Anomaly Detection via Few-shot Learning on Normality

Shin Ando[1] and Ayaka Yamamoto[2]

School of Management, Tokyo University of Science
ando@rs.tus.ac.jp[1], 8620510@ed.tus.ac.jp[2]

Abstract. One of the basic ideas for anomaly detection is to describe an enclosing boundary of normal data in order to identify cases outside as anomalies. In practice, however, normal data can consist of multiple classes, in which case the anomalies may appear not only outside such an enclosure but also in-between 'normal' classes. This paper addresses deep anomaly detection aimed at embedding 'normal' classes to individually close but mutually distant proximities. We introduce a problem setting where a limited number of labeled examples from each 'normal' class is available for training. Preparing such examples is much more feasible in practice than collecting examples of anomalies or labeling large-scale, normal data. We utilize the labeled examples in a margin-based loss reflecting the inter-class and the intra-class distances among the embedded labeled data. The two terms and their relations are derived from an information-theoretic principle. In an empirical study using image benchmark datasets, we show the advantage of the proposed method over existing deep anomaly detection models. We also show case studies using low-dimensional mappings to analyze the behavior of the proposed method.

Keywords: Deep Anomaly Detection · Generative Adversarial Networks · Deep One-class Classification · Data Description · Few-shot Learning

1 Introduction

Deep anomaly detection (DAD) [8, 11] has received strong interests in recent years, but remains to be among the challenging tasks for deep learning. A basic goal in DAD is to find a compact representation of the data observed under 'normality' such that unobserved 'anomalies' are more likely to be distant or exhibit strong discrepancy from them.

GAN-based anomaly detection is a category of DAD, which learns the manifold of normal data distribution and identifies anomalies primarily based on the error between the original and an image reconstructed through a generator network [14, 18, 1]. The deep data description models [12, 13, 9] are extensions of one-class classification and support vector data description [15]. They form a category of DAD which learns an embedding function and a data-enclosing hypersphere with the minimum volume in the embedded space, with an implicit assumption that normal data comes from a single class or source. At testing, the anomaly score is determined by the distance from the center of the hypersphere.

In practice, however, normal data can consist of multiple classes, and the anomalies may appear not only outside its boundary but also in-between 'normal' classes. In such cases, the conventional approach to find a single enclosure of normal data may increase the possibility of detecting anomalies outside, but it can also increase the possibility of overlooking anomalies between classes. In this paper, we alternatively attempt to find an embedding where each class is condensed to a proximity, but at the same time mutually distant and dispersed. It allows for a unified approach to detecting anomalies, as cases which appear far from the nearby normal classes.

We propose a framework utilizing a small number of labeled examples, or prototypes, from each 'normal' class. Practically, preparing a limited number of labeled data is far less expensive than collecting examples of anomalies or labeling large-scale normal data. The prototypes are used in a tune-up training, after a pre-training using large-scale unlabeled data by generative adversarial networks. This input setting differs from semi-supervised anomaly detection [13], which takes few examples of anomalies for calibrating anomaly scores, and also from few-shot learning [7] and out-of-distribution detection [9] which exploit a large-scale, labeled dataset from related tasks.

The training in the proposed framework is driven by an information-theoretic principle, which can formalize deep representation learning as a reduction of intra-class distances and an expansion of the inter-class distances at a trade-off. We propose a margin-based loss, which penalizes prototype pairs which increase intra-class margins or reduce inter-class margins. We conduct an empirical study to evaluate the proposed framework in comparison to existing DAD models and to analyze its embedding of the normal classes.

The main contribution of this paper is two-fold: (1) an anomaly detection framework under a new setting, utilizing small-scale, labeled normal data which are not practically expensive, (2) a margin-based loss derived from an information-theoretic principle to integrate small-scale labeled data into deep representation learning.

The rest of this paper is organized as follows. Section 2 describes the previous studies on deep anomaly detection and the relation between the information bottleneck and deep learning. Section 4 describes the technical details of the proposed framework. Section 5 presents the empirical results and the analyses from our experiments using public image datasets. We state our conclusion in Section 6.

2 Related Work

2.1 Deep Anomaly Detection

Two primary purposes of deep learning models in anomaly detection frameworks are: (1) reconstructing test samples and (2) providing distance metrics in the embedded space. The examples of (1) include deep autoencoders and GANs [6]. In AnoGAN [14], the generator network G learns the normal data distribution manifold from which a test image X is reconstructed. The anomaly score is given by the residual difference after minimizing the absolute difference with the generated image X'. GANomaly [1], and Efficient-GAN based Anomaly Detection [20] similarly uses the reconstruction loss, after mapping the test image to and from the embedded space using the encoder and the generator networks, as anomaly scores.

The basis of measuring anomalousness by reconstruction error is that the trained generator acquires a mapping from a uniform distribution to the normal data distribution manifold [14, 1]. In cases that the test sample is an anomaly, the image reconstructed by the generator should naturally deviate from the original and towards the normal data distribution.

Deep-SVDD [12] is an extension of the support vector data description [15] and an example of (2). It aims to learn an embedding in which the normal data can be enclosed by a hyper-spherical boundary. The boundary defines a one-class classifier, which identifies outliers based on the distance from its center. Deep multi-sphere SVDD (DMSVDD) extended the idea to learn multiple hyperspheres, to addressed anomalies among multiple classes of normal data[5].

