Anomaly Detection with Domain Adaptation

Ziyi Yang[1] Iman Soltani[2] Eric Darve[1]
[1]Stanford University
[2]University of California, Davis
{*ziyi.yang, darve*}*@stanford.edu, isoltani@ucdavis.edu*

Abstract

Despite great advances have been made in the field of domain adaptation (DA), the vast majority of current methods in DA solve classical ML tasks, e.g. classification. In this paper, we study a novel research direction: semi-supervised anomaly detection with domain adaptation. Given a set of normal data from a source domain and a ***limited*** *number of normal examples from a target domain, the goal is to have a well-performing anomaly detector in the target domain. We then present the Invariant Representation Anomaly Detection (IRAD) to solve this problem where we first learn to extract a domain-invariant representation. The extraction is achieved by an across-domain encoder trained together with source-specific encoders and generators by adversarial learning. An anomaly detector is then trained using the learnt representations. We evaluate IRAD extensively on anomaly detection datasets, object recognition datasets and digits benchmarks. Experimental results show that IRAD outperforms baseline models by a wide margin across different datasets. We derive a theoretical lower bound for the joint error that explains the performance decay from over-training and also an upper bound for the generalization error.*

1. Introduction

Also known as novelty detection or outlier detection, anomaly detection (AD) is the process of identifying abnormal items or observations that differ from what is defined as normal. Anomaly detection has been applied in many areas, including cyber security (detection of malicious intrusions), medical diagnosis (identification of pathological patterns), robotics (recognize abnormal objects), etc. Anomaly detection with different settings have been studied, for example, many anomaly detection works aim to solve the semi-supervised learning problem such that only normal data are available for training [3, 24, 32]. The anomaly detection models are expected to learn an anomaly score function $A(\cdot)$ such that during testing anomalous data should be assigned higher anomalous scores than the examples labelled as "normal."

In practical applications, the normal data distribution can have a shift. For example, in manufacturing, we have sufficient amount of "normal" observations of engine type A (source domain) and we want to design an anomaly detection algorithm for a different but similar engine type B (target domain). However, we may have only **limited** normal observations for the target domain. One option is to re-collect a large-scale normal dataset in the new domain, but this is often prohibitively costly and time-consuming for many practical applications, e.g., medical healthcare and autonomous driving [9, 34]. Can we design a system that can leverage data from the both domains to learn an efficient anomaly detection model for the target domain? In this paper we attempt to address this important and interesting question. This type of problem is also known as Domain Adaptation (DA), which studies the transfer learning between the source and target domains [5, 15, 25, 34].

Surprisingly, domain-adapted anomaly detection has not drawn as much interest as its classification peer, especially comprehensive studies on semi-supervised anomaly detection in the domain adaptation setting are rare. As an effort to solve the problem, we propose the Invariant Representation Anomaly Detection (IRAD) model. IRAD leverages a shared encoder to extract common features from source and target domain data. The shared encoder is adversarially trained with a source-domain specific encoder and a generator. Such design is required to avoid overfitting the target domain where training data are very limited. Then a simple and off-the-shelf anomaly detection model, Isolation Forest (IF), is trained on the extracted shared representations of source and target domain. At test time, the trained IF assigns the anomalous scores given the extracted features from the test target data.

We evaluate IRAD thoroughly on cross-domain anomaly detection benchmarks. Evaluation datasets are include anomaly detection datasets (MvTec AD), standard digit datasets (MNIST, USPS and SVHN) and Office-Home domain adaptation datasets. We compare IRAD to base-

lines including the prevailing anomaly detection models and competitive domain adaptation algorithms. Evaluation results show that IRAD outperforms the baseline models by significant margins. For example, on Office-Home dataset (Product→Clip Art), IRAD improves upon the best baseline by almost 10%. In addition, we derive a lower bound on the joint error on both domains for models based on invariant representations, which explains the observation that the accuracy on the target domain is inherently limited by the "distance" between the source and target domains. We also obtain a generalization upper bound that reveals the sources of generalization error. We conduct ablation studies to confirm the effectiveness of objective functions in IRAD. The content of this paper is also included in the first author's PhD thesis [31].

2. Related Work

One major class of anomaly detection algorithms is generative models that learn the normal data distribution via generation processes, e.g., autoencoders (AE) and generative adversarial networks (GANs). For example, GANs [13] are trained on images of healthy retina images to identify disease markers [26]. Regularized Cycle-Consistent GAN [32] introduces a regularization distribution to correctly bias the generation towards normal data. Memory augmented generative models [12, 33] maintain external memory units that interact with the encoding process to store latent representations of the normal data. An emerging type of anomaly detection methods is self-supervised models [3, 11]. They first apply different transformations to the normal data and train a classifier to predict the corresponding transformation that anomaly scores depend on.

Domain adaptation is to learn from source domain data together with limited information of target domain in order to have a well-performing model on the target domain. One heavily studied direction is the unsupervised image classification. Given labeled source-domain images and unlabeled target-domain images, the goal is to obtain a target-domain classifier. One type of methods learns a transformation from the source to target domain [15, 18]; some approaches learn invariant representations between the two domains [5, 28, 34]. There are also works addressing few-shot domain adaptation with various problem settings. Few-shot domain translation [2, 7] learns a mapping function from source to the target domain where limited target-domain data are given. Since the data setting is similar to IRAD, we include the one of state-of-the-art models BiOST [7] as a baseline.

Previous works studying the task of cross-domain anomaly detection typically have different problem setups from this paper. For example, most works assume access to labeled (both normal and abnormal) data at least in the source domain: One-class Transfer Learning [6] learns a regressor using labeled source data, which can predict the target-domain anomaly distribution from the normal distribution (estimated from target-domain normal data). TAD [17] learns the conditional data distribution with fully-supervised data in multiple target domains. A few works use only normal data in the source and target domain. Collective-AD studies the collective anomaly detection problem, where the target domain is one of the source domains, with a mixture of Gaussian graphical models [16]. AdaFlow investigates the multi-source transfer anomaly detection problem by learning normalizing flows from target domains to the source domain [30]. However, these works assume sufficient normal data in the target domain are available, which can be a demanding condition to meet as mentioned before.

