Anomaly Heterogeneity Learning for Open-set Supervised Anomaly Detection

Jiawen Zhu[1], Choubo Ding[2], Yu Tian[3], and Guansong Pang[1*]

[1]School of Computing and Information Systems, Singapore Management University
[2]Australian Institute for Machine Learning, University of Adelaide
[3]Harvard Ophthalmology AI Lab, Harvard University

Abstract

Open-set supervised anomaly detection (OSAD) – a recently emerging anomaly detection area – aims at utilizing a few samples of anomaly classes seen during training to detect unseen anomalies (i.e., samples from open-set anomaly classes), while effectively identifying the seen anomalies. Benefiting from the prior knowledge illustrated by the seen anomalies, current OSAD methods can often largely reduce false positive errors. However, these methods are trained in a closed-set setting and treat the anomaly examples as from a homogeneous distribution, rendering them less effective in generalizing to unseen anomalies that can be drawn from any distribution. This paper proposes to learn heterogeneous anomaly distributions using the limited anomaly examples to address this issue. To this end, we introduce a novel approach, namely Anomaly Heterogeneity Learning (AHL), that simulates a diverse set of heterogeneous anomaly distributions and then utilizes them to learn a unified heterogeneous abnormality model in surrogate open-set environments. Further, AHL is a generic framework that existing OSAD models can plug and play for enhancing their abnormality modeling. Extensive experiments on nine real-world anomaly detection datasets show that AHL can 1) substantially enhance different state-of-the-art OSAD models in detecting seen and unseen anomalies, and 2) effectively generalize to unseen anomalies in new domains. Code is available at https://github.com/mala-lab/AHL.

1. Introduction

Anomaly detection (AD) aims at identifying data points that significantly deviate from the majority of the data. It has gained considerable attention in both academic and industry communities due to its broad applications in diverse domains such as industrial inspection, medical imaging, and scientific discovery, etc. [33]. Since it is difficult, or
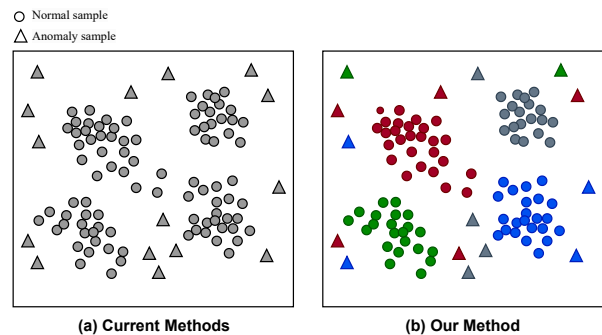

Figure 1. Current methods vs. our method AHL, where the anomaly samples of the same color indicates that they are treated as from one data distribution. Compared to existing methods that model a *homogeneous* anomaly distribution in a closed-set environment, AHL simulates a diverse set of heterogeneous anomaly distributions (Sec. 3.2) and learns *heterogeneous* abnormality from them in a surrogate open environment (Sec. 3.3).

prohibitively costly, to collect large-scale labeled anomaly data, most existing AD approaches treat it as a one-class problem, where only normal samples are available during training [2, 4, 9, 10, 14, 18, 25, 26, 36, 40, 41, 46, 48, 49, 55, 59, 61, 63, 64, 66]. However, in many applications there often exist a few accessible anomaly examples, such as defect samples identified in the past industrial inspection and tumor images of past patients. The anomaly examples offer important source of prior knowledge about abnormality, but these one-class-based approaches are unable to use them.

Open-set supervised AD (OSAD) is a recently emerging area that aims at utilizing those limited training anomaly data to learn generalized models for detecting *unseen anomalies* (*i.e.*, samples from open-set anomaly classes), while effectively identifying those *seen anomalies* (*i.e.*, anomalies that are similar to training anomaly examples). A number of methods have been introduced for this OSAD problem [1, 15, 24, 32, 68]. Benefiting from the prior knowledge illustrated by the seen anomalies, current OSAD

*Corresponding author: G. Pang (gspang@smu.edu.sg)

methods can often largely reduce false positive errors. One issue with the current OSAD methods is that they treat the anomaly examples as from a homogeneous distribution, as shown in Fig. 1(a), which can largely restrict their performance in detecting unseen anomalies. This is because anomalies can arise from a wide range of conditions and are inherently unbounded, resulting in *heterogeneous anomaly distributions* (*i.e.*, anomalies can be drawn from very different distributions). For instance, tumor images can demonstrate different features in terms of appearance, shape, size, and position, etc., depending on the nature of the tumors. The current OSAD methods ignore those anomaly heterogeneity and often fail to detect anomalies if they are drawn from data distributions dissimilar to the seen anomalies.

To address this issue, we propose to learn heterogeneous anomaly distributions with the limited training anomaly examples. These anomalies are examples of seen anomaly classes only, which do not illustrate the distribution of every possible anomaly classes, *e.g.*, those unseen ones, making it challenging to learn the underlying heterogeneous anomaly distributions with the limited anomaly information. This work introduces a novel framework, namely Anomaly Heterogeneity Learning (AHL), to tackle this challenge. As illustrated in Fig. 1(b), it first simulates a variety of heterogeneous anomaly distributions by associating fine-grained distributions of normal examples with randomly selected anomaly examples. AHL then performs a collaborative differentiable learning that synthesizes all these anomaly distributions to learn a heterogeneous abnormality model. Further, the generated anomaly data enables the training of our model in surrogate open environments, in which part of anomaly distributions are used for model training while the others are used as unseen data to validate and tune the model, resulting in better generalized models than the current methods that are trained in a closed-set setting. Additionally, the simulated anomaly distributions are typically of different quality. Thus, a self-supervised generalizability estimation is devised in AHL to adaptively adjust the importance of each learned anomaly distribution during our model training.

A straightforward alternative approach to AHL is to build an ensemble model based on a simple integration of homogeneous/heterogeneous OSAD models on the simulated heterogeneous data distributions. However, such ensembles fail to consider the commonalities and differences in the anomaly heterogeneity captured in the base models, leading to a suboptimal learning of the heterogeneity (Sec. 4.5.2).

Accordingly, this paper makes four main contributions.

- **Framework.** We propose Anomaly Heterogeneity Learning (AHL), a novel framework for OSAD. Unlike current methods that treat the training anomaly examples as a homogeneous distribution, AHL learns heterogeneous anomaly distributions with these limited examples, en-

abling more generalized detection on unseen anomalies.

- **Novel Model.** We further instantiate the AHL framework to a novel OSAD model. The model performs a collaborative differentiable learning of the anomaly heterogeneity using a diverse set of simulated heterogeneous anomaly distributions, facilitating an iterative validating and tuning of the model in surrogate open-set environments. This enables a more optimal anomaly heterogeneity learning than simple ensemble methods.
- **Generic.** Our model is generic, in which features and loss functions from different OSAD models can plug-and-play and gain substantially improved detection performance.
- **Strong Generalization Ability.** Experiments on nine real-world AD datasets show that AHL substantially outperforms state-of-the-art models in detecting unseen anomalies in the same-domain and cross-domain settings.

2. Related Work

Unsupervised Anomaly Detection. Most existing AD approaches rely on unsupervised learning with anomaly-free training samples due to the difficulty of collecting large-scale anomaly observations. One-class classification methods aim to learn a compact normal data description using support vectors [4, 10, 39, 46, 59]. Another widely used AD approach learns to reconstruct normal data based on generative models such as Autoencoder (AE) [21] and Generative Adversarial Networks (GAN) [17]. These reconstruction methods rely on the assumption that anomalies are more difficult to be reconstructed than normal samples [2, 18, 25, 36, 41, 54, 55, 61, 63, 64]. Other popular approaches include knowledge distillation [6, 9, 14, 40, 48, 49, 66] and self-supervised learning methods [16, 19, 22, 37, 57]. Another related line of research is domain-adapted AD [28, 51, 56]. Methods in this line typically focus on a cross-domain setup, requiring the data from multiple relevant domains, whereas we focus on training detection models in single-domain data. One major problem with all these unsupervised AD approaches is that they do not have any prior knowledge about real anomalies, which can lead to many false positive errors [1, 8, 12, 15, 24, 32, 34, 35, 39, 68].