Multi-class Data Description (MCDD) [9] exploits the Deep SVDD model for out-of-distribution detection (OOD). It trains a DNN such that the embedding function f maps the labeled data onto the proximity of the centers of corresponding classes. The in-distribution classes are modeled as Gaussian components in the embedded space. Deep SVDD employs a max-margin loss for training, while in MCDD, implementations with a max-margin loss and a GDA-based MAP loss were introduced.

In our proposed model, we use GANs for pre-training an initial embedding, and a margin-based loss for fine-tuning the embedding for anomaly detection.

2.2 Information Bottleneck

The information bottleneck (IB) [17, 2] is a principle for signal encoding to achieve a larger compression rate and a smaller distortion. It was adopted to machine learning for finding a sparse representation of an input variable X, which maintains the predictive power over an output variable Y. The sparseness and the predictive power of the representation Z are measured by its statistical dependence, i.e., mutual information, with respect to X and Y, respectively.

The IB principle is formalized as a minimization problem over a Lagrangian

$$\mathcal{L} = I(X;Z) - \beta I(Y;Z) \tag{1}$$

where $I(Y;Z)$ quantifies the amount of relevant information on Y. Since Z is generated from X, $I(X;Z)$ decreases as the rate of compression increases. The multiplier β represents the trade-off between the two terms.

In [16], the IB principle was introduced to analyse the layer-wise compression efficiency in DNNs. It was also employed in [13] for deriving a semi-supervised training loss.

[3] presented an analysis of the IB problem in a case where Y is the class variable and Z is the d-dimensional deep representation from the embedding function $f : \mathcal{X} \to \mathcal{Z} \subset \mathbb{R}^d$, with several modeling assumptions on Z. The first assumption is that the conditional distribution $p(z|y)$ is an isotropic Gaussian component for each class y, i.e.,

$$\begin{aligned} p(z|y) &= \mathcal{N}(z; \mu_y, \sigma_y I) \\ &= \frac{1}{(2\pi\sigma_y^2)^{d/2}} \exp\left(-\frac{\|z-\mu_y\|^2}{2\sigma_y^2}\right) \end{aligned}$$

where μ_y and σ_y denotes the class mean and standard deviation, respectively. The marginal distribution $p(z)$ is empirically approximated as an average of the Dirac delta functions

$$p(z) = \frac{1}{N}\sum_{i=1}^{N} \delta\left(z - f(x_i)\right)$$

The mutual information $I(Y;Z)$, which is equivalent to the expected Kullback-Leibler divergence between $p(y|z)$ and $p(z)$, is then rewritten as

$$\begin{aligned} I(Y;Z) &= E_{y,z}\left[\log \frac{p(z|y)}{p(z)}\right] \\ &= \sum_{y=1}^{K} \frac{1}{n_y} \sum_{z} \log \frac{\exp\left(-\frac{\|z-\mu_y\|^2}{2\sigma_y}\right) - \log \sigma_{y'}^d}{\sum_{y'} \exp\left(-\frac{\|z-\mu_{y'}\|^2}{2\sigma_{y'}}\right) - \log \sigma_{y'}^d} + \text{const.} \end{aligned} \qquad (2)$$

$p(z|x)$ was defined as a probability that x is mapped to z, given the randomness of DNN such as batch normalization and dropouts. It was also modeled by an isotropic Gaussian component centered at $f(x)$ with a common standard deviation $\hat{\sigma}$.

$$\begin{aligned} p(z|x) &= \mathcal{N}(z|f(x), \hat{\sigma}^2 I) \\ &= \frac{1}{(2\pi\hat{\sigma}^2)^{d/2}} \exp \frac{\|z-f(x)\|^2}{2\hat{\sigma}^2} \end{aligned}$$

The mutual information $I(X;Z)$ then was rewritten as

$$\begin{aligned} I(X;Z) &= E_{x,z}\left[\log \frac{p(z|x)}{p(z)}\right] \\ &= \frac{1}{N^2} \sum_{z\in\mathcal{Z}} \sum_{i=1}^{N} \left(-\frac{\|z-f(x_i)\|^2}{2\hat{\sigma}^2} + \log \hat{\sigma}^d\right) + \text{const.} \end{aligned} \qquad (3)$$

Based on (3), $I(X;Z)$ was approximated as the sum of mutual distances in the embedded space. From (2), $I(Y;Z)$ increases as the class distribution concentrates to its center, which broadly interprets as the reduction of the intra-class distances. It was argued that by jointly minimizing $I(X;Z)$ and $I(Y;Z)$, the

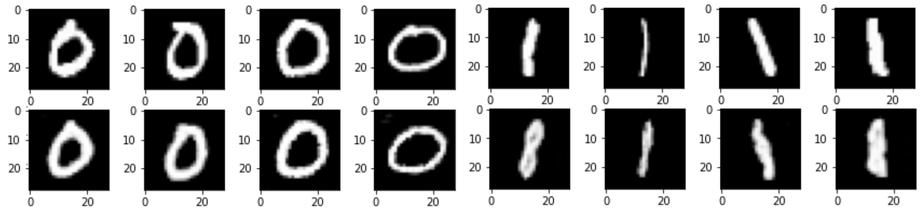


Fig. 1. Original and reconstructed images

inter-class mutual distances will increase while the intra-class distances decrease, by virtue of substantially larger class deviations σ_y compared to the deviation of the randomness, $\hat{\sigma}$, after training.

