Iterative Deployment Exposure for Unsupervised Out-of-Distribution Detection

Lars Doorenbos[1,2][0000−0002−0231−9950], Raphael Sznitman[1][0000−0001−6791−4753], and Pablo Márquez-Neila[1][0000−0001−5722−7618]

[1] University of Bern, Switzerland
[2] University of Bonn, Germany
doorenbos@iai.uni-bonn.de

Abstract. Deep learning models are vulnerable to performance degradation when encountering out-of-distribution (OOD) images, potentially leading to misdiagnoses and compromised patient care. These shortcomings have led to great interest in the field of OOD detection. Existing unsupervised OOD (U-OOD) detection methods typically assume that OOD samples originate from an unconcentrated distribution complementary to the training distribution, neglecting the reality that deployed models passively accumulate task-specific OOD samples over time. To better reflect this real-world scenario, we introduce Iterative Deployment Exposure (IDE), a novel and more realistic setting for U-OOD detection. We propose CSO, a method for IDE that starts from a U-OOD detector that is agnostic to the OOD distribution and slowly refines it during deployment using observed unlabeled data. CSO uses a new U-OOD scoring function that combines the Mahalanobis distance with a nearest-neighbor approach, along with a novel confidence-scaled few-shot OOD detector to effectively learn from limited OOD examples. We validate our approach on a dedicated benchmark, showing that our method greatly improves upon strong baselines on three medical imaging modalities.

Keywords: Out-of-Distribution Detection · Deployment · Reliability.

1 Introduction

Deep learning (DL) models fundamentally rely on the premise that the training data distribution aligns with that of the test data. However, this assumption often fails in real-world situations where the performance of a DL model diverges from its initial benchmark due to encounters with out-of-distribution (OOD) samples. This phenomenon is highly problematic in medical imaging, where the dependability of DL models is critical for safe, prolonged use in the field.

These shortcomings have sparked great interest in the field of OOD detection [7, 27, 29], and unsupervised OOD (U-OOD) detection in particular. Unlike supervised OOD detection, U-OOD assumes neither access to training labels nor OOD samples, thereby encompassing a more generally applicable albeit more challenging setting [2, 5, 6, 12, 14, 15]. The core principle of U-OOD is to identify

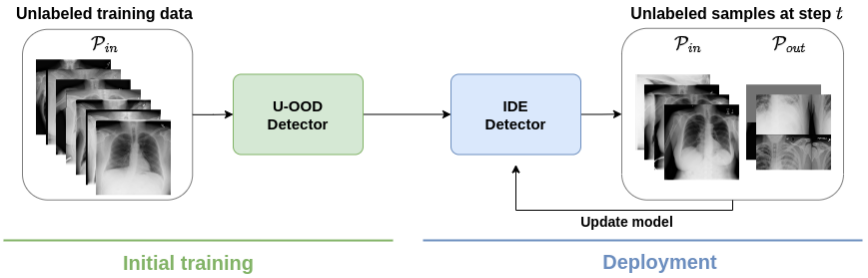


Fig. 1. Iterative deployment exposure. The initial detector, trained on unlabeled ID data, is iteratively refined with unlabeled deployment samples.

the level sets of the in-distribution (ID) training data and establish a threshold to distinguish OOD samples, where the assumption is that OOD samples are the complement of the ID and therefore unconcentrated [21, 23].

However, this assumption fails to consider the underlying motivation of OOD detection, that is, improving the reliability of *deployed* downstream models. Treating OOD data as arising from a generic, unconcentrated distribution is a disconnect from the reality that deployment environments inherently impose a specific, concentrated OOD context. We argue that the OOD context must be inferred from the deployment setting, and, in turn, the deployment context fully determines a *concentrated* OOD distribution. As this context depends on the deployment, U-OOD works that synthesize anomalies *a priori* (e.g., [20, 25]) to simulate the OOD distribution are inadequate. Instead, an approach where an initial U-OOD detector, agnostic to the OOD distribution, is gradually updated during deployment to consider the actual OOD distribution is needed (Fig. 1).

