $\mathcal{D}_s$
Require: Initial parameters Θ_f, Θ_c, Θ_{g_s}, Θ_{d_s}, Θ_{g_t}, Θ_{d_t}
Require: Source data, $\mathcal{D}_s$, $\mathcal{Y}_s$ Target data , $\mathcal{D}_t^k$
Require: Hyper-parameters : N, lr, λ_1, λ_2
while not done **do**
  for each batch with size N **do**
    for i = 1 to N **do**
      Feed-forward using Eq. (1) – Eq. (2)
    end for
    Calculate Losses based on Eq. (3) – Eq.(12)
    Update Θ_{d_s}, $\Theta_{d_s} \leftarrow \Theta_{d_s} - lr * \nabla_{\Theta_{d_s}} \mathcal{L}_{cGAN}^{D_s}$
    Update Θ_{d_t}, $\Theta_{d_t} \leftarrow \Theta_{d_t} - lr * \nabla_{\Theta_{d_t}} \mathcal{L}_{GAN}^{D_t}$
    Update Θ_{g_s}, $\Theta_{g_s} \leftarrow \Theta_{g_s} - lr * \nabla_{\Theta_{g_s}} \mathcal{L}_{cGAN}^{G_s}$
    Update Θ_f, $\Theta_f \leftarrow \Theta_f - lr * \nabla_{\Theta_f} \mathcal{L}_{total}^{\mathcal{F}}$
    Update Θ_c, $\Theta_c \leftarrow \Theta_c - lr * \nabla_{\Theta_c} \mathcal{L}_{ce}$
    Update Θ_{g_t}, $\Theta_{g_t} \leftarrow \Theta_{g_t} - lr * \nabla_{\Theta_{g_t}} \mathcal{L}_{rt}^{G_t}$
    Update Θ_{g_s}, $\Theta_{g_s} \leftarrow \Theta_{g_s} - lr * \nabla_{\Theta_{g_s}} \mathcal{L}_{rs}^{G_s}$
  end for
end while
Output: Learned parameters $\hat{\Theta}_f$,$\hat{\Theta}_s$,$\hat{\Theta}_{g_s}$,$\hat{\Theta}_{d_s}$,$\hat{\Theta}_{d_t}$,$\hat{\Theta}_{g_t}$

Details regarding the splits are described in the following sections. The novel categories are not utilized during training and only used during inference. The following methods are comapred.

• **Softmax baseline:** In this baseline, only the feature extractor network $\mathcal{F}$ and the classification network $\mathcal{C}$ are trained on the labeled source dataset using the cross entropy loss. This is the simplest baseline and follows the traditional CNN training for recognition. Maximum softmax probability score is used for novelty detection.
• **ALOCC:** ALOCC is a method proposed in [42], which utilizes a feature extractor network $\mathcal{F}$ and a decoder network $\mathcal{G}$ supervised in a generative adversarial framework with the help of a discriminator network D_i. The training is done directly on the unlabeled target data. The input is injected with a Gaussian noise η and networks $\mathcal{F}$ and $\mathcal{G}$ are forced to reconstruct a clean image. The network parameters are learned by optimizing a combination of GAN and reconstruction losses. The discriminator score of the reconstructed input $D(\mathcal{G}(\mathcal{F}(X + \eta)))$ is used for novelty detection.
• **GRL:** Gradient reversal baseline extends the softmax baseline by improving the feature space to be domain invariant. This makes the maximum softmax probability much more reliable for the novelty detection task on the target domain. For GRL, feature extractor $\mathcal{F}$ and classifier network $\mathcal{C}$ are trained using the cross entropy loss and domain classifier D_f is employed with a gradient reversal layer [11] to enforce the feature space to be domain invariant. Here, the method utilizes both labeled source data and unlabeled target data for training the network parameters.
• **ALOCC+GRL:** ALOCC+GRL combines the two method described above in an ad-hoc fashion. The ALOCC training is done as described above, which involves reconstructing a clean image when the input to the network is injected with Gaussian noise. For this baseline we add noise to both source and target data. The feature extractor network $\mathcal{F}$ is also trained to perform classification of labeled source data through classification network $\mathcal{C}$. Additionally, the feature space of network $\mathcal{F}$ is enforced to be domain invariant through domain classifier D_f and gradient reversal layer. Combination of scores from ALOCC and maximum softmax probability is used to perform novelty detection. The training utilizes both labeled source and unlabeled target data.
• **Proposed method:** The proposed method is used as described in Sec. 4.3. We use addition of maximum softmax probability scores and loss from target generator (i.e. discriminator score of generated image and reconstruction loss) for novelty detection.