Towards Supervised Anomaly Detection. Supervised AD aims to reduce the detection errors using less costly supervision information, such as weakly-supervised information like video-level supervision to detect frame-level anomalies [11, 29, 43, 47, 52, 53] and a small set of anomaly examples from partially observed anomaly classes [8, 27, 32, 34, 35, 35, 39, 62]. OSAD addresses the problem in the latter case. One OSAD approach is one-class metric learning, where the limited training anomalies are treated as negative samples during the normality learning [24, 31, 39]. However, AD is inherently an open-set task due to the unknowingness nature of anomaly, so the limited negative samples are not sufficient to support an accurate one-class learn-

ing. Recently DevNet [32] introduces a one-sided anomaly-focused deviation loss to tackle this problem by imposing a prior on the anomaly scores. It also establishes an OSAD evaluation benchmark. DRA [15] enhances DevNet via a framework that learns disentangled representations of seen, pseudo, and latent residual anomalies in order to better detect both seen and unseen anomalies. More recently, BGAD [58] uses a decision boundary generated by a normalizing flow model to learn an anomaly-informed model. PRN [65] learns residual representations across multi-scale feature maps using both image-level and pixel-level anomaly data. However, their implementation uses training anomaly examples from all anomaly types, different from our open-set AD settings that have unseen anomaly types in test data. UBnormal [1] and OpenVAD [68] extend OSAD to video data and establish corresponding benchmarks. However, these methods often treat the training anomalies as from homogeneous distributions in a closed-set setting, which can restrict their performance in detecting unseen anomalies. Using the anomaly examples for anomaly generation or pseudo anomaly labeling [3, 62] is explored as another way to reduce the false positives, but it is performed in an unsupervised setting.

3. Anomaly Heterogeneity Learning

Problem Statement: We assume to have a set of training images and annotations $\{(\omega_i, y_i)\}_{i=1}$, where $\omega_i \in \Omega \subset \mathbb{R}^{H \times W \times C}$ denotes an image with RGB channels and $y_i \in \mathcal{Y} \subset \{0, 1\}$ denotes an image-level class label, with $y_i = 1$ if ω_i is abnormal and $y_i = 0$ otherwise. Due to the rareness of anomaly, the labeled data is often predominantly presented by normal data. Given an existing AD model $f(\cdot)$ that can be used to extract low-dimensional image features for constructing the training feature set $\mathcal{D} = \{\mathbf{x}_i, y_i\}$, where $\mathbf{x}_i = f(\omega_i) \in \mathcal{X}$ indicates corresponding i-th image features, with $\mathcal{X}_n = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$ and $\mathcal{X}_a = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_M\}$ ($N \gg M$) respectively denoting the feature set of normal and abnormal images, then the goal of our proposed AHL framework is to learn an anomaly detection function $g : \mathcal{X} \longrightarrow \mathbb{R}$ that is capable of assigning higher anomaly scores to anomaly images drawn from different distributions than to the normal ones. Note that in OSAD the training anomalies $\mathcal{X}_a$ are from seen anomaly classes $\mathcal{S}$, which is only a subset of $\mathcal{C}$ that can contain a larger set of anomaly classes during inference, e.g., $\mathcal{S} \subset \mathcal{C}$.

3.1. Overview of Our Approach

The key idea of our AHL framework is to learn a unified anomaly heterogeneity model by a collaborative differentiable learning of abnormalities embedded in diverse simulated anomaly distributions. As demonstrated in Fig. 2, AHL consists of two main components: Heterogeneous Anomaly Distribution Generation (**HADG**) and Collaborative Differentiable Learning of the anomaly heterogeneity (**CDL**). Specifically, the HADG component simulates and generates T heterogeneous distribution datasets from the training set, $\mathcal{T} = \{\mathcal{D}_i\}_{i=1}^T$, with each $\mathcal{D}_i$ containing a mixture of normal data subset and randomly sampled anomaly examples. Each $\mathcal{D}_i$ is generated in a way that represents a different anomaly distribution from the others. CDL is then designed to learn a unified heterogeneous abnormality detection model $g(\mathcal{T}; \theta_g)$ that synthesizes a set of T base models, denoted as $\{\phi_i(\mathcal{D}_i; \theta_i)\}_{i=1}^T$, where θ_g and θ_i denotes learnable weight parameters of the unified model g and the base model ϕ_i respectively, and each $\phi_i : \mathcal{D}_i \rightarrow \mathbb{R}$ learns from one anomaly distribution for anomaly scoring. The weight parameters θ_g are collaboratively updated based on the base model weights $\{\theta_i\}_{i=1}^T$. Further, the effectiveness of individual base models can vary largely, so a module ψ is added in CDL to increase the importance of θ_i in the collaborative weight updating if its corresponding base model ϕ_i is estimated to have small generalization error. During inference, only the unified heterogeneous abnormality model $g(\mathcal{T}; \theta_g)$ is used for anomaly detection.

AHL is a generic framework, in which off-the-shelf OSAD models can be easily plugged to instantiate ϕ_i and gain significantly improved performance.

3.2. Heterogeneous Anomaly Distribution Data Generation

One main challenge of learning the underlying intricate abnormalities is the lack of training data that illustrate different possible anomaly distributions. Our HADG component is designed to address this challenge, in which we partition the normal examples into different clusters and associate each of the normal clusters with randomly sampled anomaly examples to create diverse anomaly distributions. The resulting distributions differ from each other in terms of normal patterns and/or abnormal patterns. Specifically, HADG generates T sets of training anomaly distribution data, $\mathcal{T} = \{\mathcal{D}_i\}_{i=1}^T$, with each $\mathcal{D}_i = \mathcal{X}_{n,i} \cup \mathcal{X}_{a,i}$, where $\mathcal{X}_{n,i} \subset \mathcal{X}_n$ and $\mathcal{X}_{a,i} \subset \mathcal{X}_a$. To simulate an anomaly distribution of good quality, $\mathcal{X}_{n,i}$ should represent one main normal pattern. To this end, HADG adopts a clustering approach to partition $\mathcal{X}_n$ into C clusters, and then randomly samples one of these C normal clusters as $\mathcal{X}_{n,i}$. On the other hand, to ensure the diversity of anomalies in each $\mathcal{D}_i$, $\mathcal{X}_{a,i}$ is randomly drawn from $\mathcal{X}_a$ and pseudo anomalies generated by popular anomaly generation methods [22, 60, 63].

Further, HADG utilizes those training data to create open-set detection and validation datasets for enabling the training of our model in a surrogate OSAD environment. Particularly, for each $\mathcal{D}_i$, HADG splits it into two disjoint subsets, i.e., $\mathcal{D}_i = \{\mathcal{D}_i^s, \mathcal{D}_i^q\}$, which correspond to support and query sets respectively, with the support set $\mathcal{D}_i^s = \mathcal{X}_{n,i}^s \cup \mathcal{X}_{a,i}^s$ used to train our base model ϕ_i and the query set
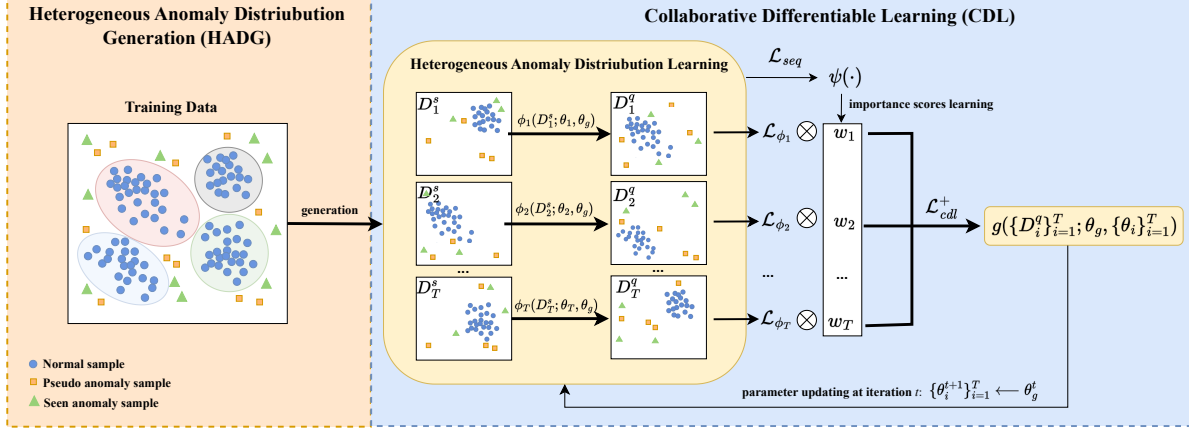
Figure 2. Overview of our approach AHL. Its HADG component first generates T heterogeneous anomaly distribution datasets from the training set, $\mathcal{T} = \{\mathcal{D}_i\}_{i=1}^T$, each of which includes a support set and open-set query set, *i.e.*, $\mathcal{D}_i = \{\mathcal{D}_i^s, \mathcal{D}_i^q\}$. It then utilizes them to learn T heterogeneous AD models $\{\phi_i\}_{i=1}^T$ in a simulated open-set environment and synthesizes these heterogeneous anomaly models into a unified AD model $g(\cdot)$ by a collaborative differential learning (CDL). Different ϕ_i learn anomaly distributions of various quality, so we also devise a model $\psi(\cdot)$ to assign an importance score to each ϕ_i to enhance the CDL component.