Motivated by the above analysis from [3], we implement a margin-based loss with a focus on the inter-class and intra-class properties for deep anomaly detection.

3 Motivating Example

In this section, we examine the behavior of GAN-based anomaly detection to motivate the proposed framework. Adversarial training has several attractive properties for anomaly detection. For example, GANs can learn deep representation from 'normal' data in an unsupervised manner. A trained generator network can be used to 'reconstruct' a test case, and its 'error' from the original case can provide a natural anomaly score as mentioned in Section 2.1.

In the following, we describe a DAD process using the BiGAN [4] framework, comprised of a generator, discriminator, and an encoder, used in EGBAD [19] and GANomaly [1]. We conducted an unsupervised, adversarial training of BiGAN in a standard setup for anomaly detection using the MNIST benchmark, which is to remove one class designated as an 'anomaly' class from training and compile the 'normal' data from the remaining classes.

After training, the test cases were initially mapped to a Euclidean space and reconstructed back to an image using the encoder and the generator networks. The examples of the original and the reconstructed images, from the setup that the digit '1' was designated as anomalies, are shown in Fig. 1.

The original images are shown in the first row while the reconstructed images are shown in the second row. The images shown on the left half is those of digit '0', a normal class, and the images on the right half are those of digit '1', the anomaly class, respectively.

Graphically, the reconstructed normal images resemble natural handwriting while the reconstructed anomaly images exhibit unnatural forms, with subtle resemblance of other digits. In terms of the pixel-wise comparison, however, the reconstructed 'anomalous' images, are not substantially different from their originals. Meanwhile, the reconstructed normal images exhibit slight modifications from their originals.

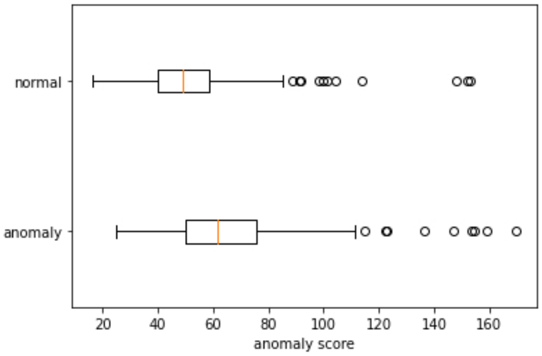


Fig. 2. Anomaly score distributions

Fig. 2 compares the distribution of mean absolute errors between the original and the reconstructed test images belonging to the normal and the anomalous classes. There is a notable overlap of interquartiles between the two distributions. In this case, it is therefore unlikely that the reconstruction errors as anomaly scores produce a good detection performance.

The graphical results suggests that the learned manifold can include intermediate patterns of different classes in the training data, since GANs learn a mapping between a continuous, Euclidean unit space and a distribution manifold. With a large variety in normal data, intermediate patterns may allow a reconstruction of anomalous cases without significant error. We should also beware of the class-wise bias over the reconstruction error, producing relatively higher anomaly scores for classes with larger and more complex patterns.

Based on these preliminary analyses, we avoid the reconstruction and pixel-wise comparison process but instead were motivated to find an embedding in which the anomalies can be detected by the distances from the nearby 'normal' classes, thus consider a setting where typical examples of normal classes can be utilized.

4 Prototype Data Description

In the previous section, we introduced our motivation to utilize labeled examples of normal data into training and exploit distances in the embedded space in testing. This section describes the framework which integrates these examples into training based on an information-theoretic principle.

Generally, a set of normal data for training is large-scale and unlabeled as the cost of observation under normality is small, but the reward for labeling such data is also small to none. However, it can be feasible to collect a limited number of examples from each class. Here, we assume such a small-scale dataset comprised of K samples from each of N classes, much like the setting of N-way-K-shot learning, is available. The problem input thus consists of a large-scale, unlabeled normal dataset available for pre-training, and a set of labeled examples for the tune-up training.

Let us denote the unlabeled dataset by $\mathcal{X} = \{x_i\}_{i=1}^{M}$ and the set of labeled data, or prototypes, by $\mathcal{P} = \{(p_j, y_j)\}_{j=1}^{K \times N}$. N and K denotes the number of classes and the number of prototypes for each class, respectively.

We represent by random variables X and Z, the structured input data, e.g., images, and their the embedding, respectively. The class variable Y takes a value from $\mathcal{Y} = \{1, \dots, N\}$. We denote the embedding function of a DNN with parameters W by $f : \mathcal{X} \to \mathcal{Z}$.

As referenced in Section 2.2, minimizing the information bottleneck loss interprets to expanding inter-class distances and reducing intra-class distances at a trade-off. The intra-class distances, represented as (2), are measured with regards to class means and variances, reflecting the modeling assumption that the class distributions are Gaussian components. For the task at hand, however, the estimated parameters may not be robust given the small scale of the labeled data.