Yet, current U-OOD research has mainly neglected this important consideration. While related fields like OOD detection with in-the-wild data [3, 9] and OOD test-time adaptation [4, 28] exist, they are predicated on the availability of a large number of test samples: the former relies on a large set of unlabeled data consisting of both ID and OOD samples, while the latter uses the entire unlabeled test set to update a model. In both cases, hundreds of OOD samples are used to update their respective models, and performance is only measured after seeing all of them. Consequently, this is often unrealistic, considering that OOD samples can be scarce in medical applications.

Instead, we focus on improving detection with only a few OOD samples and tackle the realistic setting where this process is iterative: detect OOD samples, update the OOD detector, and repeat the procedure with a refined model several times. Consequently, evaluating U-OOD detectors should extend beyond a single instance and consider their effectiveness *over time*. We refer to this setting as *IDE* (Iterative Deployment Exposure). Hence, in this work, we,

1. introduce the problem of IDE for U-OOD detection, which closely matches the reality of model deployment.

2. introduce new metrics and benchmarks to evaluate methods in the IDE setting.
3. propose CSO, a novel method for IDE that outperforms strong baselines from related fields.

2 Iterative Deployment Exposure

In U-OOD detection, the training distribution of the downstream model is called the *in distribution* $\mathcal{P}_{\text{in}}$, and the *out distribution* $\mathcal{P}_{\text{out}}$ is assumed to be its *complement*. Given its large support, producing a set of OOD samples representative of $\mathcal{P}_{\text{out}}$ for supervised classification of ID vs. OOD samples is intractable. Instead, U-OOD detection methods rely on in-distribution (ID) samples to train a detector $\sigma^{\text{in}}\colon \mathcal{X} \to \mathbb{R}$ that scores the *OOD-ness* of test samples at inference time. Critically, the likelihood of observing a specific image from $\mathcal{P}_{\text{out}}$ is not uniform once a downstream task is established and the model is deployed in a given environment. Instead, OOD samples seen during the operation of the deployed system constrain $\mathcal{P}_{\text{out}}$ to the specific application. The goal of IDE is to progressively enrich the detector σ^{in} with observed OOD samples, thus adapting the detection model to the deployment environment.

More specifically, an IDE system builds a sequence of detectors $s_t(\mathbf{x})\colon \mathcal{X} \mapsto \mathbb{R}$ for time steps $t \in \{0, 1, 2, \ldots\}$. The sequence starts with the base U-OOD detector, $s_0 = \sigma^{\text{in}}$, and progressively adapts to the OOD samples observed after deployment. At each time t, the detector s_t is trained with the dataset of samples observed until time step t, denoted $\mathcal{D}_t = \mathcal{D}^{\text{train}} \cup \mathcal{D}_t^{\text{deploy}}$. We assume that $\mathcal{D}_0^{\text{deploy}} = \varnothing$. Crucially, not all elements in $\mathcal{D}_t$ are labeled as ID or OOD. While elements of $\mathcal{D}^{\text{train}}$ are known to come from distribution $\mathcal{P}_{\text{in}}$ and are, therefore, ID samples, elements of $\mathcal{D}_t^{\text{deploy}}$ are unlabeled. These samples of the *deployment distribution* $\mathcal{P}_{\text{deploy}}$, following [3, 9], are modeled with the Huber contamination model [8],

$$\mathcal{P}_{\text{deploy}} = (1 - \pi)\mathcal{P}_{\text{in}} + \pi\mathcal{P}_{\text{out}}, \tag{1}$$

with contamination ratio π.

To address the lack of labels in $\mathcal{D}_t^{\text{deploy}}$, s_{t-1} is used to pseudo-label the elements of $\mathcal{D}_t^{\text{deploy}}$ for training s_t. Hence, at each time t, $\mathcal{D}_t$ can be split into two disjoint subsets $\mathcal{D}_t^{\text{in}}$ and $\mathcal{D}_t^{\text{out}}$ containing the samples labeled as ID and OOD, respectively. For simplicity and without loss of generality, we assume that the deployment dataset $\mathcal{D}_t^{\text{deploy}}$ grows K elements at each time step, whereby $\left|\mathcal{D}_t^{\text{deploy}}\right| = t \cdot K$. We will also omit the index t where not explicitly needed.