$\mathcal{D}_i^q = \mathcal{X}_{n,i}^q \cup \mathcal{X}_{a,i}^q$ used to validate its open-set performance. To guarantee the openness in the validation/query set $\mathcal{D}_i^q$, we perform sampling in a way to ensure that $\mathcal{X}_{n,i}^s$ and $\mathcal{X}_{n,i}^q$ are two different normal clusters, and $\mathcal{X}_{a,i}^s$ and $\mathcal{X}_{a,i}^q$ do not overlap with each other, *i.e.*, $\mathcal{X}_{a,i}^s \cap \mathcal{X}_{a,i}^q = \emptyset$.

3.3. Collaborative Differentiable Learning of the Anomaly Heterogeneity

Our CDL component aims to first learn heterogeneous anomaly distributions hidden in $\mathcal{T} = \{\mathcal{D}_i\}_{i=1}^T$ by using T base model ϕ_i and then utilize these models to collaboratively optimize the unified detection model g in an end-to-end manner. CDL is presented in detail as follows.

Learning T Heterogeneous Anomaly Distributions. We first train T base models $\{\phi_i\}_{i=1}^T$ to respectively capture heterogeneous anomaly distributions in $\{\mathcal{D}_i\}_{i=1}^T$, with each ϕ_i optimized using the following loss:

$$\mathcal{L}_{\phi_i} = \sum_{j=1}^{|\mathcal{D}_i^s|} \ell_{dev}\left(\phi_i(\mathbf{x}_j; \theta_i), y_j\right), \qquad (1)$$

where ℓ_{dev} is specified by a deviation loss [32], following previous OSAD methods DRA [15] and DevNet [32], and $\mathcal{D}_i^s$ is the support set in $\mathcal{D}_i$. Although only limited seen anomalies are available during the training stage, the mixture of normal and anomaly samples in each $\mathcal{D}_i$ differs largely from each other, allowing each ϕ_i to learn a different anomaly distribution for anomaly scoring.

Collaborative Differentiable Learning. Each ϕ_i captures only one part of the whole picture of the underlying anomaly heterogeneity. Thus, we then perform a collaborative differentiable learning that utilizes the losses from the T base models to learn the unified AD model g for capturing richer anomaly heterogeneity. The key insight is that

g is optimized in a way to work well on a variety of possible anomaly distributions, mitigating potential overfitting to a particular anomaly distribution. Further, the optimization of g is based on the losses on the query sets that are not seen during training the base models in Eq. 1, *i.e.*, g is optimized under a surrogate open environment, which helps train a more generalized OSAD model g. Specifically, g is specified to have exactly the same network architecture as the based model ϕ_i, and its weight parameters θ_g at epoch $t+1$ are optimized based on the losses resulted from all base models at epoch t:

$$\theta_g^t \longleftarrow \theta_g^{t-1} - \alpha \nabla \mathcal{L}_{cdl}, \qquad (2)$$

where α is a learning rate and $\mathcal{L}_{cdl}$ is an aggregated loss over T base models on the query sets:

$$\mathcal{L}_{cdl} = \sum_{i=1}^{T} \sum_{j=1}^{|\mathcal{D}_i^q|} \mathcal{L}_{\phi_i}\left(\phi_i(\mathbf{x}_j; \theta_i^t), y_j\right). \qquad (3)$$

In the next training epoch, all θ_i^{t+1} of the base models are set to θ_g^t as their new weight parameters. We then optimize the base models ϕ_i using Eq. 1 on the support sets, and subsequently optimize the unified model g using Eq. 2 on the query sets. This alternative base model and unified model learning is used to obtain an AD model g that increasingly captures richer anomaly heterogeneity.

Learning the Importance Scores of Individual Anomaly Distributions. The quality of the simulated anomaly distribution data $\mathcal{D}_i$ can vary largely, leading to base models of large difference in their effectiveness. Also, a base model that is less effective at one epoch can become more effective at another epoch. Therefore, considering every base model equally throughout the optimization dynamic may lead to

inferior optimization because poorly performing base models can affect the overall performance of the unified model g. To address this issue, we propose a self-supervised sequential modeling module to dynamically estimate the importance of each base model at each epoch. This refines the $\mathcal{L}_{cdl}$ loss as follows:

$$\mathcal{L}_{cdl}^{+} = \sum_{i=1}^{T} \sum_{j=1}^{|\mathcal{D}_i^q|} w_i^t \mathcal{L}_{\phi_i} \left(\phi_i(\mathbf{x}_j; \theta_i^t), y_j \right), \quad (4)$$

where w_i^t denotes the importance score of its base model ϕ_i at epoch t. Below we present how w_i^t is learned via ψ.

Our sequential modeling-based estimation of the dynamic importance score is built upon the intuition that if a base model ϕ_i has good generalization ability, its predicted anomaly scores for different input data should be consistent and accurate at different training stages, where various anomaly heterogeneity gradually emerges as the training unfolds. To this end, we train a sequential model ψ to capture the consistency and accuracy of the anomaly scores yielded by all base models. This is achieved by training ψ to predict the next epoch's anomaly scores of the base models using their previous output anomaly scores. Specifically, given a training sample $\mathbf{x}_j$, the base models $\{\phi_i\}_{i=1}^T$ make a set of anomaly scoring predictions $\mathbf{s}_j = \{s_{ji}\}_{i=1}^T$, resulting in a sequence of score predictions prior to an epoch t, $\mathbf{S}_j^t = [\mathbf{s}_j^{t-K}, \cdots, \mathbf{s}_j^{t-2}, \mathbf{s}_j^{t-1}]$ tracing back to K previous steps, then $\psi : \mathbf{S} \to \mathbb{R}^T$ aims to predict the scoring predictions of all T base models at epoch t. In our implementation, ψ is specified by a sequential neural network parameterized by θ_ψ, and it is optimized using the following next sequence prediction loss:

$$\mathcal{L}_{seq} = \sum_{\mathbf{x}_j \in \mathcal{D}} \mathcal{L}_{mse}(\hat{\mathbf{s}}_j^t, \mathbf{s}_j^t), \quad (5)$$

where $\hat{\mathbf{s}}_j^t = \psi(\mathbf{S}_j^t; \theta_\psi)$ and $\mathbf{s}_j^t$ are respectively the predicted and actual anomaly scores of $\mathbf{x}_j$ from the base models at epoch t, and $\mathcal{L}_{seq}$ is a mean squared error function. Instead of using supervised losses, the model ψ is trained using a self-supervised loss function in Eq. 5 so as to withhold the ground truth labels for avoiding overfitting the labeled data and effectively evaluating the generalization ability of the base models.

The generalization error r_i^t for base model ϕ_i is then defined using the difference between the predicted anomaly score $\hat{s}_{ji}^t$ and the real label y_j as follows:

$$r_i^t = \frac{1}{|\mathcal{D}'|} \sum_{\mathbf{x}_j \in \mathcal{D}'} c_j \mathcal{L}_{mse}(\hat{s}_{ji}^t, y_j), \quad (6)$$

where $\mathcal{D}' = \mathcal{D} \setminus \mathcal{X}_{n,i}$ and c_j is a pre-defined category-wise weight associate with each example $\mathbf{x}_j$. In other words, r_i^t measures the detection error of ϕ_i in predicting the anomaly scores for the seen anomalies in $\mathcal{X}_{a,i}$ and all other unseen normal and anomaly training examples, excluding the seen normal examples $\mathcal{X}_{n,i}$ w.r.t. ϕ_i. A larger c_j is assigned if $\mathbf{x}_j$ is an unseen anomaly to highlight the importance of detecting unseen anomalies; and it is assigned with the same value for the other examples otherwise.