3. Methodology

Problem Statement We investigate the problem of semi-supervised anomaly detection in the domain adaptation setting. For training, the learning algorithm has access to n data points $\{(\boldsymbol{x}_{src}^{(i)}, y_{src}^{(i)})\}_{i=1}^{n} \in (X \times Y)^n$ sampled i.i.d. from the source domain $\mathcal{D}_S$ and **limited** target data points $\{(\boldsymbol{x}_{tgt}^{(j)}, y_{tgt}^{(j)})\}_{j=1}^{n_t} \in (X \times Y)^{n_t}$ sampled i.i.d. from the target domain $\mathcal{D}_t$ (where n_t is small and $n_t \ll n$). Let $y = 0$ ($y = 1$) denote normal (abnormal). In semi-supervised anomaly detection, we only have access to normal data, i.e., $y_{src}^{(i)} = 0$ and $y_{tgt}^{(j)} = 0, \forall i, j$. The goal is to build an anomaly score function $A(\boldsymbol{x}_{tgt}) : X \to a \in \mathbb{R}$ in the target domain. The test set consists of both normal and abnormal target domain data. An evaluation metric of learnt models is the area under the Receiver Operating Characteristic (ROC) curve, or AUROC, w.r.t. the true labels and anomaly scores of test examples.

3.1. Invariant Representations Extraction by Adversarial Learning

Learning the domain-invariant features is a prevailing solution for the domain adaptation problem [5, 10, 34]. IRAD includes a shared encoder E_{sh} to extract common features between the source and target data, a method commonly used in previous works [5, 10]. To enable an appropriate split of shared and domain-specific components, IRAD also trains a private encoder E_{pv} in the source domain to remove the source-specific information from the domain shared encodings (see Sec. 3.2). To ensure that the learned components actually contain useful information, we also introduce a generator to map from the latent space to data space in the source domain G_{src}. The generator G_{src}, encoders E_{sh} and E_{pv} are adversarially trained together using a discriminator D_{src} in the source domain. The adversarial loss is given as

follows:

$$\min_{\{E_{sh},E_{pv},G_{src}\}} \max_{D_{src}} V_{src}(D_{src}, G_{src}, E_{pv}, E_{sh}) = \mathbb{E}_{\boldsymbol{x}_{src}}[\log D_{src}(\boldsymbol{x}_{src})] + \mathbb{E}_{\boldsymbol{x}_{src}}[\log(1 - D_{src}(\boldsymbol{x}'_{src})] + \mathbb{E}_{\boldsymbol{x}_{src},\boldsymbol{x}_{tgt}}[\log(1 - D_{src}(\boldsymbol{x}'_{tgt})] + \mathbb{E}_{\boldsymbol{x}_{src}}[\log(1 - D_{src}(\boldsymbol{x}_{rnd})] \quad (1)$$

where $\boldsymbol{x}'_{src} = G(E_{pv}(\boldsymbol{x}_{src}) + E_{sh}(\boldsymbol{x}_{src}))$ represents the reconstruction of the source data; $\boldsymbol{x}'_{tgt} = G(E_{pv}(\boldsymbol{x}_{src}) + E_{sh}(\boldsymbol{x}_{tgt}))$ denotes the generation using the extracted common information $E_{sh}(\boldsymbol{x}_{tgt})$ from the target data and private encodings from the source data; $\boldsymbol{x}_{rnd} = G(\boldsymbol{z} + E_{sh}(\boldsymbol{x}_{src}))$ is generated using a variable $\boldsymbol{z}$ sampled from a random distribution (empirically we find $\mathcal{N}(0, 1)$ works well) together with shared encodings $E_{sh}(\boldsymbol{x}_{src})$. The $\boldsymbol{x}_{rnd}$ term is designed to avoid the scenario in which the private encoder is (incorrectly) so powerful that all latent information for the source domain is encoded with E_{pv}. By taking a random vector as part of the input, the shared encoder is trained adversarially to capture the essential information of the source data such that the generated $\boldsymbol{x}_{rnd}$ is close to $\boldsymbol{x}_{src}$. We conduct an ablation study about $\boldsymbol{x}_{rnd}$ in Sec. 5. The discriminator D_{src} is trained to distinguish real source data $\boldsymbol{x}_{src}$ from $\boldsymbol{x}'_{src}$, $\boldsymbol{x}'_{tgt}$, and $\boldsymbol{x}_{rnd}$. The shared encoder E_{sh}, E_{pv}, and G_{src} are trained to maximize the error D_{src} makes. At optimality, $\boldsymbol{x}'_{src}$, $\boldsymbol{x}'_{tgt}$, and $\boldsymbol{x}_{rnd}$ should resemble real data $\boldsymbol{x}_{src}$ w.r.t. D_{src}.

Besides adversarial training, we also optimize with the following cycle consistent losses:

$$l_1 = \|\boldsymbol{x}_{src} - \boldsymbol{x}'_{src}\|_2, \qquad l_2 = \|\boldsymbol{x}_{src} - \boldsymbol{x}'_{tgt}\|_2 \quad (2)$$

The first loss enforces the cycle consistency property in the source data space. The second one ensures that components extracted from the target data $E_{sh}(\boldsymbol{x}_{tgt})$ are actually shared features such that they reside in the same subspace as $E_{sh}(\boldsymbol{x}_{src})$. The cycle consistency losses are crucial in our experiments with high-dimensional real-world images (e.g., in Office-Home dataset where image sizes are usually larger than 300×300). We speculate this is due to the instability in GAN training for high-dimensional data [1]. Stronger signals like direct cycle consistency losses should help the optimizations of generators and encoders.