Alternatively, we attempt to reduce the diameter of the class-enclosing convex. Let R_c^* denote the largest intra-class distance among samples of class c,

$$R_c^* = \max_{k,j:y_k=y_j=c} \|f(p_j;W) - f(p_k;W)\| \tag{4}$$

As R_c^* is equivalent to the diameter of the convex hull of the samples of c, we can minimize its volume and subsequently the intra-class distances, by descending along the gradients of R_c^* with regards to W. Note that the small scale of the labeled data allows for computing the mutual distance matrix in a feasible time. Still, it is inefficient to iterate the descent and the update for the single largest distance, R_c^*. We, instead, take the sum of intra-class, pair-wise distances over a threshold R_{intra}, as a loss $\mathcal{L}_{\text{intra}}(W)$.

$$\mathcal{L}_{\text{intra}}(W) = \sum_{j,k:y_j=y_k} \max\{0, \|f(p_j;W) - f(p_k;W)\| - R_{\text{intra}}\} \tag{5}$$

By setting R_{intra} to a q_{intra}-quantile over the intra-class distances, (5) takes a summation over the largest $q \times N$ intra-class pairs. Using the gradients of (5) achieves a substantially faster descent compared to using that of R_c^*.

With regards to the inter-class distances, we look at (3), which can be approximated by the sum of all pair-wise distances in the embedded space. The constants and the terms related to the variance $\hat{\sigma}^2$ are irrelevant to the optimization and can be ignored. Further, we take the homogeneous-class pairs out of consideration, as a joint minimization with (2) should reduce the distances between them as argued in Section 2.2.

The inter-class loss $\mathcal{L}_{\text{inter}}$ is defined to take a summation over the heterogeneous-class pairs, which falls short of the inter-class margin threshold R_{inter}.

$$\mathcal{L}_{\text{inter}}(W) = \sum_{j,k:y_j \neq y_k} \max\{0, R_{\text{inter}} - \|f(p_j;W) - f(p_k;W)\|\} \tag{6}$$

R_{inter} is set to a q_{inter}-quantile among the inter-class distances. (6), thus, is focused on penalizing pairs which correspond to relatively smaller inter-class margins.

We minimize $\mathcal{L} = \mathcal{L}_{\text{intra}} + \beta\mathcal{L}_{\text{inter}}$ to increase the overall margins between classes. For anomaly detection, it is intuitive to employ a margin-based loss, as data near the boundaries of normal data or classes are more critical to the detection performance compared to those near the class center. We will refer to this framework as Prototype Data Description (PDD), as it models the enclosing hull of normal-class prototypes.

At testing, we compute the anomaly score using kernel density estimation (KDE) in the embedded space. Let $\hat{\mathcal{P}}_n = \{f(p_j) : y_j = n\}$ denote the set of embedded prototypes with class label n, and $D^{(n)}(z)$ the kernel density estimation of z given $\hat{\mathcal{P}}_n$. A large density indicates a closeness to the prototypes of n.

We define a scaled density function a_n such that

$$a_n(x) = \frac{D^{(n)}\left(f(x)\right) - D^{(n)}_{\max}}{D^{(n)}_{\max} - D^{(n)}_{\min}}$$

with scaling parameters

$$D^{(n)}_{\max} = \max_{z \in \bigcup_{\mathcal{Y}\setminus n} \hat{\mathcal{P}}_i} D_n(z),\ \ D^{(n)}_{\min} = \min_{z \in \bigcup_{\mathcal{Y}} \hat{\mathcal{P}}_i} D_n(z)$$

In [?,?], the test cases were scored for open-set recognition using the distance to the closest class-prototypes. In our study, we similarly use the inverse of the largest scaled density of the test case x as the anomaly score.

$$A(x) = \exp\left(-\max_n a_n(x)\right) \tag{7}$$

[1]

5 Empirical Results

5.1 Setup

This section presents an empirical study for comparative and graphical evaluation of PDD. We set up anomaly detection tasks using benchmark image datasets following previous studies, by excluding one class designated as anomalies from training. The training set is thus compiled from the remaining classes and the performance is measured for the detection of the anomaly class samples in the test set. The following experiments are conducted with three public datasets: MNIST [21], Fashion-MNIST [22], and CIFAR10 [23]. Their properties are summarized in Table 1.

[1] A sample code of the PDD is provided in `https://github.com/ProtoDD/pdd`

Table 1. Datasets

Dataset	#Image Size×Channels	#Instances	# Classes
MNIST	$28 \times 28 \times 1$	70,000	10
Fashion-MNIST	$28 \times 28 \times 1$	60000	10
CIFAR	$32 \times 32 \times 3$	60,000	10

As baselines for comparison, we conducted the same experiments using EGBAD [19], GANomaly [1], and MCDD [9]. EGBAD and GANomaly are baselines of unsupervised GAN-based DAD, given only the 'normal' class images without labels in training. MCDD is a baseline of data description and OOD, given the same dataset but with complete labels of 'normal' classes in training. PDD is given a 9-way, 20-shot labeled prototypes in addition to the same unlabeled dataset. The prototypes were chosen randomly and removed from the default training set. Note that PDD is at a disadvantage compared to the OOD model, while at an advantage compared to the unsupervised DAD with regards to the amount of supervising information used in training. For performance measures, we compute the Area Under the ROC curve (AUROC) and the Area Under the Precision-Recall Curve (AUPRC).