2.1 Model

The main challenge of OOD detection is the scarcity of representative OOD samples. In IDE, the expected number of OOD samples at time step t is $\pi \cdot t \cdot K$, which for a small t is too small to effectively train a binary classifier, preventing

us from simply using binary classifiers to model the detector s_t. As t increases, however, the feasibility of training a binary classifier improves. We, therefore, design our detection model s_t to behave as a few-shot learner s^- when t is close to 0 and to gradually transition towards a strong binary learner s^+ as t increases. Formally, we model the detector s_t as a convex combination of two learner modalities controlled by a mixing factor α_t,

$$s_t(\mathbf{x}) = (1 - \alpha_t)s_t^-(\mathbf{x}) + \alpha_t s_t^+(\mathbf{x}). \tag{2}$$

The key difference between the learners s^- and s^+ lies in their inductive biases. The few-shot learner s^- is a low-variance/high-bias classifier with strong assumptions about in- and out-distributions. Its design is based on the U-OOD detector σ^{in}, as detailed below. In contrast, the strong learner s^+ is a low-bias binary classifier, and its architecture can be chosen according to the nature of the input space $\mathcal{X}$ (e.g., a CNN or a transformer architecture for image data), as our approach is agnostic to the internal specifics of s^+. At each step t, both s_t^- and s_t^+ are trained independently with the dataset $\mathcal{D}_t$ pseudo-labeled with s_{t-1}.

The factor $\alpha_t \in [0, 1]$ controls the transition between both models and is proportional to the number of elements pseudo-labeled as OOD,

$$\alpha_t = \min\left(1, \beta \cdot \left|\mathcal{D}_t^{\text{out}}\right|\right), \tag{3}$$

where the factor β is a hyperparameter of our method. We refer to our method as CSO (confidence-scaled U-OOD detector). The next sections describe our U-OOD detector σ^{in} and how it is used to define the few-shot learner s^-.

U-OOD detector: Our U-OOD detector combines elements from the Mahalanobis anomaly detector (MahaAD) [19], known for its robustness and speed [2], and from non-parametric nearest-neighbor scoring methods [1, 17, 24]. As in MahaAD, given the collection of ID samples $\mathcal{D}^{\text{in}} = \{\mathbf{x}_i\}_{i=1}^N$, we fit a Gaussian distribution parameterized by the data mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. To prevent numerical problems with near-singular covariance matrices in high-dimensional or low-data regimes, we use shrinkage following the standard hyperparameter-free method of [11]. MahaAD uses the induced Mahalanobis distance $d_{\boldsymbol{\Sigma}'}(\mathbf{x}, \boldsymbol{\mu})$ between $\boldsymbol{\mu}$ and a test sample $\mathbf{x}$ to estimate its OOD score. We instead use this distance to perform a 2-NN search (following, e.g., [1]) and score test samples with the average distance to their 2 nearest neighbors,

$$\sigma^{\text{in}}(\mathbf{x}) = \frac{1}{2} \sum_{\mathbf{x}' \in N^2_{\boldsymbol{\Sigma}'}(\mathbf{x})} d_{\boldsymbol{\Sigma}'}(\mathbf{x}, \mathbf{x}'), \tag{4}$$

where $N^2_{\boldsymbol{\Sigma}'}(\mathbf{x})$ denotes the 2-nearest neighbors of $\mathbf{x}$ in the training data measured with the Mahalanobis distance induced by $\boldsymbol{\Sigma}'$. We refer to this as MkNN.