Since a large r_i^t implies a poor generalization ability of the base model ϕ_i at epoch t, we should pay less attention to it when updating the unified model g. Therefore, the importance score of ϕ_i is defined as the inverse of its generalization error as follows:

$$w_i^t = \frac{\exp(-r_i^t)}{\sum_i^T \exp(-r_i^t)}. \quad (7)$$

4. Experiments

4.1. Experimental Setup

Datasets. Following prior OSAD studies [15, 32], we conduct extensive experiments on nine real-world anomaly detection datasets, including five industrial defect inspection datasets (MVTec AD [5], AITEX [42], SDD [44], ELPV [13] and Optical [50]), one planetary exploration dataset (Mastcam [20]), and three medical datasets (HeadCT [40], BrainMRI [40] and Hyper-Kvasir [7]). Depending on how we sample the seen anomaly examples, two protocols are used to evaluate the detection performance, the general and hard settings [15]. The *general setting* assumes the anomaly examples are randomly sampled from the anomaly classes, while the *hard setting* presents a more challenging case where the anomaly examples are sampled exclusively from only one class to evaluate the generalization ability to novel or unseen anomaly classes. Both protocols are used in our experiments. Following [15], we also evaluate the performance with the number of anomaly examples set to respectively $M = 10$ and $M = 1$. Further details about these datasets are available in `Appendix A`.

Competing Methods and Evaluation Metrics. AHL is compared with five closely related state-of-the-art (SOTA) methods, including MLEP [24], SAOE [22, 30, 45], FLOS [23], DevNet [32], and DRA [15]. MLEP, DevNet and DRA are specifically designed for OSAD. SAOE is a supervised detector augmented with synthetic anomalies and outlier exposure, while FLOS is a focal-loss-based imbalanced classifier. For evaluation metrics, we adopt the widely used Area Under ROC Curve (AUC) to measure the performance of all methods and settings. All reported results are averaged results over three independent runs, and stated otherwise.

Implementation Details. To generate a diverse set of anomaly distributions, our proposed approach uses a mixture of randomly selected normal clusters and labeled

Dataset	SAOE	MLEP	FLOS	DevNet	DRA	AHL (DevNet)	AHL (DRA)
Ten Anomaly Examples (Random)							
AITEX	$0.874_{\pm0.024}$	$0.867_{\pm0.037}$	$0.841_{\pm0.049}$	$0.889_{\pm0.007}$	$0.892_{\pm0.007}$	**$0.903_{\pm0.011}$**↑	**$0.925_{\pm0.013}$**↑
SDD	$0.955_{\pm0.020}$	$0.783_{\pm0.013}$	$0.967_{\pm0.018}$	$0.985_{\pm0.004}$	**$0.990_{\pm0.000}$**	**$0.991_{\pm0.001}$**↑	**$0.991_{\pm0.000}$**↑
ELPV	$0.793_{\pm0.047}$	$0.794_{\pm0.047}$	$0.818_{\pm0.032}$	$0.843_{\pm0.001}$	$0.843_{\pm0.002}$	**$0.849_{\pm0.003}$**↑	**$0.850_{\pm0.004}$**↑
Optical	$0.941_{\pm0.013}$	$0.740_{\pm0.039}$	$0.720_{\pm0.055}$	$0.785_{\pm0.012}$	**$0.966_{\pm0.002}$**	$0.841_{\pm0.010}$↑	**$0.976_{\pm0.004}$**↑
Mastcam	$0.810_{\pm0.029}$	$0.798_{\pm0.026}$	$0.703_{\pm0.029}$	$0.797_{\pm0.021}$	**$0.849_{\pm0.003}$**	$0.825_{\pm0.020}$↑	**$0.855_{\pm0.005}$**↑
BrainMRI	$0.900_{\pm0.041}$	$0.959_{\pm0.011}$	$0.955_{\pm0.011}$	$0.951_{\pm0.007}$	**$0.971_{\pm0.001}$**	$0.959_{\pm0.008}$↑	**$0.977_{\pm0.001}$**↑
HeadCT	$0.935_{\pm0.021}$	$0.972_{\pm0.014}$	$0.971_{\pm0.004}$	$0.997_{\pm0.002}$	$0.978_{\pm0.001}$	**$0.999_{\pm0.003}$**↑	**$0.993_{\pm0.002}$**↑
Hyper-Kvasir	$0.666_{\pm0.050}$	$0.600_{\pm0.069}$	$0.773_{\pm0.029}$	$0.822_{\pm0.031}$	$0.844_{\pm0.001}$	**$0.873_{\pm0.009}$**↑	**$0.880_{\pm0.003}$**↑
MVTec AD (mean)	$0.926_{\pm0.010}$	$0.907_{\pm0.005}$	$0.939_{\pm0.007}$	$0.948_{\pm0.005}$	**$0.966_{\pm0.002}$**	$0.954_{\pm0.003}$↑	**$0.970_{\pm0.002}$**↑
One Anomaly Example (Random)							
AITEX	$0.675_{\pm0.094}$	$0.564_{\pm0.055}$	$0.538_{\pm0.073}$	$0.609_{\pm0.054}$	$0.693_{\pm0.031}$	**$0.704_{\pm0.004}$**↑	**$0.734_{\pm0.008}$**↑
SDD	$0.781_{\pm0.009}$	$0.811_{\pm0.045}$	$0.840_{\pm0.043}$	$0.851_{\pm0.003}$	**$0.907_{\pm0.002}$**	$0.864_{\pm0.001}$↑	**$0.909_{\pm0.001}$**↑
ELPV	$0.635_{\pm0.092}$	$0.578_{\pm0.062}$	$0.457_{\pm0.056}$	**$0.810_{\pm0.024}$**	$0.676_{\pm0.003}$	**$0.828_{\pm0.005}$**↑	$0.723_{\pm0.008}$↑
Optical	$0.815_{\pm0.014}$	$0.516_{\pm0.009}$	$0.518_{\pm0.003}$	$0.513_{\pm0.001}$	**$0.880_{\pm0.002}$**	$0.547_{\pm0.009}$↑	**$0.888_{\pm0.007}$**↑
Mastcam	$0.662_{\pm0.018}$	$0.625_{\pm0.045}$	$0.542_{\pm0.017}$	$0.627_{\pm0.049}$	**$0.709_{\pm0.011}$**	$0.644_{\pm0.013}$↑	**$0.743_{\pm0.003}$**↑
BrainMRI	$0.531_{\pm0.060}$	$0.632_{\pm0.017}$	$0.693_{\pm0.036}$	**$0.853_{\pm0.045}$**	$0.747_{\pm0.001}$	**$0.866_{\pm0.004}$**↑	$0.760_{\pm0.013}$↑
HeadCT	$0.597_{\pm0.022}$	$0.758_{\pm0.038}$	$0.698_{\pm0.092}$	$0.755_{\pm0.029}$	**$0.804_{\pm0.010}$**	$0.781_{\pm0.007}$↑	**$0.825_{\pm0.014}$**↑
Hyper-Kvasir	$0.498_{\pm0.100}$	$0.445_{\pm0.040}$	$0.668_{\pm0.004}$	$0.734_{\pm0.020}$	$0.712_{\pm0.010}$	**$0.768_{\pm0.015}$**↑	$0.742_{\pm0.015}$↑
MVTec AD (mean)	$0.834_{\pm0.007}$	$0.744_{\pm0.019}$	$0.792_{\pm0.014}$	$0.832_{\pm0.016}$	**$0.889_{\pm0.013}$**	$0.843_{\pm0.021}$↑	**$0.901_{\pm0.003}$**↑

Table 1. AUC results(mean±std) on nine real-world AD datasets under the general setting. Best results and the second-best results are respectively highlighted in **red** and **bold**. '↑' ('↑') indicates increased performance over DRA (DevNet).

anomaly examples to create each individual anomaly distribution data $\mathcal{D}_i$. Specifically, k-means clustering is first used to partition normal samples into three normal clusters (*i.e.*, $k = 3$ is used). Then two randomly selected clusters are chosen, combining with the seen anomalies, to construct $\mathcal{D}_i$, having one normal clusters and 50% of the seen anomaly set as the support set $\mathcal{D}_i^s$ while the rest of samples are used as the query set $\mathcal{D}_i^q$ (Under the protocol of having only one seen anomaly example, the example is included in both sets). This helps effectively simulate open-set environments with partially observed anomaly distributions. To further increase the heterogeneity within and between the anomaly distribution datasets, we randomly pick one of the three popular anomaly generation techniques, including CutMix [60], CutPaste [22], and DRAEM Mask [63], to generate and inject pseudo anomalies into the support and query sets of $\mathcal{D}_i$. To guarantee the openness w.r.t. the pseudo anomaly detection, the pseudo anomalies in $\mathcal{D}_i^s$ and $\mathcal{D}_i^q$ are generated from two different anomaly generation approaches. For each dataset, $T = 6$ is used in generating the individual anomaly distribution data. c_j is set to 1.0 when $\mathbf{x}_j$ represents unseen anomaly samples, and 0.5 when $\mathbf{x}_j$ represents seen anomaly or unseen normal samples.