The baseline models were implemented based on their publicly available codes [2][3][4]. The hyperparameters of the baselines were determined by grid search around the suggested values from their respective papers. The summary of the main training parameters are shown in the appendix (Table 2).

The hyperparameters related to IB and KDE were empirically determined: IB trade-off $\beta = 3$, KDE bandwith $w = 10$, distance quantiles $q_{\text{intra}} = 0.5$, $q_{\text{inter}} = 0.25$. The GAN-architectures for pre-training are shown in the appendix (Tables 3 and 4). The training were conducted in a single-GPU environment with a Tesla P100 with 16GB memory. The optimizer was ADAM with learning rate 0.002.

5.2 Comparative Analysis

This section presents comparisons between performances of PDD and the baseline models. We report the AUROC measures due to the low AUPRCs of the baselines. The AUPRCs of the proposed model are reported in the next section.

Fig. 3 shows the comparisons with unsupervised DAD baselines on ten anomaly detection tasks based on MNIST datasets. The markers indicate the mean over ten repetitions, while the error bars indicate the standard deviation. Similarly, Figs. 4 and 5 show comparisons over the tasks based on Fashion-MNIST and CIFAR-10 datasets, respectively.

From Figs. 3-5, PDD has substantial advantage over GAN-based DADs in these thirty tasks. Overall, EGBAD may not be adequate for handling multi-

[2] `https://github.com/houssamzenati/Efficient-GAN-Anomaly-Detection`

[3] `https://github.com/samet-akcay/ganomaly`

[4] `https://github.com/donalee/DeepMCDD`

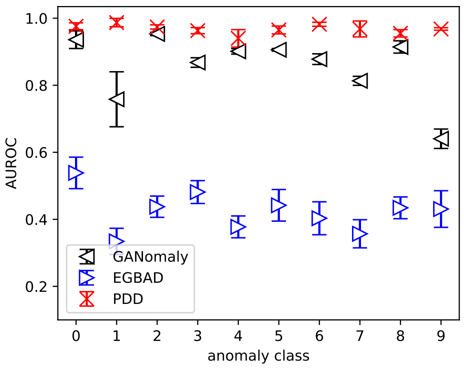


Fig. 3. vs Unsupervised-DAD (MNIST)

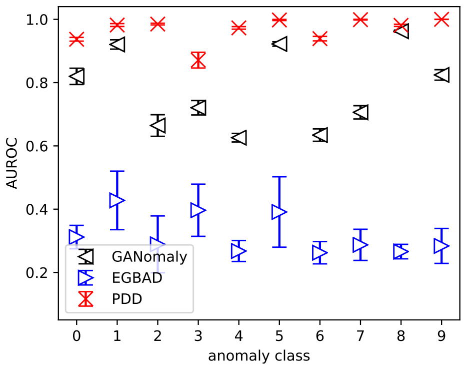


Fig. 4. vs Unsupervised DAD (Fashion)

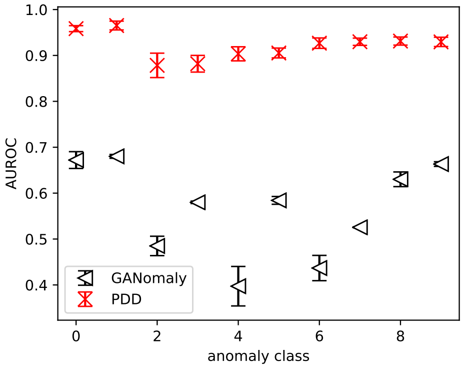


Fig. 5. vs Unsupervised DAD (CIFAR-10)

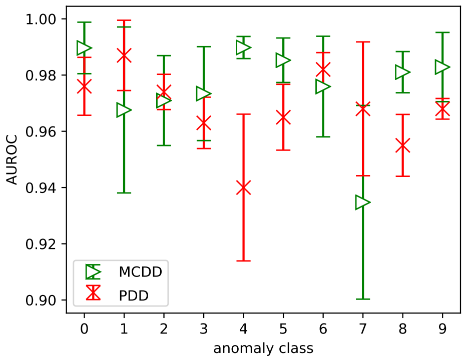


Fig. 6. vs OOD (MNIST)

class normal data. It is omitted from Fig. 5 due to its low measures. Additionally, PDD showed relatively small variances over the class designated as anomaly while GANomaly showed high variance depending on the class.

The comparison between PDD and the OOD baseline are shown separately in Figs. 6-8. The markers and the error bars respectively indicate the mean and the standard deviation over ten repetitions in each task.

Over the thirty tasks, PDD and MCDD averaged AUC higher than 0.85. MCDD generally showed larger variances than PDD, and neither outperformed the other overall. We note that PDD shows comparable performances while exploiting a limited amount of supervising information compared to MCDD.

The run time of PDD were 1.5 minutes per epoch on average. The run time of the baselines in proportion to that of PDD were as follows: PDD:1.0, EGBAD: 0.56, GANomaly: 0.84, MCDD: 1.4.