In all experiments, we use Adam optimizer [20] with the learning rate (η) of 0.0001 and batch size (N) of 64. The hyper-parameter λ_1 and λ_2 are both set equal to 0.03. The parameters are chosen using validation performance from the source domain data. Details regarding the network architectures used for $\mathcal{F}$, $\mathcal{C}$, $\mathcal{G}_s$, $\mathcal{G}_t$, D_s and D_t are provided in supplementary material.

Method	SVHN→MNIST	MNIST→USPS	USPS→MNIST	SVHN→USPS	Average Performance
Softmax (S)	0.642	0.602	0.651	0.587	0.620
ALOCC (T)	0.702	0.633	0.702	0.633	0.667
GRL (ST)	0.718	0.863	0.859	0.667	0.776
ALOCC+GRL (ST)	0.851	0.903	0.895	0.845	0.873
Proposed (ST)	**0.919**	**0.945**	**0.928**	**0.895**	**0.921**

Table 1: Performance on the digits datasets - SVHN, MNIST and USPS evaluated using area under the roc metric. (S), (T) and (ST) respectively denote only labeled source data, only unlabeled target data and both labeled source-unlabeled target data used for training.

Digits: SVHN, USPS, MNIST In the first set of experiments, SVHN, USPS and MNIST digit datasets are used to create four different scenarios, SVHN→MNIST, SVHN→USPS, USPS→MNIST and MNIST→USPS. First five digits, digits 0 to 4, are used as known categories and the remaining digits, digit 5 to 9, are considered as novel categories. Only the known categories are used during training and novel categories are used only for evaluating the methods. For the problem setting proposed in this paper, we utilize training split provided by the respective datasets to train the models and test split are used for evaluating the performance. All images in SVHN, MNIST and USPS are resized to 32×32. The feature extractor used in this paper is inspired from the LeNet architecture [22] (details are provided in supplementary material).

The performance of each method is reported in the Table. 1. The softmax baseline performs worst out of all the methods. This is expected as softmax baseline is trained on only labeled source dataset. Also, it is not specifically trained for the novelty detection task. ALOCC performs better than softmax as it is trained on the target dataset and is specifically designed for the task of novelty detection. GRL baseline learns a domain invariant feature encoder, and hence is able to produce reasonable softmax probabilities on the target dataset. ALOCC+GRL combines the ideas from domain adversarial training and novelty detection training. Specifically, ALOCC learns a good model for novelty detection task and GRL helps the feature extractor of the ALOCC model to learn domain invariant feature. Additional training with classification loss on the labeled source data helps the ALOCC+GRL to better utilize multi-class structure of the dataset, making it the best performing method among the baselines. All of the above methods are simple extensions or ad-hoc combinations of the work available in the literature. Whereas, the proposed approach tackles the distribution shift issue along with novelty detection training in a single model. This helps the proposed approach perform better than the ad-hoc solutions, performing $\sim 5\%$ better than ALOCC+GRL.

Office31 : Amazon, Webcam, DSLR Finally, we evaluate the proposed method on the Office31 benchmark [43]. The Office31 benchmark has a total 31 object categories and three different domains. Image samples for the dataset are acquired in three different domains, i.e. Amazon (A), Webcam (W) and DSLR

Methods	A→D	A→W	W→A	W→D	D→A	D→W	Average
Softmax	0.719	0.835	0.655	0.862	0.606	0.842	0.737
ALOCC	0.776	0.725	0.608	0.983	0.570	0.884	0.758
GRL	0.766	0.730	0.624	**0.988**	0.572	0.890	0.762
ALOCC+GRL	0.783	0.759	0.640	0.987	0.576	0.898	0.774
Proposed	**0.877**	**0.863**	**0.824**	0.938	**0.807**	**0.940**	**0.877**

Table 2: AUC performance of different methods on the Office31 [43] datasset.

(D). First 10 categories from all three domains are considered as known. Categories from $11, 12, ..., 30$ are considered as novel categories for all domains. For all the methods compared, AlexNet [21] is used as the base feature extractor. During training we freeze all the convolutional layers of AlexNet and only fine tune the fully-connected layers. For training the generator networks $\mathcal{G}_s$ and $\mathcal{G}_t$ we resize the images to 32×32 and the discriminator architectures are used accordingly (more details in supplementary material). Three domains of the dataset form in total 6 pairs of source→target combinations. For each source→target combination, we report AUROC performance.