The detector σ^{in} defined in Eq. (4) is inappropriate for image samples, as the Mahalanobis distance is not a reliable measure of image similarity on high-dimensional spaces. Instead, when dealing with images, we first describe each

image $\mathbf{x}_i$ with a sequence of feature vectors $\{\mathbf{f}_\ell(\mathbf{x}_i)\}_{\ell=1}^L$, where $\mathbf{f}_\ell$ denotes the result of applying global average pooling on the feature map of the ℓ-th layer of a pre-trained convolutional neural network $\mathbf{f}$. We then use the descriptors of the training images to build a collection of layer-wise detectors $\{\sigma_\ell^{\text{in}}\}_{\ell=1}^L$. In particular, the detector σ_ℓ^{in} at layer ℓ applies the Eq. (4) with the covariance matrix $\mathbf{\Sigma}'_\ell$ computed from the collection of features $\{\mathbf{f}_\ell(\mathbf{x}_i)\}_{i=1}^N$. The final OOD score for a test image $\mathbf{x}$ is the sum of the scores over all layers,

$$\sigma^{\text{in}}(\mathbf{x}) = \sum_{\ell=1}^{L} \sigma_\ell^{\text{in}}(\mathbf{f}_\ell(\mathbf{x})). \tag{5}$$

Few-shot learner s^-: Our few-shot learner extends the OOD detector σ^{in} trained with $\mathcal{D}^{\text{in}}$ by incorporating a twin detector σ^{out} trained with $\mathcal{D}^{\text{out}}$. The OOD score of the few-shot learner is computed as the difference between both detectors,

$$s^-(\mathbf{x}) = \sigma^{\text{in}}(\mathbf{x}) - \lambda\sigma^{\text{out}}(\mathbf{x}), \tag{6}$$

where the factor λ controls the influence of σ^{out} in the final score. The value of λ depends on the confidence levels of the detectors, which, in turn, rely on the contents in $\mathcal{D}^{\text{in}}$ and $\mathcal{D}^{\text{out}}$.

To measure the confidence of a detector σ trained with the dataset $\mathcal{D}$, we assess its variability under bootstrapping. More specifically, we produce M bootstrap samples $\{\mathcal{D}^{(m)}\}_{m=1}^M$ of N elements randomly sampled from $\mathcal{D}$ with replacement, and compute the covariance matrices $\mathbf{\Sigma}^{(m)}$ for each sample $\mathcal{D}^{(m)}$. The uncertainty of the detector is measured as the variability of the bootstrapped covariance matrices,

$$U(\mathcal{D}) = \frac{1}{Md^2} \sum_{m=1}^{M} \left\| \mathbf{\Sigma}^{(m)} - \bar{\mathbf{\Sigma}} \right\|_F^2, \tag{7}$$

where $\bar{\mathbf{\Sigma}} = \frac{1}{M}\sum_m \mathbf{\Sigma}^{(m)}$. The factor λ is then computed as the ratio between the uncertainties of the detectors,

$$\lambda = \min\left(1, \gamma \frac{U(\mathcal{D}^{\text{in}})}{U(\mathcal{D}^{\text{out}})}\right), \tag{8}$$

where $\gamma > 0$ is a hyperparameter of our method. If no OOD samples are available, $U(\mathcal{D}^{out}) \to \infty$ and $\lambda = 0$, thus making the few-shot learner s^- equivalent to the base OOD detector σ^{in}.

When working with image data, we proceed layer by layer, as previously discussed for the U-OOD detector. In particular, we build a few-shot learner s_ℓ^- per each layer ℓ of the feature extractor $\mathbf{f}$, and the final score is the sum of the layer-wise scores,

$$s^-(\mathbf{x}) = \sum_{\ell=1}^{L} s_\ell^-(\mathbf{f}_\ell(\mathbf{x})). \tag{9}$$

3 Experiments

3.1 Experimental set-up

Datasets: We introduce three IDE benchmarks to compare methods comprising various modalities, contamination ratios, and other settings. The samples for evaluating the methods and those in $\mathcal{D}_t^{\text{deploy}}$ do not overlap.

1. NIH [26]: Training: 4'261 healthy chest X-rays. Testing: 250 healthy scans as ID and 250 pathological chest X-rays as OOD. All methods see $T = 10$ steps of $K = 50$ test samples, with contamination $\pi = 0.2$.
2. MURA [16]: Training: 5'106 musculoskeletal radiographs of fingers. Testing: 250 finger scans as ID with scans of elbows, forearms, hands, humeri, shoulders, and wrists as OOD, where $T = 5$, $K = 100$, and $\pi = 0.1$.
3. DRD [6]: Training: 25'809 healthy retinal fundus photographs as ID. Testing: 250 healthy scans as ID, with 250 scans of strongest level of diabetic retinopathy as OOD, where $T = 5$, $K = 50$, and $\pi = 0.1$.