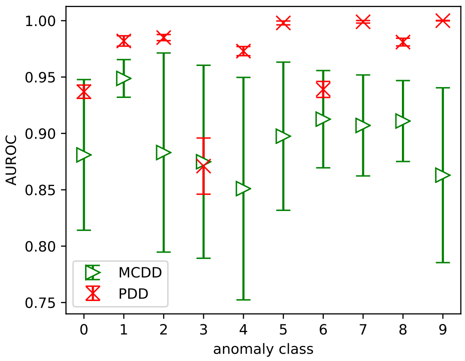


Fig. 7. vs OOD (Fashion)

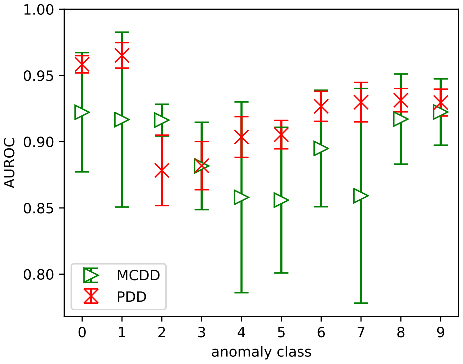


Fig. 8. vs OOD (CIFAR10)

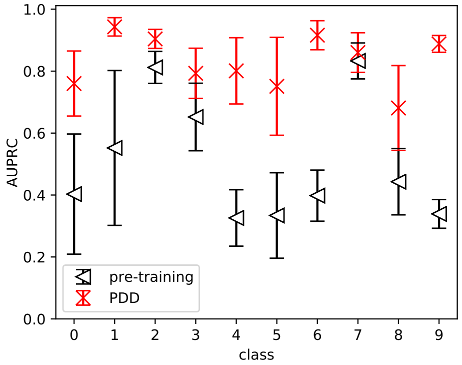


Fig. 9. Ablation Study: MNIST

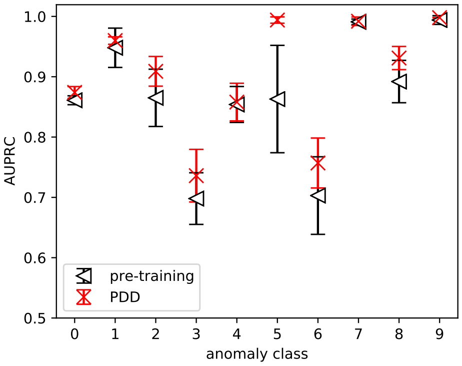


Fig. 10. Ablation Study: Fashion-MNIST

5.3 Ablation Study

In this and the following sections, we evaluate the impact of the IB-training by quantitative and graphical comparison of the proposed model after pre-training and after IB-loss training.

Figs. 9-11 show the comparisons of AUPRCs in the same thirty tasks as the previous experiment. The markers indicate the means over ten repetitions, while the error bars reflect the standard deviations.

The initial embedding by GANs produced comparative performances in several tasks, but in many tasks it yields substantially low AUPRC measures. Over the thirty tasks, IB-training achieves substantial improvements on top of the initially acquired embeddings.

5.4 Graphical Analysis

This section presents graphical analyses on low-dimensional mappings of the embedded images from typical cases. Figs. 12 and 13 show 2-D mapping of the

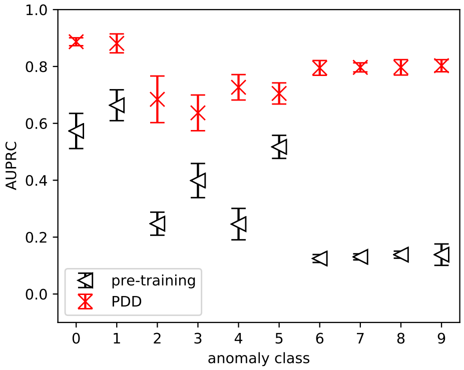
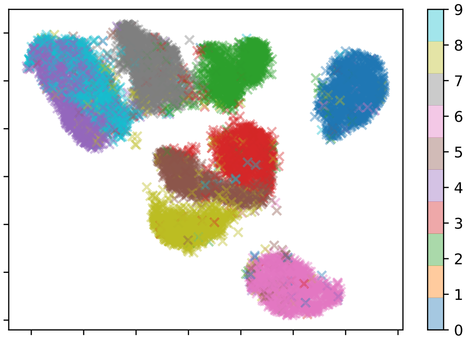


Fig. 11. Ablation Study: CIFAR-10



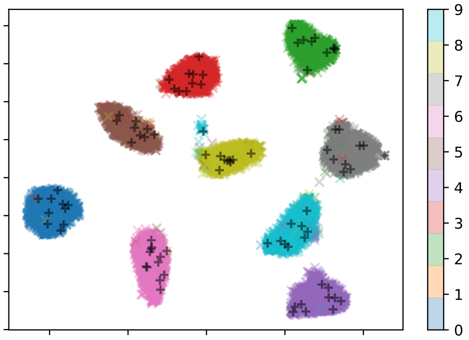


Fig. 12. 2D-map: pre-training (MNIST) **Fig. 13.** 2D-map: IB-training (MNIST)

MNIST test images after GAN pre-training and IB training, respectively. The 2-D mappings were generated by UMAP [10], in the setup where the anomaly class is digit ‘1’, The ’×’ markers of different colors indicate the embeddings of respective classes. The black ‘+’ markers indicate the prototypes.