3.2. Split of Private and Shared Components

The subspace of shared and private encodings of the source data should be dissimilar since they extract different features of $\boldsymbol{x}_{src}$. For instance, in a domain adaptation problem on MNIST (source) and SVHN (target), denoted as MNIST→SVHN, the shared encodings should learn to extract information relevant to the digit, while the private encodings are expected to contain components about digits style, size, etc. To enforce this characteristic, we introduce an optimization objective to minimize the similarity between the (normalized) shared encodings and private encodings, similar to [5]:

$$l_{dis} = \|E_{sh}(\boldsymbol{x}_{src})^T E_{pv}(\boldsymbol{x}_{src})\| \quad (3)$$

Also, the shared encodings extracted from two domains are expected to be similar, since they should capture the common information between the two domains. Therefore, we minimize the negative of inner product between the (normalized) shared encodings of the source and target data:

$$l_{sim} = -\|E_{sh}(\boldsymbol{x}_{src})^T E_{sh}(\boldsymbol{x}_{tgt})\| \quad (4)$$

We show in Fig. 8 that l_{sim} objective is essential to ensure proximity between the shared encodings extracted from the source and target data. Without l_{sim}, we observe that the shared encodings of the source and target data are too far apart which undermines the performance of the anomaly detection algorithm. More details on this ablation study will be given in Sec. 5.

The final objective function of IRAD is the weighted sum of the losses mentioned above:

$$V_{src} + \alpha_1 l_1 + \alpha_2 l_2 + \beta(l_{dis} + l_{sim}) \quad (5)$$

Empirically, we find $\alpha_1 = 1$, $\alpha_2 = 1$, $\beta = 0.5$ works well. Unless otherwise stated, these values are used in the experiments.

3.3. Anomaly Detection

After the shared encoder is trained, we can conveniently leverage an off-the-shelf anomaly detection algorithm A' to train an anomaly detection model using the shared representations extracted from both source and target data in the training set. In general, any semi-supervised anomaly detection models can be used here. In this paper, we explore the options of using Isolation Forest (IF) [20] and One-Class SVM (OCSVM) [27] as A' in IRAD, denoted as IRAD (IF) and IRAD (OC) respectively in later sections. The description of IF and OCSVM can be found in the next section. We choose IF and OCSVM because they are prevailing and effective methods with the standard implementation available [22]. We conduct detailed comparisons between IRAD(IF)/IRAD(OC) and vanilla IF/OCSVM in the experiments.

For testing, given a test example $\boldsymbol{x}$, we encode $\boldsymbol{x}$ to the shared subspace between source and target space $E_{sh}(\boldsymbol{x})$. The anomaly score $A(\boldsymbol{x})$ is then given as $A'(E_{sh}(\boldsymbol{x}))$. Fig. 1 illustrates an overview of IRAD framework where the source and target domain is Carpet and Leather respectively from MVTec AD dataset.

4. Experimental and Theoretical Results

We evaluate IRAD extensively on various kinds of benchmarks. The datasets include a recent comprehen-

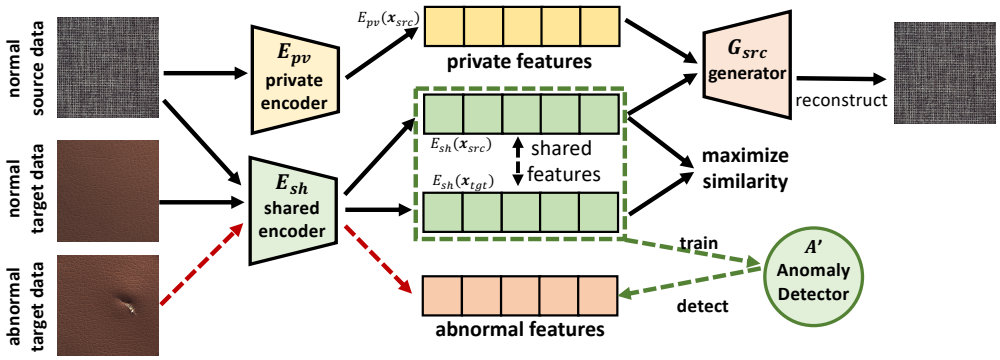

Figure 1. The overview of Invariant Representation Anomaly Detection (IRAD). IRAD first learns to extract shared features of normal source (category Carpet from MVTec AD) and target (Leather) data by adversarial learning and dividing the latent space of source data. An anomaly detector is then trained with the extracted features (dashed green box) of the source and target domain.

sive anomaly detection benchmark MVTec AD [4]; second, MNIST digit images (source domain), SVHN and USPS (target domain); third, Office-Home object recognition Dataset [29]. Finally, we also derive the theoretical bounds for joint error and generalization error of IRAD, which are consistent with experimental observations.

4.1. MVTec AD Dataset

MVTec Anomaly Detection (MVTec AD) [4] is a real-world, comprehensive and multi-object dataset. We build an AD domain adaptation benchmark by leveraging texture pattern objects in MVTec AD, including Carpet (C), Leather (L), and Wood (W). For example, assume the source and target domain is Carpet and Leather respectively, denoted as $C \rightarrow L$. In the training phase, images of Carpet and a limited number of Leather images are available (images from both domains are normal). Test data include images of normal and abnormal Leather images. We use the original train/text split in the dataset. We compare IRAD with the following baselines:

Isolation Forest (IF) is a tree ensemble method that "isolates" data by randomly selecting a feature and then randomly selecting a split value between the maximum and minimum values of the selected feature to construct trees [20]. The averaged path length from the root node to the example is a measure of normality. We experiment with two types of isolation forest: IF (T) trained with only target data; IF (S+T) trained with both source and target data.