The peformance of each method is reported in Table 2. Overall the trend of performance improvements are similar to the digits experiment. Among all the methods, softmax baseline achieves the lowest performance. ALOCC improves by $\sim 2\%$ over the softmax baseline, while GRL is able to improve $\sim 1\%$ over ALOCC. Utilizing gradient reversal along with ALOCC training further improves the performance by $\sim 1\%$. The proposed approach on average performs better than the other approaches. Specifically, the proposed approach on average provides $\sim 9\%$ improvement over the next best baseline of ALOCC+GRL.

6 Conclusion

We considered the problem of novelty detection under dataset distribution shift and showed the challenges it poses with experiments. To the best of our knowledge, this is the first work to address such problem for novelty detection. We also discussed the differences between the proposed problem setting and some of the related problems like open-set domain adaptation. We also developed a few trivial baseline methods based on the related works available in the literature by combining the techniques from novelty detection and domain adaptation. Finally, we proposed an approach to tackle the distribution shift by learning a shared feature space that can generalize better in comparison with the baseline methods.

Acknowledgement

This work was supported by the NSF grant 1910141.

References

1. Amarbayasgalan, T., Jargalsaikhan, B., Ryu, K.H.: Unsupervised novelty detection using deep autoencoders with density based clustering. Applied Sciences **8**(9), 1468 (2018) 4
2. Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., Mané, D.: Concrete problems in ai safety. arXiv preprint arXiv:1606.06565 (2016) 1
3. 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) 2
4. 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) 2
5. Cao, Z., Ma, L., Long, M., Wang, J.: Partial adversarial domain adaptation. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 135–150 (2018) 5
6. 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
7. Chen, C., Yuan, W., Xie, Y., Qu, Y., Tao, Y., Song, H., Ma, L.: Novelty detection via non-adversarial generative network. arXiv preprint arXiv:2002.00522 (2020) 1, 2, 4
8. Chen, X., Liu, C., Li, B., Lu, K., Song, D.: Targeted backdoor attacks on deep learning systems using data poisoning. arXiv preprint arXiv:1712.05526 (2017) 1
9. Chen, Y., Li, W., Sakaridis, C., Dai, D., Van Gool, L.: Domain adaptive faster r-cnn for object detection in the wild. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3339–3348 (2018) 3
10. DeVries, T., Taylor, G.W.: Learning confidence for out-of-distribution detection in neural networks. arXiv preprint arXiv:1802.04865 (2018) 1
11. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. arXiv preprint arXiv:1409.7495 (2014) 3, 4, 7, 8, 12
12. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014) 4, 10
13. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. arXiv preprint arXiv:1610.02136 (2016) 1
14. Hoffman, J., Tzeng, E., Park, T., Zhu, J.Y., Isola, P., Saenko, K., Efros, A.A., Darrell, T.: Cycada: Cycle-consistent adversarial domain adaptation. arXiv preprint arXiv:1711.03213 (2017) 3, 4
15. Hoffman, J., Wang, D., Yu, F., Darrell, T.: Fcns in the wild: Pixel-level adversarial and constraint-based adaptation 3
16. Hsu, H.K., Yao, C.H., Tsai, Y.H., Hung, W.C., Tseng, H.Y., Singh, M., Yang, M.H.: Progressive domain adaptation for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 1–5 (2019) 3
17. Hu, L., Kan, M., Shan, S., Chen, X.: Duplex generative adversarial network for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1498–1507 (2018) 4
18. Hull, J.J.: A database for handwritten text recognition research. IEEE Transactions on pattern analysis and machine intelligence **16**(5), 550–554 (1994) 11