AHL is a generic framework, under which features and loss functions from existing OSAD models can be easily plugged in as the base features and the base loss. Particularly, the image features are extracted from one of the OSAD model (*e.g.*, DRA), and then AHL is trained using our proposed loss function built on the base loss (see Eq. 4). DRA [15], DevNet [32] and BGAD [58] are the current SOTA models for OSAD, but BGAD uses quite different benchmark datasets from the other two. Our experiments strictly follow the seminal OSAD evaluation protocol

and benchmarks used in DRA [15] and DevNet [32], and choose DRA [15] and DevNet [32] to respectively plug in AHL, denoting as **AHL (DRA)** and **AHL (DevNet)**. Adam is used as the optimizer. The initial learning rate for learning heterogeneous T base models is set to 0.0002, while that for the unified AD model g is set to 0.002. In the self-supervised importance score estimator, a two-layer Bidirectional LSTM [67] is used as the backbone, with the hidden dimension set to 6. It is followed by a fully-connected layer with 12 hidden nodes, before the prediction layer. The initial learning rate is set to 0.002 for this component.

The above settings are used by default for the reported results of AHL across all the datasets. The results of MLEP, SAOE, and FLOS are taken from [15]. The results of DevNet and DRA are reproduced using their official codes to obtain the features used in AHL, meaning that DevNet and AHL (DevNet) use the same set of features, which also applies to DRA and AHL (DRA) (see Appendix B for more implementation details).

4.2. Performance under General Settings

Table 1 shows the comparison results under the general setting, where models are trained using one or ten randomly sampled anomaly examples. The results on MVTec AD are averaged over its 16 data subsets (see Appendix C for detailed results on the subsets). Overall, our approach AHL brings consistently substantial improvement to the respective DRA and DevNet in both ten-shot and one-shot setting protocols across all the datasets of three application scenarios. Since DRA is a stronger base model than DevNet, AHL (DRA) generally obtains much better performance than AHL (DevNet). On average, AHL respectively enhances the performance of DRA and DevNet by up to

Dataset	SAOE	MLEP	FLOS	DevNet	DRA	AHL (DevNet)	AHL (DRA)
Ten Examples from One Anomaly Class							
Carpet (mean)	$0.762_{\pm0.073}$	$0.935_{\pm0.013}$	$0.761_{\pm0.012}$	$0.853_{\pm0.005}$	**$0.940_{\pm0.006}$**	$0.860_{\pm0.013}\uparrow$	**$0.949_{\pm0.002}$**$\uparrow$
Metal_nut (mean)	$0.855_{\pm0.016}$	$0.945_{\pm0.017}$	$0.922_{\pm0.014}$	$0.970_{\pm0.009}$	$0.968_{\pm0.006}$	**$0.972_{\pm0.002}$**$\uparrow$	$0.971_{\pm0.001}\uparrow$
AITEX (mean)	$0.724_{\pm0.032}$	$0.733_{\pm0.009}$	$0.635_{\pm0.043}$	$0.685_{\pm0.016}$	**$0.733_{\pm0.011}$**	$0.709_{\pm0.006}\uparrow$	**$0.747_{\pm0.002}$**$\uparrow$
ELPV (mean)	$0.683_{\pm0.047}$	$0.766_{\pm0.029}$	$0.646_{\pm0.032}$	$0.722_{\pm0.018}$	**$0.771_{\pm0.005}$**	$0.752_{\pm0.005}\uparrow$	**$0.788_{\pm0.003}$**$\uparrow$
Mastcam (mean)	$0.697_{\pm0.014}$	$0.695_{\pm0.004}$	$0.616_{\pm0.021}$	$0.588_{\pm0.025}$	**$0.704_{\pm0.007}$**	$0.602_{\pm0.008}\uparrow$	**$0.721_{\pm0.003}$**$\uparrow$
Hyper-Kvasir (mean)	$0.698_{\pm0.021}$	$0.844_{\pm0.009}$	$0.786_{\pm0.021}$	$0.827_{\pm0.008}$	$0.822_{\pm0.013}$	**$0.845_{\pm0.003}$**$\uparrow$	**$0.854_{\pm0.004}$**$\uparrow$
One Example from One Anomaly Class							
Carpet (mean)	$0.753_{\pm0.055}$	$0.679_{\pm0.029}$	$0.678_{\pm0.040}$	$0.774_{\pm0.007}$	**$0.905_{\pm0.006}$**	$0.785_{\pm0.015}\uparrow$	**$0.932_{\pm0.003}$**$\uparrow$
Metal_nut (mean)	$0.816_{\pm0.029}$	$0.825_{\pm0.023}$	$0.855_{\pm0.024}$	$0.861_{\pm0.019}$	**$0.936_{\pm0.011}$**	$0.869_{\pm0.004}\uparrow$	**$0.939_{\pm0.004}$**$\uparrow$
AITEX (mean)	$0.674_{\pm0.034}$	$0.466_{\pm0.030}$	$0.624_{\pm0.024}$	$0.646_{\pm0.014}$	**$0.696_{\pm0.011}$**	$0.660_{\pm0.007}\uparrow$	**$0.707_{\pm0.007}$**$\uparrow$
ELPV (mean)	$0.614_{\pm0.048}$	$0.566_{\pm0.111}$	$0.691_{\pm0.008}$	$0.663_{\pm0.008}$	**$0.722_{\pm0.006}$**	$0.678_{\pm0.006}\uparrow$	**$0.740_{\pm0.003}$**$\uparrow$
Mastcam (mean)	$0.689_{\pm0.037}$	$0.541_{\pm0.007}$	$0.524_{\pm0.013}$	$0.519_{\pm0.014}$	**$0.658_{\pm0.021}$**	$0.535_{\pm0.003}\uparrow$	**$0.673_{\pm0.010}$**$\uparrow$
Hyper-Kvasir (mean)	$0.406_{\pm0.018}$	$0.480_{\pm0.044}$	$0.571_{\pm0.004}$	$0.598_{\pm0.006}$	**$0.699_{\pm0.009}$**	$0.619_{\pm0.005}\uparrow$	**$0.706_{\pm0.007}$**$\uparrow$

Table 2. AUC results(mean±std) on nine real-world AD datasets under the hard setting. Best results and the second-best results are respectively highlighted in **red** and **bold**.'$\uparrow$' ('$\uparrow$') indicates increased performance over DRA (DevNet). Carpet and Meta_nut are two subsets of MVTec AD. The same set of datasets is used as in [15].

4% and 9% AUC, indicating that the anomaly heterogeneity learned by AHL enable the base models to gain significantly improved generalization ability.

4.3. Generalization to Unseen Anomaly Classes

Table 2 shows the comparison results under the hard setting, where models are trained with one or ten examples randomly sampled from exclusively one known anomaly class to detect the anomalies in the rest of all other anomaly classes. This protocol means that we can obtain one AUC result for having each anomaly class as the seen anomaly class. The reported results here are averaged over all the anomaly classes (see Appendix C for detailed class-level results). In general, our models – AHL (DRA) and AHL (DevNet) – achieves the best AUC results in both $M = 1$ and $M = 10$ settings. To be more precise, AHL respectively improves the performances of DRA and DevNet by up to 3.2% and 3% AUC. Similar to the general setting, AHL (DRA) is the best performer. Since the model is only exposed to one anomaly class, the improvement is fully due to the ability of AHL in generalizing to unseen anomaly classes.