One Class Support Vector Machines (OCSVM) is a classical anomaly detection algorithm similar to the regular SVM. OCSVM is a kernel-based method that learns a decision function for novelty detection [27]. It classifies new data as similar or different to the normal data. Similar to IF, we test with two variants of OCSVM: OCSVM (T) and OCSVM (S+T).

Bidirectional One-Shot Unsupervised Domain Mapping (BiOST) is a recent work on few-shot domain transformation [7]. BiOST learns a encoder-generator pair for each domain respectively. Networks are then trained with across domains cyclic mapping losses and a KL divergence in the latent space similar to the one in Variational Autoencoder (VAE). The anomaly score of a target data example is its reconstruction error. BiOST is a representative baseline of methods that leverage cross-domain transformation [15,30].

Deep Support Vector Data Description (DSVDD) is a competitive deep learning one-class classification model for anomaly detection [24]. DSVDD projects data to a sphere in the latent space by learning the feature encoder and the data center of the sphere. We train DSVDD on the union of source and target domain data.

Data Augmentation (AGT): We also test an intuitive approach by augmenting the target domain training data, denoted as "AGT." The data are augmented by image rotations and flipping. An IF is then trained on the augmented data.

The training and implementations details are as follows. The encoders E_{sh} and E_{pv} in IRAD are ResNet-50 [14] pretrained on ImageNet where the last layer is removed, and a fully connected layer is added. The decoder is a ten-layer transpose convolution neural networks. The discriminator D_{src} is a ResNet-18 network (without pretraining) followed by a final layer for classification. To improve the optimization process, we use the adversarial objective as in least-square GANs [21]. To have a fair comparison, baseline models with encoding networks, e.g., DSVDD and BiOST, also leverage the pretrained ResNet-50 as the encoders. The size of latent representation output by encoder is 128 chosen by cross-validation. More details on implementations and training are available in the appendix. The number of target training data $n_t = 10$.

Experimental results averaged on 10 runs with different random seeds are presented in Fig. 2. Recall the evaluation metric is AUROC w.r.t. the true labels and anomaly scores of test examples. For a clearer visualization, best

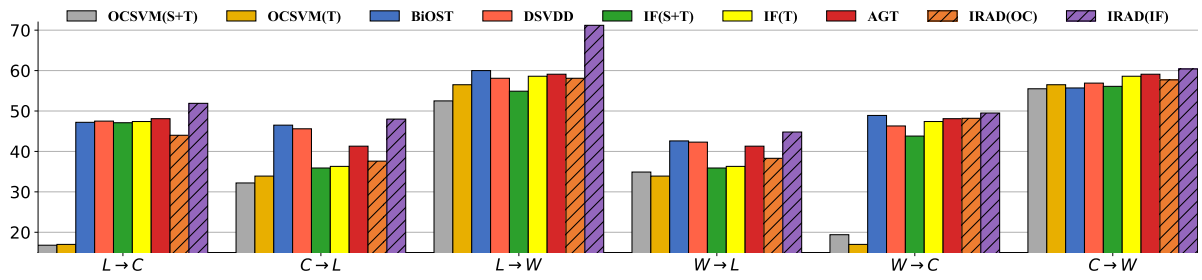

Figure 2. AUROC% of IRAD and baselines on MVTec AD dataset in six different adaptation settings (x-axis). Our IRAD models are highlighted in red text.

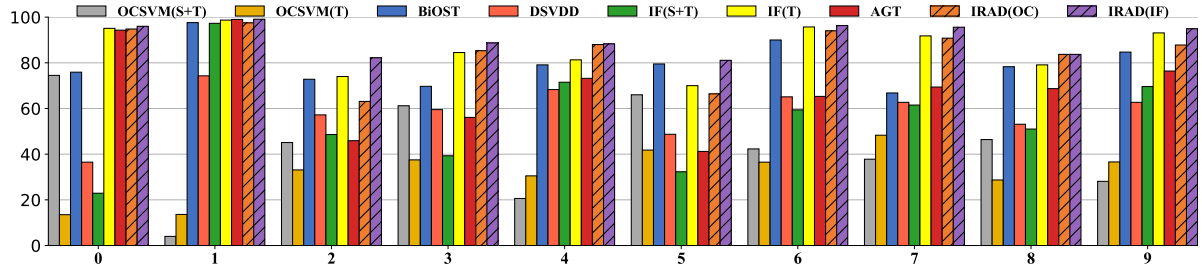

Figure 3. Experiments results (AUROC%) on digits datasets with MNIST as the source and USPS as the target domain. Our IRAD models are highlighted in red text.

three models for each class are presented here. Full results of all baseline models, including standard deviations, are available in Table 1 in the appendix. IRAD (IF) outperforms other baselines in all six adaptation settings. Also note that IRAD (OC) outperforms the vanilla OCSVM, which further validates the effectiveness of the extracted shared features.

4.2. Digits Anomaly Detection

We then evaluate on digits datasets with two adaptation scenarios: adaptation from MNIST (source) to USPS (target) and MNIST (source) to SVHN (target). An example of the evaluation setup is as follows. Following previous works on anomaly detection [11, 24, 32], assume digit 0 is the normal class. In the training phase, digit 0 from source domain (e.g., MNIST) as well as a limited number of digits 0 from the target domain (e.g., USPS) are available ($n_t = 50$). We explore with different values of n_t in IRAD in the discussion section. Test data contain all the categories of digits in the target domain where digits 0 are labelled as "normal" and other digits are labelled as "abnormal." We use the original train/test split in the target dataset.

Images are preprocessed into gray scale single-channel images of size 32×32 so that they can be input to the same network. The shared encoder, private encoder and discriminator in IRAD follow the configurations in standard DCGANs [23]. To ensure a fair comparison, we use the same neural network architectures in IRAD, BiOST and DSVDD (the feature extractor). Hyperparameters are chosen by cross-validation, e.g., the size of latent representation output by encoder is 64.