19. Kim, T., Jeong, M., Kim, S., Choi, S., Kim, C.: Diversify and match: A domain adaptive representation learning paradigm for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 12456–12465 (2019) 3
20. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2015) 12
21. 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) 14
22. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998) 13
23. LeCun, Y., Cortes, C., Burges, C.: Mnist handwritten digit database (2010) 11
24. 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
25. Liu, K., Dolan-Gavitt, B., Garg, S.: Fine-pruning: Defending against backdooring attacks on deep neural networks. In: International Symposium on Research in Attacks, Intrusions, and Defenses. pp. 273–294. Springer (2018) 1
26. Liu, Y., Xie, Y., Srivastava, A.: Neural trojans. In: 2017 IEEE International Conference on Computer Design (ICCD). pp. 45–48. IEEE (2017) 1
27. Long, M., Zhu, H., Wang, J., Jordan, M.I.: Unsupervised domain adaptation with residual transfer networks. In: Advances in Neural Information Processing Systems. pp. 136–144 (2016) 4
28. Long, M., Zhu, H., Wang, J., Jordan, M.I.: Deep transfer learning with joint adaptation networks. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 2208–2217. JMLR. org (2017) 4
29. Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., Frey, B.: Adversarial autoencoders. arXiv preprint arXiv:1511.05644 (2015) 4
30. Markou, M., Singh, S.: Novelty detection: a review—part 1: statistical approaches. Signal processing **83**(12), 2481–2497 (2003) 4
31. Murez, Z., Kolouri, S., Kriegman, D., Ramamoorthi, R., Kim, K.: Image to image translation for domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4500–4509 (2018) 4
32. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Reading digits in natural images with unsupervised feature learning (2011) 11
33. Odena, A., Olah, C., Shlens, J.: Conditional image synthesis with auxiliary classifier gans. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 2642–2651. JMLR. org (2017) 9, 10
34. Oza, P., Patel, V.M.: One-class convolutional neural network. IEEE Signal Processing Letters **26**(2), 277–281 (2018) 2
35. Oza, P., Patel, V.M.: Utilizing patch-level category activation patterns for multiple class novelty detection. In: European Conference on Computer Vision. Springer (2020) 1
36. Panareda Busto, P., Gall, J.: Open set domain adaptation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 754–763 (2017) 3, 5
37. 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
38. 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) 4

39. Perera, P., Patel, V.M.: Learning deep features for one-class classification. IEEE Transactions on Image Processing **28**(11), 5450–5463 (2019) 1
40. Pidhorskyi, S., Almohsen, R., Doretto, G.: Generative probabilistic novelty detection with adversarial autoencoders. In: Advances in neural information processing systems. pp. 6822–6833 (2018) 1, 2, 4
41. Pinheiro, P.O.: Unsupervised domain adaptation with similarity learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8004–8013 (2018) 4
42. Sabokrou, M., Khalooei, M., Fathy, M., Adeli, E.: Adversarially learned one-class classifier for novelty detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3379–3388 (2018) 1, 2, 4, 5, 7, 8, 12
43. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: European conference on computer vision. pp. 213–226. Springer (2010) 11, 13, 14
44. Saito, K., Watanabe, K., Ushiku, Y., Harada, T.: Maximum classifier discrepancy for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3723–3732 (2018) 4
45. Sankaranarayanan, S., Balaji, Y., Castillo, C.D., Chellappa, R.: Generate to adapt: Aligning domains using generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8503–8512 (2018) 4
46. Schölkopf, B., Smola, A.J., Bach, F., et al.: Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT press (2002) 4
47. Shao, R., Perera, P., Yuen, P.C., Patel, V.M.: Open-set adversarial defense. In: European Conference on Computer Vision. Springer (2020) 1
48. Shu, R., Bui, H.H., Narui, H., Ermon, S.: A dirt-t approach to unsupervised domain adaptation. arXiv preprint arXiv:1802.08735 (2018) 4
49. Sun, B., Saenko, K.: Deep coral: Correlation alignment for deep domain adaptation. In: European Conference on Computer Vision. pp. 443–450. Springer (2016) 3
50. Tsai, Y.H., Hung, W.C., Schulter, S., Sohn, K., Yang, M.H., Chandraker, M.: Learning to adapt structured output space for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7472–7481 (2018) 3
51. Turk, M., Pentland, A.: Face recognition using eigenfaces. In: Proceedings. 1991 IEEE computer society conference on computer vision and pattern recognition. pp. 586–587 (1991) 4
52. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7167–7176 (2017) 3, 4
53. You, K., Long, M., Cao, Z., Wang, J., Jordan, M.I.: Universal domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2720–2729 (2019) 5
54. Zhang, H., Patel, V.M.: Sparse representation-based open set recognition. IEEE transactions on pattern analysis and machine intelligence **39**(8), 1690–1696 (2016) 1
55. Zhang, Y., David, P., Foroosh, H., Gong, B.: A curriculum domain adaptation approach to the semantic segmentation of urban scenes. IEEE transactions on pattern analysis and machine intelligence (2019) 3