Baselines: We compare our method to seven baselines that comprise the top-performing methods from related fields: AdaODD [28], ETLT [4], SAL [3], BCE [13], the Mahalanobis difference (MDiff) [22], and HSC [13]. Furthermore, we include MahaAD [19] as a U-OOD baseline. All methods use the same ImageNet pre-trained backbone. To ensure fairness, all methods use the same grid search procedure to set their hyperparameters. We measure the performance of each method on the CIFAR10 experiment `Plane:Rest` and select the highest-performing configuration over the grid.

Evaluation metrics: In contrast with previous works, we are interested in measuring the quality of the OOD detector over time. As such, we propose two metrics that consider this aspect: the Area Under the FPR@95 curve (AUF) and the Area Under the AUC curve (AUA). These are computed by first evaluating the FPR@95 and AUC at every timestep and plotting the resulting FPR@95/AUC curves with respect to t. Then, the AUF and AUA are given by the area under the FPR@95 and AUC curves, respectively, normalized by the time elapsed.

Implementation details: For s^-, we extract features from $L = 4$ layers at the end of every ResNet-18 block and normalize the features of the final layer following [4, 18]. The binary classifier s^+ also uses a ResNet-18 architecture. It is trained with Adam [10] using a learning rate of 10^{-5} and batch size 256 for ten epochs on NIH and an equivalent number of iterations on the other datasets. We apply data augmentation in the form of random resized crops, color jitter, and horizontal flips and initialize the binary classifier with the weights from the previous timestep. At every step, the elements of $\mathcal{D}_t^{\text{deploy}}$ are re-labeled with the current model $s_t(\mathbf{x})$ using a threshold. The threshold is determined on the training data such that 95% is considered in-distribution. We standardize the

Table 1. Comparative evaluation. We report the mean of the AUF (↓) and AUA (↑) over five trials. **Bold** and underlined indicate best and second best, respectively. Our method obtains the best performance overall.

	NIH		MURA		DRD		Mean	
	AUF	AUA	AUF	AUA	AUF	AUA	AUF	AUA
SAL	91.5$_{\pm 6.3}$	58.3$_{\pm 12.1}$	90.0$_{\pm 8.3}$	56.6$_{\pm 9.2}$	77.4$_{\pm 8.9}$	67.3$_{\pm 7.9}$	86.3	60.7
HSC	70.4$_{\pm 0.9}$	78.6$_{\pm 0.2}$	68.8$_{\pm 6.5}$	79.8$_{\pm 2.3}$	81.8$_{\pm 2.0}$	71.6$_{\pm 1.3}$	73.7	76.7
ETLT	54.9$_{\pm 4.0}$	82.0$_{\pm 1.0}$	61.5$_{\pm 4.9}$	86.5$_{\pm 0.7}$	64.8$_{\pm 2.0}$	78.3$_{\pm 1.1}$	60.4	82.3
BCE	60.1$_{\pm 16.7}$	74.0$_{\pm 9.6}$	56.8$_{\pm 8.2}$	84.3$_{\pm 3.1}$	60.5$_{\pm 16.9}$	80.3$_{\pm 7.3}$	59.1	79.5
MahaAD	53.8$_{\pm 3.9}$	85.3$_{\pm 0.9}$	47.0$_{\pm 4.5}$	90.8$_{\pm 0.7}$	71.4$_{\pm 3.9}$	78.1$_{\pm 1.0}$	57.4	84.7
AdaODD	46.9$_{\pm 5.5}$	87.0$_{\pm 0.9}$	48.3$_{\pm 7.9}$	88.6$_{\pm 1.3}$	**57.8**$_{\pm 4.4}$	**83.5**$_{\pm 1.1}$	51.0	86.4
MDiff	48.1$_{\pm 4.4}$	83.0$_{\pm 2.1}$	42.9$_{\pm 6.6}$	91.3$_{\pm 1.1}$	60.3$_{\pm 4.6}$	82.7$_{\pm 3.1}$	50.4	85.7
CSO (ours)	**34.9**$_{\pm 6.1}$	**90.9**$_{\pm 1.7}$	**37.5**$_{\pm 5.6}$	**91.8**$_{\pm 0.9}$	61.9$_{\pm 5.6}$	83.1$_{\pm 1.6}$	**44.8**	**88.6**