Fig. 3 and Fig. 4 shows result of MNIST→USPS and MNIST→SVHN respectively (averaged over 10 runs) by regarding each class of digits as normal. Full results containing all evaluated models are available in Table 2 and Table 3 in the appendix. For AGT methods, image transformations that can change the digits (e.g. 9 and 6) are avoided. IRAD outperforms all baseline models in both domain adaptation settings. An interesting observation is that IF (S+T) actually performs worse than IF (T). We speculate that this is because MNIST and USPS digits are from close but still distinct distributions. MNIST data actually add noises to the training of IF and undermine the performance. We provide a theoretical explanation for this observation in Section 4.4.

4.3. Objects Recognition Anomaly Detection

The Office-Home objects recognition dataset [29] is a prevailing and challenging domain adaptation benchmark. The images are high-dimensional where the average side length is more than 300. We test with ten categories that have reasonably sufficient data for evaluation in Clip Art and Product domains domains, as listed in the x-axis in Fig. 5. Object examples are shown in the appendix. We test on two experimental scenarios: Product→Clip Art and Clip Art →Product. Since the number of images in a domain is limited, we augment the training data in the source

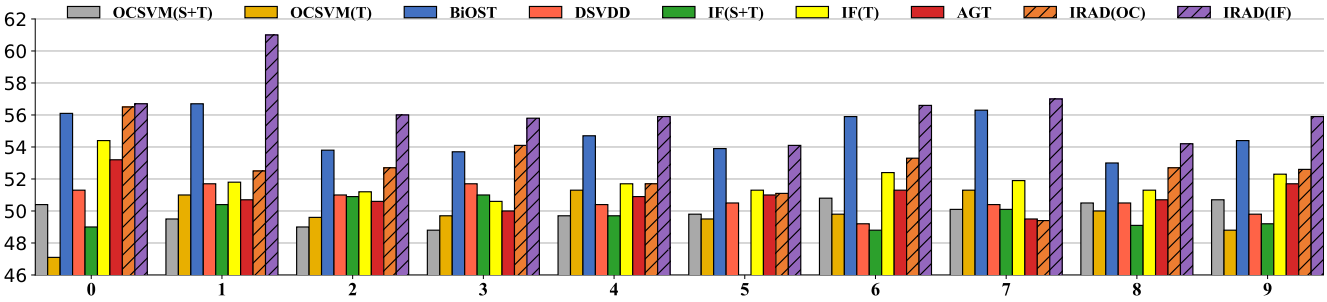

Figure 4. Experiments results (AUROC%) on digits datasets with MNIST as the source and SVHN as the target domain. Our IRAD models are highlighted in red text.

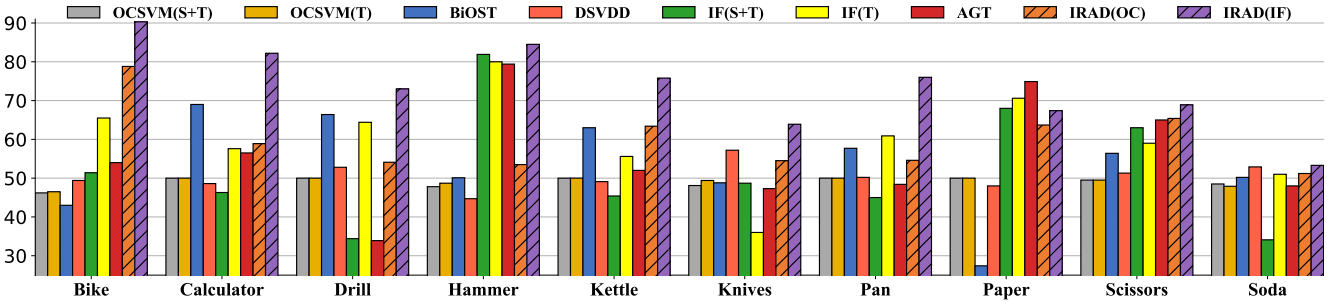

Figure 5. Results (AUROC%) on Office-Home dataset with Clip Art as the source and Product as the target domain. Our IRAD models are highlighted in red text.

domain by rotations and flipping, which increases the size of training source data by eight times. For a fair comparison, baseline models are also trained with the augmented datasets. The number of target domain images in the training set $n_t = 10$.

Experiment results are shown in Fig. 5 and Fig. 6. Full results averaged on 10 runs of all baselines are available in the appendix. In each bars cluster, the corresponding objects category on the x-axis is regarded as the normal class. IRAD shows strong performance in both adaptation scenarios and outperforms all baseline models in 18 out of 20 experiments. We will show later that cycle-consistency losses are crucial in the high-dimensional Office-Home dataset. We speculate that due to the increased complexity in images and the generation process, the transformation-based BiOST is not as good as in digits benchmarks.

4.4. Bounds for the Joint Error and the Generalization Error

Recent theoretical works on classification domain adaptation discover that minimizing the empirical error on the source domain can be detrimental for the model's performance in the target domain [34]. We observe the same phenomenon in domain-adaptation AD that overtraining IRAD leads to less accurate detection, as shown in Fig. 7. The adaptation performance first grows and gradually decreases after 5 epochs. We derive an information-theoretic lower bound of the joint error (Thm. 1) to explain this phenomenon.

We start with definitions and notations. Let $\mathcal{D}^{Y_S}$ and $\mathcal{D}^{Y_T}$ denote the marginal label distribution in the source and target domains. The projection from the data space X to the latent invariant representation space Z, induced by E_{sh} in the case of IRAD, is denoted as g. The hypothesis (labeling) function h is shared between two domains that map invariant representations Z to predictions $\hat{Y}$. For IRAD, the hypothesis h is induced by the IF learned on the invariant representations (IF learns the anomaly function). To ease the proof process, we assume the anomaly scores are transferred to classification probabilities, for example by applying a threshold.