4.4. Unseen Anomaly Detection in Novel Domains

We evaluate the effectiveness of AHL in generalizing to unseen anomalies in a novel domain (*i.e.*, a cross-domain AD task), where a model is trained on a source domain and is subsequently tested on datasets from a target domain that differs from the source. This is used to further verify the generalization capacity of AHL. Specifically, following the DRA work [15], we employ the DRA and AHL (DRA) models, trained on one of five datasets (the source domain) and fine-tune them for 10 epochs using only normal samples on the other four datasets (the target domains). The results of this experiment are presented in Table 3. Overall, the AHL (DRA) model outperforms the DRA baseline significantly in this setting, and it can even achieves comparable AUC to the same-domain performance (*i.e.*, the diagonal results). However, we do observe a slight drop in

	DRA					AHL (DRA)				
	Carpet	Grid	Leather	Tile	Wood	Carpet	Grid	Leather	Tile	Wood
Carpet	0.945	0.833	0.921	0.930	0.917	0.953	**0.979**	**1.000**	**1.000**	**0.998**
Grid	**0.983**	0.990	0.924	0.940	0.916	0.959	0.992	**1.000**	**1.000**	**0.973**
Leather	**0.988**	0.998	1.000	0.994	**1.000**	0.963	0.998	1.000	**1.000**	0.998
Tile	0.917	0.971	0.958	1.000	0.955	**0.943**	**0.982**	**1.000**	1.000	**0.999**
Wood	0.993	0.985	0.972	0.948	0.998	**0.995**	**0.989**	**1.000**	**1.000**	0.998

Table 3. AUC results of DRA and AHL (DRA) on detecting texture anomalies in cross-domain datasets (all are MVTec AD datasets). The top row is the source domain for training, while the left column is the target domain for inference. The same-domain results are shown in *gray* for reference. Best results are **boldfaced**.

performance on some target domains. We attribute this to the learned anomaly heterogeneity being based on anomaly samples from the source domain, which may introduce bias when testing the AD model on the target domains. Nevertheless, our findings indicate that the AHL framework enhances the generalization ability of DRA on novel domains, demonstrating promising cross-domain performance.

4.5. Analysis of AHL

4.5.1 Utility of Few-shot Samples

To investigate the utility of few-shot samples in the OSAD task, AHL is also compared with state-of-the-art models of unsupervised anomaly detection (UAD) and fully-supervised anomaly detection (FSAD), including KDAD [40] and PatchCore [38] for UAD and fully supervised DRA (FS-DRA for short) for FSAD. DRA is transformed to a fully supervised approach FS-DRA by using a set of 10 anomaly examples that illustrate all possible anomaly classes in the test data. The results of AHL (DRA) using randomly sampled 10 anomaly examples are used here for comparison. The results are presented in Table 4. It shows that AHL (DRA) substantially outperforms the unsupervised detectors KDAD and PatchCore, demonstrating better generalization ability than the unsupervised methods. FS-DRA is the best-performing model on six out of nine datasets. This is expected due to its fully supervised nature. Although AHL (DRA) is an open-set detector, it is the best performer on AITEX and SDD and performs comparably

Dataset	KDAD	PatchCore	AHL (DRA)	FS-DRA
AITEX	$0.576_{\pm 0.002}$	0.783	$0.925_{\pm 0.013}$	$0.919_{\pm 0.004}$
SDD	$0.842_{\pm 0.006}$	**0.873**	$0.991_{\pm 0.000}$	$0.991_{\pm 0.000}$
ELPV	$0.744_{\pm 0.001}$	0.916	$0.850_{\pm 0.004}$	$0.874_{\pm 0.004}$
Optical	$0.579_{\pm 0.002}$	-	$0.976_{\pm 0.004}$	$0.982_{\pm 0.000}$
Mastcam	$0.642_{\pm 0.007}$	0.809	$0.855_{\pm 0.005}$	$0.877_{\pm 0.003}$
BrainMRI	$0.733_{\pm 0.016}$	0.754	$0.977_{\pm 0.001}$	$0.979_{\pm 0.001}$
HeadCT	$0.793_{\pm 0.017}$	0.864	$0.993_{\pm 0.002}$	$0.998_{\pm 0.003}$
Hyper-Kvasir	$0.401_{\pm 0.002}$	0.494	$0.880_{\pm 0.003}$	$0.900_{\pm 0.009}$
MVTec AD (mean)	$0.863_{\pm 0.029}$	0.992	$0.970_{\pm 0.002}$	$0.984_{\pm 0.004}$

Table 4. AUC results comparison of AHL (DRA) to unsupervised anomaly detection methods – KDAD and PatchCore – and a fully-supervised DRA model (FS-DRA). Best results and the second-best results are respectively highlighted in **red** and **bold**

well to the fully-supervised model FS-DRA on the other seven datasets, indicating that AHL (DRA) can accurately generalize to unseen anomalies while maintaining the effectiveness on the seen anomalies. These results suggest a very effective utilization of the few-shot examples in AHL, while avoiding the overfitting of the seen anomalies.

4.5.2 Ablation Study

Our ablation study uses DRA as the baseline. To evaluate our first component HADG, we compare DRA with its three variants, including an ensemble of DRA trained on the full data using different random seeds (**+ Ensemble**); DRA supplemented with the initial CDL component (specified in Eq. 3) and trained on random subsets of the data (**+CDL**) rather than the subsets generated by HADG; and DRA enhanced with both the initial CDL component and the HADG component (**+ HADG + CDL**). We then examine the effectiveness of ψ by by substituting CDL with CDL$^+$ (*i.e.* Eq. 4) within the **+ HADG + CDL** configuration, which is also our full AHL model, denoted as '**+ HADG+ CDL$^+$**'. 'We further compare **+ HADG+ CDL$^+$**' with its simplified version '**+ HADG+ CDL$^-$**' in which the weights w_i of ϕ_i are simply computed based on the detection accuracy on the training data excluding $\mathcal{D}_i^s$ instead of using our model importance learning ψ. Table 5 illustrates the results under both settings using $M = 10$. We can observe that using only HADG and the initial CDL, '**+ HADG + CDL**', largely improves DRA on most datasets, and it also substantially outperforms both the simple DRA ensemble and the variant trained on randomly sampled data distribution subsets, showing the effectiveness of HADG component. Replacing the CDL component to '**+ CDL$^+$**' consistently improves the variant '**+ HADG + CDL**' across all datasets. Simplifying '**+ CDL$^+$**' to '**+ CDL$^-$**' leads to performance drop on most datasets, with relatively large drops on challenging datasets like AITEX (both settings), Mastcam, and Hyper-Kvasir, indicating the significance of ψ in CDL$^+$.

4.5.3 Hyperparameter Analysis

We conduct an analysis of two key hyperparameters in AHL, the number of normal clusters (C) in HADG (Sec. 3.2) and

the length of score sequences (K) in CDL (Sec. 3.3), with the results shown in Fig. 3 (left) and Fig. 3 (right), respectively. We observe that increasing C does not always result in improved performance. This is mainly because that excessively dividing normal samples into too many small clusters can introduce biased individual anomaly distributions into the learning process due to the presence of too few samples per cluster. As for K, the results show that setting $K = 3$ is generally sufficient for accurate generalization error estimation. However, having $K = 5$ can further improve the performance on a few datasets where more accurate generalization error estimation requires longer input sequences. $C = 3$ and $K = 5$ are generally suggested, which are the default settings for AHL models throughout our large-scale experiments.