Fig. 13 shows that the class-wise distributions from MNIST can be identified after pre-training, but they are mostly unseparated and few are overlapping at that point. Meanwhile, from Fig. 13, the class distributions are evenly separated and individually enclosed in different proximities after IB training.

The 2-D mappings of the Fashion-MNIST testset images after pre-training and IB training are shown in Figs. 14 and 15, respectively. The black and colored markers indicate the classes and the prototypes, and the anomaly class is digit ‘6’. Fig. 14 shows that most of the classes are overlapping with one or more other class distributions after pre-training. In comparison, the classes in Fig. 15 shows either some reduction in overlaps or increased separation. Note that groups of classes with inherently similar patterns, e.g., {5: Sandal, 7: Sneaker, 9: Boots} and {2: Pullover, 4: Coat}, are unseparated in 2-D, but their overlaps are reduced substantially after IB-training.

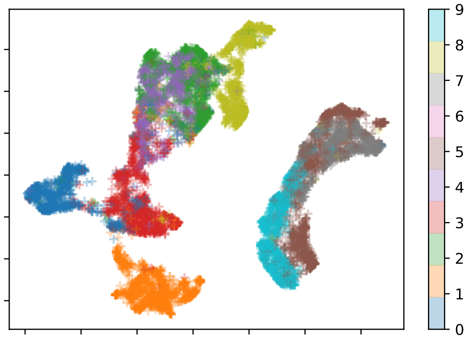


Fig. 14. 2D-map: pre-training (Fashion)

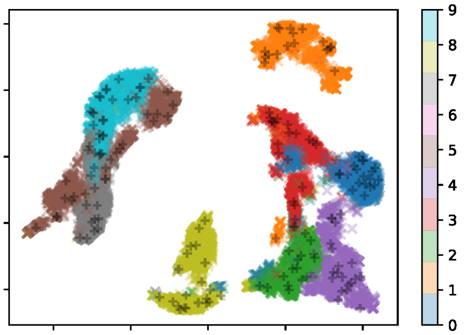


Fig. 15. 2D-map: IB-training (Fashion)

From many cases similar to those above, we can expect PDD to expand the inter-class margins while gathering each class to a different proximity. The graphical analysis also indicates that the KDE-based anomaly scores can be effective for detecting anomalies that appear between separated normal classes.

Generally, not all normal classes can be separated. As seen in the Fashion-MNIST experiment, when classes are intrinsically similar, they are embedded to adjoint regions. We note that it is not the goal to cluster all independent classes, and the capacity for anomaly detection will not necessarily be impaired when similar classes are unseparated. Practically, if when classes are similar, e.g., shoes and boots, there may be instances which are hard to discern, but they are unlikely to be considered anomalies.

6 Conclusion

In this paper, we addressed the task of anomaly detection in the presence of multi-class, normal data, with a new, practically feasible input setting, utilizing a small set of class prototypes. We implemented a deep neural network training driven by an information-theoretic principle, with a loss based on intra-class and inter-class distances among the prototypes. Our empirical evaluation showed that the proposed method holds substantial advantages over unsupervised DAD models and also is comparable to an OOD data description model in terms of the performance measures. From the graphical analyses, the proposed model can typically learn mutually distant and dispersed embedding of the class prototypes, which enables its density-based anomaly scores.

Given that proposed model addresses a setting slightly different from those of existing tasks, e.g., ODD and Unsupervised Anomaly Detection, building a benchmark which is also practically relevant is an important future work.

Futhermore, we aim to understand its more practical characteristics such its sensitivity to noise and the number of prototypes, the means and its impact of selecting “good” prototypes.