The above process can be denoted as the Markov chain $X \xrightarrow{g} Z \xrightarrow{h} \hat{Y}$ [10, 34]. Let d_{JS} denote the JS distance which is the square root of JS divergence [8]. Let $\varepsilon_S(h \circ g)$ and $\varepsilon_T(h \circ g)$ denote the error of the learned model in the source and target domain respectively. Then we have the following theorem on the lower bound for joint error (the proofs of theorems are provided in the appendix):

Theorem 1. *Assume the chain is Markov, a lower bound for the joint error on the source and target domains is:*

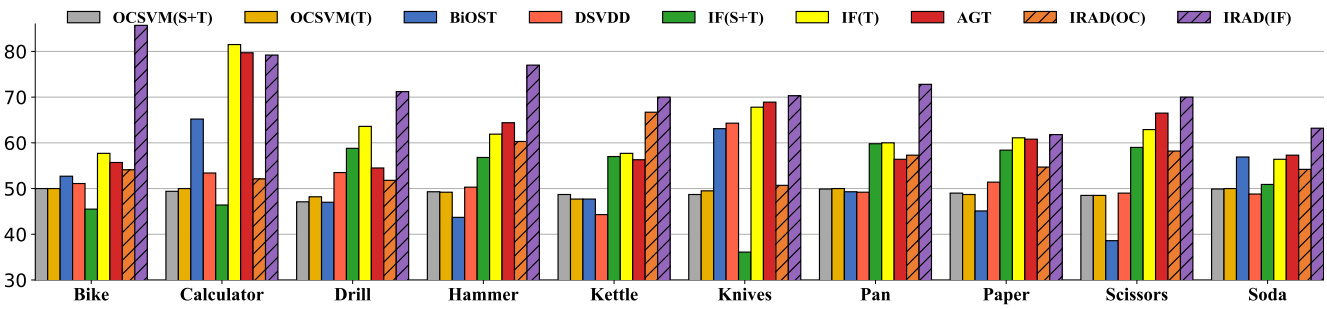

Figure 6. Results (AUROC%) on Office-Home dataset with Product as the source and Clipart as the target domain. Our IRAD models are highlighted in red text.

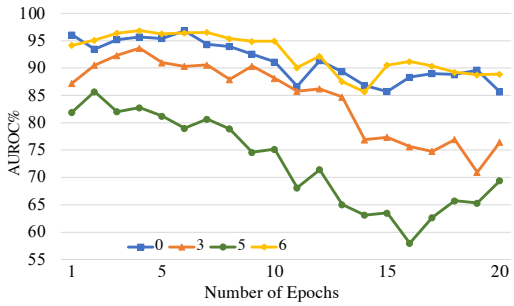

Figure 7. Overtraining to minimize the source domain error hurts the performance on the target domain (experiments conducted on MNIST→USPS).

$$\varepsilon_S\left(h \circ g\right) + \varepsilon_T\left(h \circ g\right) \geq \frac{1}{2} d_{\mathrm{JS}}(\mathcal{D}^{Y_S}, \mathcal{D}^{Y_T})^2 \qquad (6)$$

Remark: Since the definitions of normal data are different in source and target domains, $d_{\mathrm{JS}}(\mathcal{D}^{Y_S}, \mathcal{D}^{Y_T}) > 0$. This term is dataset-intrinsic and independent of the learning models. The lower bound explains the phenomenon in Fig. 7: overtraining to minimize ε_S actually increases the error on the target domain ε_T. Learning without adaptation (e.g. IF (S+T)) can have small ε_S but still large error in the target domain. This lower bound also holds for other domain adaptation anomaly detection methods that use invariant representations. This theorem reveals that to have a well-performing model on the target domain, one needs to balance between learning effective invariant representations for accurate AD on the source domain while accommodating the target domain data. This trade-off is hard to avoid and is a consequence of our assumption that the data for T is insufficient for accurate training of the model. So the best outcome is a balanced trade-off between our learning from S and making corrections based on our limited sampling of T. We use cross-validation to estimate the optimal number of training epochs as mentioned before.

We also derive an upper bound for the generalization error. Let f_S, f_T be the true labeling function for the source and target domains respectively. Let $\widehat{D}_S$ and $\widehat{D}_T$ denote the empirical source and target distributions from source domain samples $\mathbf{S}$ and target domain samples $\mathbf{T}$ of size n_t:

Theorem 2. *For a hypothesis space $\mathcal{H} \subseteq [0,1]^X$, $\forall h \in \mathcal{H}$, $\forall \delta > 0$, w.p. at least $1-\delta$:*

$$\begin{aligned}\varepsilon_T(h) \leq\ & \widehat{\varepsilon}_S(h) + d_{\tilde{\mathcal{H}}}\left(\widehat{\mathcal{D}}_S, \widehat{\mathcal{D}}_T\right) \\ & + 2\,\mathrm{Rad}_{\mathbf{S}}(\mathcal{H}) + 2\,\mathrm{Rad}_{\mathbf{S}}(\tilde{\mathcal{H}}) + 2\,\mathrm{Rad}_{\mathbf{T}}(\tilde{\mathcal{H}}) \\ & + \min\left\{\mathbb{E}_{\mathcal{D}_S}\left[|f_S - f_T|\right], \mathbb{E}_{\mathcal{D}_T}\left[|f_S - f_T|\right]\right\} \\ & + O(\sqrt{\log(1/\delta)/n_t}) \\ \tilde{\mathcal{H}} :=\ & \left\{\mathrm{sgn}\left(|h(\mathbf{x}) - h'(\mathbf{x})| - t\right) \mid h, h' \in \mathcal{H}, t \in [0,1]\right\}\end{aligned}$$

$\mathrm{Rad}_{\mathbf{S}}$ *denotes the empirical Rademacher complexity w.r.t. samples* $\mathbf{S}$ *(see the formal definition in the appendix).*

Remark: this bound is formed by the following components (left to right): (1) empirical error on S, (2) distance between the training sets of S and T, (3) complexity measures of $\mathcal{H}$ and $\tilde{\mathcal{H}}$, (4) differences in labels between source and target, (5) error caused by limited target samples.