Dataset	DRA	+Ensemble	+CDL	+HADG+CDL	+HADG+CDL$^-$	+HADG+CDL$^+$
General Setting						
AITEX	$0.892_{\pm 0.007}$	$0.905_{\pm 0.011}$	$0.908_{\pm 0.003}$	$0.916_{\pm 0.004}$	$0.915_{\pm 0.007}$	$0.925_{\pm 0.013}$
SDD	$0.990_{\pm 0.000}$	$0.985_{\pm 0.002}$	$0.990_{\pm 0.000}$	$0.990_{\pm 0.000}$	$0.991_{\pm 0.001}$	$0.991_{\pm 0.000}$
ELPV	$0.843_{\pm 0.002}$	$0.843_{\pm 0.000}$	$0.844_{\pm 0.002}$	$0.846_{\pm 0.006}$	$0.847_{\pm 0.002}$	$0.850_{\pm 0.004}$
Optical	$0.966_{\pm 0.002}$	$0.963_{\pm 0.006}$	$0.970_{\pm 0.004}$	$0.974_{\pm 0.002}$	$0.974_{\pm 0.002}$	$0.976_{\pm 0.004}$
Mastcam	$0.849_{\pm 0.003}$	$0.849_{\pm 0.001}$	$0.815_{\pm 0.003}$	$0.852_{\pm 0.001}$	$0.841_{\pm 0.004}$	$0.855_{\pm 0.005}$
BrainMRI	$0.971_{\pm 0.001}$	$0.973_{\pm 0.007}$	$0.974_{\pm 0.001}$	$0.973_{\pm 0.002}$	$0.977_{\pm 0.002}$	$0.977_{\pm 0.001}$
HeadCT	$0.978_{\pm 0.001}$	$0.988_{\pm 0.003}$	$0.986_{\pm 0.001}$	$0.992_{\pm 0.002}$	$0.991_{\pm 0.001}$	$0.993_{\pm 0.002}$
Hyper-Kvasir	$0.844_{\pm 0.001}$	$0.863_{\pm 0.001}$	$0.865_{\pm 0.002}$	$0.874_{\pm 0.005}$	$0.877_{\pm 0.005}$	$0.880_{\pm 0.003}$
MVTecAD(mean)	$0.966_{\pm 0.002}$	$0.968_{\pm 0.004}$	$0.964_{\pm 0.003}$	$0.968_{\pm 0.003}$	$0.966_{\pm 0.003}$	$0.970_{\pm 0.002}$
Hard Setting						
carpet(mean)	$0.940_{\pm 0.006}$	$0.947_{\pm 0.005}$	$0.943_{\pm 0.002}$	$0.943_{\pm 0.003}$	$0.951_{\pm 0.002}$	$0.949_{\pm 0.002}$
AITEX(mean)	$0.733_{\pm 0.011}$	$0.733_{\pm 0.004}$	$0.733_{\pm 0.005}$	$0.739_{\pm 0.007}$	$0.720_{\pm 0.005}$	$0.747_{\pm 0.002}$
elpv(mean)	$0.771_{\pm 0.005}$	$0.779_{\pm 0.002}$	$0.774_{\pm 0.004}$	$0.784_{\pm 0.004}$	$0.779_{\pm 0.001}$	$0.788_{\pm 0.003}$
Hyper-Kvasir(mean)	$0.822_{\pm 0.013}$	$0.829_{\pm 0.003}$	$0.835_{\pm 0.004}$	$0.847_{\pm 0.008}$	$0.838_{\pm 0.005}$	$0.854_{\pm 0.004}$

Table 5. Ablation study results of AHL and its variants under both general and hard settings. DRA serves as the base model, to which the other variants are applied.
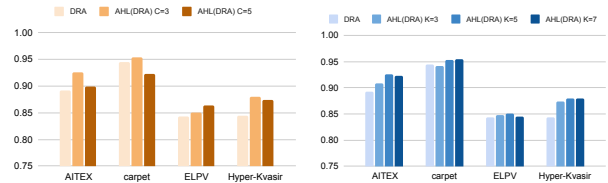

Figure 3. Hyperparameter analysis of AHL (DRA) based on the general setting using ten anomaly examples. **Left:** AUC results w.r.t. different number of clusters (C). **Right:** AUC results w.r.t. the length of sequences (K) as input to the sequential model ψ.

5. Conclusion

In this work, we explore the OSAD problem and introduce a novel, generic framework, namely anomaly heterogeneity learning (AHL). It learns generalized, heterogeneous abnormality detection capability by training on diverse generated anomaly distributions in simulated OSAD scenarios. AHL models such anomaly heterogeneity using a collaborative differentiable learning on a set of heterogeneous based models built on the generated anomaly distribution. Experiments on nine real-world anomaly detection datasets demonstrate that the AHL approach can substantially enhance different state-of-the-art OSAD models in detecting unseen anomalies in the same-domain and cross-domain cases, with the AUC improvement up to 9%.

References

[1] Andra Acsintoae, Andrei Florescu, Mariana-Iuliana Georgescu, Tudor Mare, Paul Sumedrea, Radu Tudor Ionescu, Fahad Shahbaz Khan, and Mubarak Shah. Ubnormal: New benchmark for supervised open-set video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20143–20153, 2022. 1, 2, 3

[2] Samet Akcay, Amir Atapour-Abarghouei, and Toby P Breckon. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part III 14*, pages 622–637. Springer, 2019. 1, 2

[3] Marcella Astrid, Muhammad Zaigham Zaheer, and Seung-Ik Lee. Pseudobound: Limiting the anomaly reconstruction capability of one-class classifiers using pseudo anomalies. *Neurocomputing*, 534:147–160, 2023. 3

[4] Liron Bergman and Yedid Hoshen. Classification-based anomaly detection for general data. *arXiv preprint arXiv:2005.02359*, 2020. 1, 2

[5] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9592–9600, 2019. 5

[6] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4183–4192, 2020. 2

[7] Hanna Borgli, Vajira Thambawita, Pia H Smedsrud, Steven Hicks, Debesh Jha, Sigrun L Eskeland, Kristin Ranheim Randel, Konstantin Pogorelov, Mathias Lux, Duc Tien Dang Nguyen, et al. Hyperkvasir, a comprehensive multi-class image and video dataset for gastrointestinal endoscopy. *Scientific data*, 7(1):283, 2020. 5

[8] Jakob Bozic, Domen Tabernik, and Danijel Skocaj. Mixed supervision for surface-defect detection: From weakly to fully supervised learning. *Comput. Ind.*, 129:103459, 2021. 2

[9] Tri Cao, Jiawen Zhu, and Guansong Pang. Anomaly detection under distribution shift. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6511–6523, 2023. 1, 2

[10] Yuanhong Chen, Yu Tian, Guansong Pang, and Gustavo Carneiro. Deep one-class classification via interpolated gaussian descriptor. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 383–392, 2022. 1, 2

[11] Yingxian Chen, Zhengzhe Liu, Baoheng Zhang, Wilton Fok, Xiaojuan Qi, and Yik-Chung Wu. Mgfn: Magnitude-contrastive glance-and-focus network for weakly-supervised video anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 387–395, 2023. 2

[12] Wen-Hsuan Chu and Kris M. Kitani. Neural batch sampling with reinforcement learning for semi-supervised anomaly detection. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXVI*, pages 751–766. Springer, 2020. 2

[13] Sergiu Deitsch, Vincent Christlein, Stephan Berger, Claudia Buerhop-Lutz, Andreas Maier, Florian Gallwitz, and Christian Riess. Automatic classification of defective photovoltaic module cells in electroluminescence images. *Solar Energy*, 185:455–468, 2019. 5

[14] Hanqiu Deng and Xingyu Li. Anomaly detection via reverse distillation from one-class embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9737–9746, 2022. 1, 2

[15] Choubo Ding, Guansong Pang, and Chunhua Shen. Catching both gray and black swans: Open-set supervised anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7388–7398, 2022. 1, 2, 3, 4, 5, 6, 7

[16] Mariana-Iuliana Georgescu, Antonio Barbalau, Radu Tudor Ionescu, Fahad Shahbaz Khan, Marius Popescu, and Mubarak Shah. Anomaly detection in video via self-supervised and multi-task learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12742–12752, 2021. 2

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 2

[18] Jinlei Hou, Yingying Zhang, Qiaoyong Zhong, Di Xie, Shiliang Pu, and Hong Zhou. Divide-and-assemble: Learning block-wise memory for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8791–8800, 2021. 1, 2

[19] Chaoqin Huang, Qinwei Xu, Yanfeng Wang, Yu Wang, and Ya Zhang. Self-supervised masking for unsupervised anomaly detection and localization. *IEEE Transactions on Multimedia*, 2022. 2

[20] Hannah R Kerner, Kiri L Wagstaff, Brian D Bue, Danika F Wellington, Samantha Jacob, Paul Horton, James F Bell, Chiman Kwan, and Heni Ben Amor. Comparison of novelty detection methods for multispectral images in rover-based planetary exploration missions. *Data Mining and Knowledge Discovery*, 34:1642–1675, 2020. 5

[21] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2

[22] Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9664–9674, 2021. 2, 3, 5, 6

[23] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 5

[24] Wen Liu, Weixin Luo, Zhengxin Li, Peilin Zhao, Shenghua Gao, et al. Margin learning embedded prediction for video anomaly detection with a few anomalies. In *IJCAI*, pages 3023–3030, 2019. 1, 2, 5

[25] Wenrui Liu, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Diversity-measurable anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12147–12156, 2023. 1, 2