References

1. Akcay, S., Atapour-Abarghouei, A., Breckon, T.P.: Ganomaly: Semi-supervised anomaly detection via adversarial training. In Computer Vision – ACCV 2018. pp. 622–637. Springer International Publishing (2019)
2. Alemi, A.A., Fischer, I., Dillon, J.V., Murphy, K.: Deep variational information bottleneck. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings. OpenReview.net (2017),
3. Ando, S.: Deep Representation Learning with an Information-theoretic Loss. CoRR **abs/2111.12950** (2021),
4. Ding, R., Guo, G., Yang, X., Chen, B., Liu, Z., He, X.: Bigan: Collaborative filtering with bidirectional generative adversarial networks. In: Proceedings of the 2020 SIAM International Conference on Data Mining, SDM 2020, pp. 82–90. SIAM (2020). ,
5. Ghafoori, Z., Leckie, C.: Deep multi-sphere support vector data description. In: Proceedings of the 2020 SIAM International Conference on Data Mining, SDM 2020, pp. 109–117. SIAM (2020). ,
6. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2. pp. 2672–2680. NIPS'14, MIT Press, Cambridge, MA, USA (2014),
7. Jeong, T., Kim, H.: Ood-maml: Meta-learning for few-shot out-of-distribution detection and classification. In: Advances in Neural Information Processing Systems. vol. 33, pp. 3907–3916. Curran Associates, Inc. (2020),
8. Kwon, D., Kim, H., Kim, J., Suh, S.C., Kim, I., Kim, K.J.: A survey of deep learning-based network anomaly detection. Cluster Computing (Sep 2017). ,
9. Lee, D., Yu, S., Yu, H.: Multi-class data description for out-of-distribution detection. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 1362–1370. KDD '20, Association for Computing Machinery, New York, NY, USA (2020). ,
10. McInnes, L., Healy, J., Saul, N., Großberger, L.: UMAP: uniform manifold approximation and projection. J. Open Source Softw. **3**(29), 861 (2018). ,
11. Pang, G., Shen, C., Cao, L., Hengel, A.V.D.: Deep learning for anomaly detection: A review. ACM Comput. Surv. **54**(2) (Mar 2021). ,
12. Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S.A., Binder, A., Müller, E., Kloft, M.: Deep one-class classification. In: Proceedings of the 35th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 80, pp. 4393–4402. PMLR,
13. Ruff, L., Vandermeulen, R.A., Görnitz, N., Binder, A., Müller, E., Müller, K., Kloft, M.: Deep semi-supervised anomaly detection. In: 8th International Conference on Learning Representations, ICLR 2020. OpenReview.net (2020),
14. Schlegl, T., Seeböck, P., Waldstein, S.M., Schmidt-Erfurth, U., Langs, G.: Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery. In: Information Processing in Medical Imaging - 25th International Conference, IPMI 2017, Boone, NC, USA, June 25-30, 2017, Proceedings. Lecture Notes in Computer Science, vol. 10265, pp. 146–157. Springer (2017). ,
15. Tax, D.M.J., Duin, R.P.W.: Support vector data description. Mach. Learn. **54**, 45–66 (January 2004). ,
16. Tishby, N., Zaslavsky, N.: Deep learning and the information bottleneck principle. In: 2015 IEEE Information Theory Workshop (ITW). pp. 1–5 (2015).

17. Tishby, N., Pereira, F.C., Bialek, W.: The information bottleneck method. Computing Research Repository(CoRR) **physics/0004057** (2000)
18. Zenati, H., Romain, M., Foo, C., Lecouat, B., Chandrasekhar, V.: Adversarially learned anomaly detection. In: 2018 IEEE International Conference on Data Mining (ICDM). pp. 727–736 (2018)
19. Zenati, H., Foo, C.S., Lecouat, B., Manek, G., Chandrasekhar, V.R.: Efficient gan-based anomaly detection. CoRR **abs/1802.06222** (2018),
20. Zenati, H., Foo, C.S., Lecouat, B., Manek, G., Chandrasekhar, V.R.: Efficient gan-based anomaly detection (2019)
21. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (Nov 1998).
22. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms (Aug 2017)
23. Krizhevsky, A.: Learning Multiple Layers of Features from Tiny Images. Master's thesis (2009),

Appendix

Table 2. Summary of training parameters

	MNIST		Fashion-MNIST		CIFAR10	
	BatchSize	#Epochs	BatchSize	#Epochs	BatchSize	#Epochs
EGBAD	100	20	100	30	150	30
GANomaly	300	15	300	15	300	40
MCDD	150	40	150	40	200	80
PDD (pre-training)	200	20	100	40	200	30
PDD (IB training)	200	40	100	40	200	30

Table 3. GAN architecture (MNIST/Fashion-MNIST)

(a) Generator

Layer unit	D_{out}
Input	100
Lin + BN + ReLU	$7 \times 7 \times 512$
CnvTr+BN+LkReLU	$14 \times 14 \times 256$
CnvTr+BN+LkReLU	$14 \times 14 \times 128$
CnvTr+BN+LkReLU	$28 \times 28 \times 64$
CnvTr+Tanh	$28 \times 28 \times 1$

(b) Discriminator

Layer unit	D_{out}
Input	$28 \times 28 \times 1$
Cnv+BN+LkReLU	$28 \times 28 \times 8$
Cnv+BN+LkReLU	$14 \times 14 \times 16$
Cnv+BN+LkReLU	$7 \times 7 \times 32$
Cnv+BN+LkReLU	$7 \times 7 \times 64$
Fltn+Lin+Sgmd	1

Table 4. GAN architecture (CIFAR-10)

(a) Generator

Layer unit	D_{out}
Input	100
ConvTr+BN+ReLU	$4 \times 4 \times 512$
CnvTr+BN+ReLU	$8 \times 8 \times 256$
CnvTr+BN+ReLU	$16 \times 16 \times 128$
CnvTr+BN+ReLU	$32 \times 32 \times 64$
CnvTr+Tanh	$64 \times 64 \times 3$

(b) Discriminator

Layer unit	D_{out}
Input	$64 \times 64 \times 3$
Cnv+LkReLU	$32 \times 32 \times 64$
Cnv+BN+LkReLU	$16 \times 16 \times 128$
Cnv+BN+LkReLU	$8 \times 8 \times 256$
Cnv+BN+LkReLU	$4 \times 4 \times 512$
Conv+Sgmd	1