5. Discussion

Ablation Study of Objective Functions. To better understand the objective functions of IRAD, we conduct the following ablation studies by removing certain terms in the training process. We first investigate Eq. (4) that encourages the similarity between the shared encodings of the source and target data. Ideally, the shared encodings $E_{sh}(\boldsymbol{x}_{tgt})$ and $E_{sh}(\boldsymbol{x}_{src})$ should reside in the same region. For the purpose of illustration, we visualize $E_{sh}(\boldsymbol{x}_{tgt})$ and $E_{sh}(\boldsymbol{x}_{src})$ in 2D by linear PCA as shown in the left sub-figures of Fig. 8(a) and Fig. 8(b). With the similarity objective function in Eq. (4), $E_{sh}(\boldsymbol{x}_{tgt})$ and $E_{sh}(\boldsymbol{x}_{src})$ are close in the latent space (Fig. 8(a)); without Eq. (4), $E_{sh}(\boldsymbol{x}_{tgt})$ and $E_{sh}(\boldsymbol{x}_{src})$ are apart (Fig. 8(b)). We also plotted the magnitude of normalized inner products between 10 $E_{sh}(\boldsymbol{x}_{tgt})$ and $E_{sh}(\boldsymbol{x}_{src})$ in the right sub-figures in Fig. 8(a) and

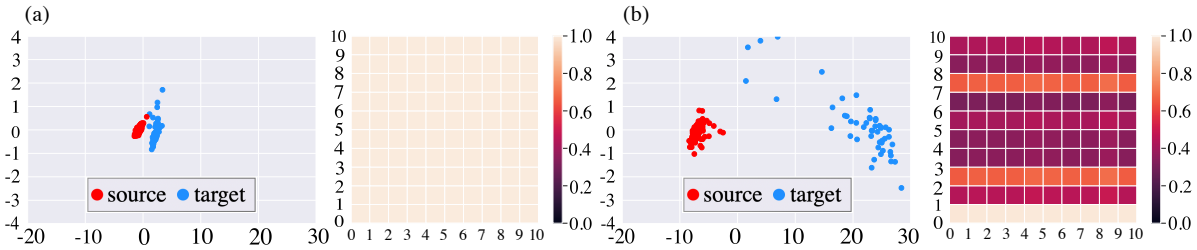

Figure 8. Ablation study (digit 7, MNIST→USPS) on the similarity objective function in Eq. (4). Part (a) is training with Eq. (4): the left figure shows the 2D linear PCA projection of $E_{sh}(\boldsymbol{x}_{tgt})$ and $E_{sh}(\boldsymbol{x}_{src})$. The right sub-figure shows the magnitude of normalized inner products of ten randomly selected $E_{sh}(\boldsymbol{x}_{src})$ and $E_{sh}(\boldsymbol{x}_{tgt})$. Part (b) is trained without Eq. (4). $E_{sh}(\boldsymbol{x}_{tgt})$ and $E_{sh}(\boldsymbol{x}_{src})$ are geometrically and numerically apart from each other in this case.

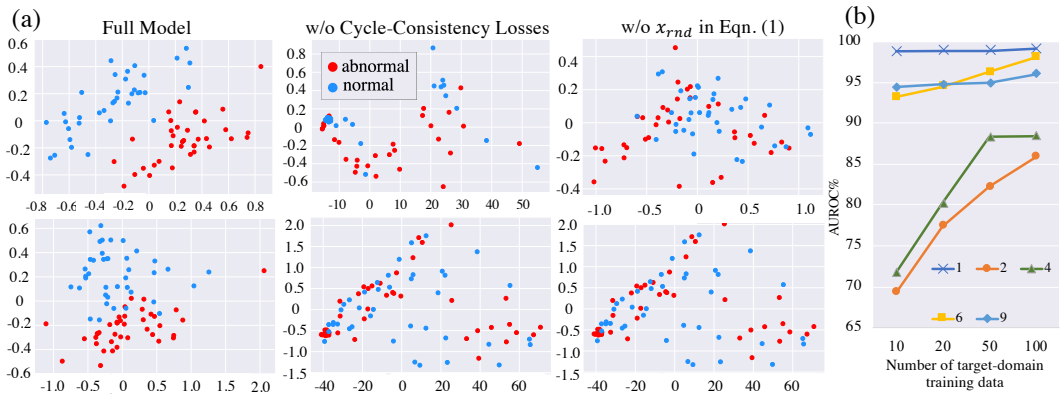

Figure 9. (a) Shared features of normal (blue) and abnormal (red) target domain data. The first and second rows are "Calculator" and "Pan" ("Product"→"Clip Art"). The first column is training with the full model and normal and abnormal encodings are well separated. The second column is training without the cycle-consistency losses. Third column is from removing the term $\boldsymbol{x}_{rnd}$ in Eq. (1). Normal and abnormal data in the two later cases are mixing, making detection hard. (b) AUROC on MNIST→USPS with different numbers of target-domain training data. We only present 5 digits due to the space limit; the full results are provided in the appendix.

Fig. 8(b). The results indicate that optimizing with Eq. (4) indeed makes $E_{sh}(\boldsymbol{x}_{tgt})$ and $E_{sh}(\boldsymbol{x}_{src})$ close numerically.