[26] Zhikang Liu, Yiming Zhou, Yuansheng Xu, and Zilei Wang. Simplenet: A simple network for image anomaly detection and localization, 2023. 1

[27] Philipp Liznerski, Lukas Ruff, Robert A. Vandermeulen, Billy Joe Franks, Marius Kloft, and Klaus-Robert Müller. Explainable deep one-class classification. *CoRR*, abs/2007.01760, 2020. 2

[28] Yiwei Lu, Frank Yu, Mahesh Kumar Krishna Reddy, and Yang Wang. Few-shot scene-adaptive anomaly detection. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part V*, pages 125–141. Springer, 2020. 2

[29] Hui Lv, Zhongqi Yue, Qianru Sun, Bin Luo, Zhen Cui, and Hanwang Zhang. Unbiased multiple instance learning for weakly supervised video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8022–8031, 2023. 2

[30] Amir Markovitz, Gilad Sharir, Itamar Friedman, Lihi Zelnik-Manor, and Shai Avidan. Graph embedded pose clustering for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10539–10547, 2020. 5

[31] Guansong Pang, Longbing Cao, Ling Chen, and Huan Liu. Learning representations of ultrahigh-dimensional data for random distance-based outlier detection. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2041–2050, 2018. 2

[32] Guansong Pang, Choubo Ding, Chunhua Shen, and Anton van den Hengel. Explainable deep few-shot anomaly detection with deviation networks. *arXiv preprint arXiv:2108.00462*, 2021. 1, 2, 3, 4, 5, 6

[33] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep learning for anomaly detection: A review. *ACM computing surveys (CSUR)*, 54(2):1–38, 2021. 1

[34] Guansong Pang, Anton van den Hengel, Chunhua Shen, and Longbing Cao. Toward deep supervised anomaly detection: Reinforcement learning from partially labeled anomaly data. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pages 1298–1308, 2021. 2

[35] Guansong Pang, Chunhua Shen, Huidong Jin, and Anton van den Hengel. Deep weakly-supervised anomaly detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1795–1807, 2023. 2

[36] Hyunjong Park, Jongyoun Noh, and Bumsub Ham. Learning memory-guided normality for anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14372–14381, 2020. 1, 2

[37] Nicolae-Cătălin Ristea, Neelu Madan, Radu Tudor Ionescu, Kamal Nasrollahi, Fahad Shahbaz Khan, Thomas B Moes-

lund, and Mubarak Shah. Self-supervised predictive convolutional attentive block for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13576–13586, 2022. 2

[38] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14318–14328, 2022. 7

[39] Lukas Ruff, Robert A Vandermeulen, Nico Görnitz, Alexander Binder, Emmanuel Müller, Klaus-Robert Müller, and Marius Kloft. Deep semi-supervised anomaly detection. In *ICLR*, 2020. 2

[40] Mohammadreza Salehi, Niousha Sadjadi, Soroosh Baselizadeh, Mohammad H Rohban, and Hamid R Rabiee. Multiresolution knowledge distillation for anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14902–14912, 2021. 1, 2, 5, 7

[41] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Georg Langs, and Ursula Schmidt-Erfurth. f-anogan: Fast unsupervised anomaly detection with generative adversarial networks. *Medical image analysis*, 54:30–44, 2019. 1, 2

[42] Javier Silvestre-Blanes, Teresa Albero-Albero, Ignacio Miralles, Rubén Pérez-Llorens, and Jorge Moreno. A public fabric database for defect detection methods and results. *Autex Research Journal*, 19(4):363–374, 2019. 5

[43] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6479–6488, 2018. 2

[44] Domen Tabernik, Samo Šela, Jure Skvarč, and Danijel Skočaj. Segmentation-based deep-learning approach for surface-defect detection. *Journal of Intelligent Manufacturing*, 31(3):759–776, 2020. 5

[45] Jihoon Tack, Sangwoo Mo, Jongheon Jeong, and Jinwoo Shin. Csi: Novelty detection via contrastive learning on distributionally shifted instances. *Advances in neural information processing systems*, 33:11839–11852, 2020. 5

[46] David MJ Tax and Robert PW Duin. Support vector data description. *Machine learning*, 54:45–66, 2004. 1, 2

[47] Yu Tian, Guansong Pang, Yuanhong Chen, Rajvinder Singh, Johan W Verjans, and Gustavo Carneiro. Weakly-supervised video anomaly detection with robust temporal feature magnitude learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4975–4986, 2021. 2

[48] Tran Dinh Tien, Anh Tuan Nguyen, Nguyen Hoang Tran, Ta Duc Huy, Soan Duong, Chanh D Tr Nguyen, and Steven QH Truong. Revisiting reverse distillation for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24511–24520, 2023. 1, 2

[49] Guodong Wang, Shumin Han, Errui Ding, and Di Huang. Student-teacher feature pyramid matching for anomaly detection. *arXiv preprint arXiv:2103.04257*, 2021. 1, 2

[50] Matthias Wieler and Tobias Hahn. Weakly supervised learning for industrial optical inspection. In *DAGM symposium in*, 2007. 5

[51] Jhih-Ciang Wu, Ding-Jie Chen, Chiou-Shann Fuh, and Tyng-Luh Liu. Learning unsupervised metaformer for anomaly detection. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 4349–4358. IEEE, 2021. 2

[52] Peng Wu, Jing Liu, Yujia Shi, Yujia Sun, Fangtao Shao, Zhaoyang Wu, and Zhiwei Yang. Not only look, but also listen: Learning multimodal violence detection under weak supervision. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXX 16*, pages 322–339. Springer, 2020. 2

[53] Peng Wu, Xuerong Zhou, Guansong Pang, Lingru Zhou, Qingsen Yan, Peng Wang, and Yanning Zhang. Vadclip: Adapting vision-language models for weakly supervised video anomaly detection. *arXiv preprint arXiv:2308.11681*, 2023. 2

[54] Tiange Xiang, Yixiao Zhang, Yongyi Lu, Alan L Yuille, Chaoyi Zhang, Weidong Cai, and Zongwei Zhou. Squid: Deep feature in-painting for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23890–23901, 2023. 2

[55] Xudong Yan, Huaidong Zhang, Xuemiao Xu, Xiaowei Hu, and Pheng-Ann Heng. Learning semantic context from normal samples for unsupervised anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3110–3118, 2021. 1, 2

[56] Huaxiu Yao, Linjun Zhang, and Chelsea Finn. Meta-learning with fewer tasks through task interpolation. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. 2

[57] Xincheng Yao, Ruoqi Li, Zefeng Qian, Yan Luo, and Chongyang Zhang. Focus the discrepancy: Intra- and inter-correlation learning for image anomaly detection. 2023. 2

[58] Xincheng Yao, Ruoqi Li, Jing Zhang, Jun Sun, and Chongyang Zhang. Explicit boundary guided semi-push-pull contrastive learning for supervised anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24490–24499, 2023. 3, 6

[59] Jihun Yi and Sungroh Yoon. Patch svdd: Patch-level svdd for anomaly detection and segmentation. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 1, 2

[60] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019. 3, 6

[61] Muhammad Zaigham Zaheer, Jin-ha Lee, Marcella Astrid, and Seung-Ik Lee. Old is gold: Redefining the adversarially learned one-class classifier training paradigm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14183–14193, 2020. 1, 2

[62] M Zaigham Zaheer, Arif Mahmood, M Haris Khan, Mattia Segu, Fisher Yu, and Seung-Ik Lee. Generative cooperative learning for unsupervised video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14744–14754, 2022. 2, 3

[63] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Draem-a discriminatively trained reconstruction embedding for surface anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8330–8339, 2021. 1, 2, 3, 6

[64] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Reconstruction by inpainting for visual anomaly detection. *Pattern Recognition*, 112:107706, 2021. 1, 2

[65] Hui Zhang, Zuxuan Wu, Zheng Wang, Zhineng Chen, and Yu-Gang Jiang. Prototypical residual networks for anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16281–16291, 2023. 3

[66] Xuan Zhang, Shiyu Li, Xi Li, Ping Huang, Jiulong Shan, and Ting Chen. Destseg: Segmentation guided denoising student-teacher for anomaly detection, 2023. 1, 2

[67] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: Short papers)*, pages 207–212, 2016. 6

[68] Yuansheng Zhu, Wentao Bao, and Qi Yu. Towards open set video anomaly detection. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIV*, pages 395–412. Springer, 2022. 1, 2, 3