We further study the cycle-consistency losses in Eq. (2). We find them critical in Office-Home dataset evaluations. Training without them can lead to more than 10% decrease in performance. We visualize the extracted features from the normal and abnormal target data, $E_{sh}(\boldsymbol{x}_{nor})$ and $E_{sh}(\boldsymbol{x}_{abn})$, in 2D with PCA. Ideally, $E_{sh}(\boldsymbol{x}_{nor})$ and $E_{sh}(\boldsymbol{x}_{abn})$ should be separated so the abnormal can be detected. This is what we observe when training with the full model (the first column of Fig. 9(a)). However, if optimized without Eq. (2), encoded normal and abnormal data are mixing together (the second column in Fig. 9(a)). We also investigate term $\boldsymbol{x}_{rnd}$ in Eq. (1). Removing $\boldsymbol{x}_{rnd}$ from the adversarial training results in $E_{sh}(\boldsymbol{x}_{nor})$ and $E_{sh}(\boldsymbol{x}_{abn})$ mingling together (the third column of Fig. 9(a)). We conjecture that for high dimensional data like images, it is challenging for the discriminator to form an effective decision boundary [32], therefore additional regularization terms ($\boldsymbol{x}_{rnd}$) and objective functions (cycle-consistent losses) are helpful for modeling the normal data distribution.

Effects of the number of target domain training data. We investigate IRAD performance w.r.t. the number of target-domain training data n_t. The results are presented in Fig. 9(b) with $n_t = 10, 20, 50, 100$. IRAD is able to leverage more target data to achieve better performance.

6. Conclusion

We proposed IRAD to address the domain adaptation problem in anomaly detection. IRAD first learns invariant representations between the source and target domains. This is achieved by isolating the shared encodings from domain-specific encodings through adversarial learning and enforcing subspace similarity/dissimilarity. The domain-invariant representations are then used to train an anomaly detection model in the target domain. IRAD significantly outperform baseline models in most experiments on real-world anomaly detection datasets. We prove a lower bound for the joint error and an upper bound for the generalization error. Experimental observations corroborate our theoretical results.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017. 3

[2] Sagie Benaim and Lior Wolf. One-shot unsupervised cross domain translation. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 2104–2114. Curran Associates, Inc., 2018. 2

[3] Liron Bergman and Yedid Hoshen. Classification-based anomaly detection for general data. In *International Conference on Learning Representations*, 2020. 1, 2

[4] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger. Mvtec ad — a comprehensive real-world dataset for unsupervised anomaly detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9584–9592, 2019. 4

[5] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. In *Advances in neural information processing systems*, pages 343–351, 2016. 1, 2, 3

[6] Jixu Chen and Xiaoming Liu. Transfer learning with one-class data. *Pattern Recognition Letters*, 37:32–40, 2014. 2

[7] Tomer Cohen and Lior Wolf. Bidirectional one-shot unsupervised domain mapping. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1784–1792, 2019. 2, 4

[8] Dominik Maria Endres and Johannes E Schindelin. A new metric for probability distributions. *IEEE Transactions on Information theory*, 49(7):1858–1860, 2003. 6

[9] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning*, pages 1180–1189, 2015. 1

[10] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016. 2, 6

[11] Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 9758–9769. Curran Associates, Inc., 2018. 2, 5

[12] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1705–1714, 2019. 2

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 2

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[15] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pages 1989–1998. Pmlr, 2018. 1, 2, 4

[16] Tsuyoshi Idé, Dzung T Phan, and Jayant Kalagnanam. Multi-task multi-modal models for collective anomaly detection. In *2017 IEEE International Conference on Data Mining (ICDM)*, pages 177–186. IEEE, 2017. 2

[17] Atsutoshi Kumagai, Tomoharu Iwata, and Yasuhiro Fujiwara. Transfer anomaly detection by inferring latent domain representations. In *Advances in Neural Information Processing Systems*, pages 2467–2477, 2019. 2

[18] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. In *Proceedings of the European conference on computer vision (ECCV)*, pages 35–51, 2018. 2

[19] Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 1991. 11

[20] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining*, pages 413–422. IEEE, 2008. 3, 4

[21] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley. Least squares generative adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2813–2821, Oct 2017. 4

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. 3

[23] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 5

[24] Lukas Ruff, Nico Görnitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Robert Vandermeulen, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *International Conference on Machine Learning*, pages 4390–4399, 2018. 1, 4, 5

[25] Swami Sankaranarayanan, Yogesh Balaji, Carlos D Castillo, and Rama Chellappa. Generate to adapt: Aligning domains using generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8503–8512, 2018. 1

[26] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International Conference on Information Processing in Medical Imaging*, pages 146–157. Springer, 2017. 2

[27] Bernhard Schölkopf, Robert C Williamson, Alex J Smola, John Shawe-Taylor, and John C Platt. Support vector method for novelty detection. In *Advances in neural information processing systems*, pages 582–588, 2000. 3, 4

[28] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176, 2017. 2

[29] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *(IEEE) Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 4, 5

[30] Masataka Yamaguchi, Yuma Koizumi, and Noboru Harada. Adaflow: Domain-adaptive density estimator with application to anomaly detection and unpaired cross-domain translation. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3647–3651. IEEE, 2019. 2, 4

[31] Ziyi Yang. *Deep Learning for Anomaly Detection and Representation Learning*. PhD thesis, Stanford university, 2021. 2

[32] Ziyi Yang, Iman Soltani Bozchalooi, and Eric Darve. Regularized cycle consistent generative adversarial network for anomaly detection. In *ECAI 2020*, pages 1618–1625. IOS Press, 2020. 1, 2, 5, 8

[33] Ziyi Yang, Teng Zhang, Iman Soltani Bozchalooi, and Eric Darve. Memory-augmented generative adversarial networks for anomaly detection. *IEEE Transactions on Neural Networks and Learning Systems*, 33(6):2324–2334, 2021. 2

[34] Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. On learning invariant representations for domain adaptation. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 7523–7532, Long Beach, California, USA, 09–15 Jun 2019. PMLR. 1, 2, 6, 11