scores of s^+ and s^- before combining them to ensure they have similar scales. The hyperparameters β and γ were set via the grid search to $1/300$ and 3, respectively. We found CSO robust to their settings, as shown in the next section.

3.2 Results

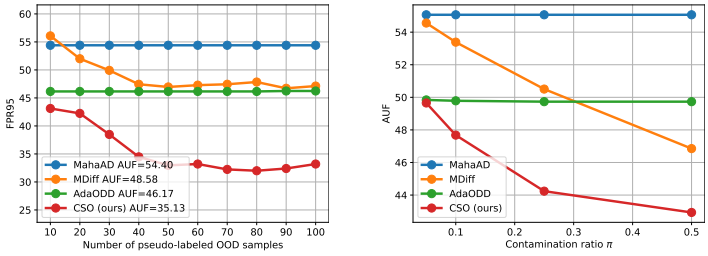
Tab. 1 reports the results on our benchmark. Of all methods, SAL, HSC, ETLT, and BCE do not outperform the unsupervised MahaAD baseline on average. BCE is especially inconsistent; for instance, it achieves excellent results on DRD but performs below average in other cases, demonstrating the need for a more robust method. In contrast, AdaODD and MDiff score consistently high in all experiments. Nonetheless, they are outclassed by CSO, which reaches the best score for NIH and MURA and the best overall performance by 5.6 AUF and 2.2 AUA compared to the next-best method.

The performance evolution over time for some of the best methods on NIH is shown in Fig. 2(a). All methods benefit from incorporating unlabeled samples, with our method improving fastest. The only exception is AdaODD, whose optimal hyperparameters from the grid search assign low importance to the unlabeled NIH data.

3.3 Ablations

Hyperparameter sensitivity From the grid search conducted on `Plane:Rest`, we find CSO to be robust to its main hyperparameters β and γ: all combinations with $\gamma \in [1, 3, 5]$ and $\beta \in [1/100, 1/300, 1/500]$ achieve between 92.6 and 93.2 AUA.

Contamination ratio We probe the effectiveness of CSO compared to the three best baselines under varying contamination ratios π while keeping the number



(a) (b)

Fig. 2. Ablation curves. In (a), we show how the FPR@95 evolves over time for the best methods on NIH. Our method achieves the best results, already after one iteration. In (b), we compare methods by AUF under varying contamination ratios on NIH. Our method consistently outperforms the baselines at different contamination levels.

Table 2. Ablation study on CIFAR10. (left) U-OOD performance in AUC over one run as the methods are deterministic. MkNN outperforms the other U-OOD scoring functions. (right) Comparing few-shot OOD performance. We report the mean AUC over five runs. n-shot refers to using n ground-truth OOD samples. The confidence scaling is important to achieve the best results.

	AUC
kNN	81.7
MahaAD	86.0
MkNN (ours)	**87.6**

	5-shot AUC	*10-shot AUC*
kNN	75.6	79.8
MDiff	84.7	89.9
s^-_{Maha} (ours)	91.2	91.9
s^-_{MkNN} (ours)	**91.8**	**92.7**

of OOD samples per step fixed at ten in Fig. 2(b). As expected, all iterative methods benefit from having a higher fraction of unlabeled OOD samples in the test set. Nonetheless, CSO achieves the best AUF for all contamination levels.

Scoring function We ablate our design choices by showing that (1) MkNN outperforms both kNN and MahaAD for U-OOD detection and (2) s^- outperforms unscaled scoring functions on few-shot OOD. To do so, we run experiments with our method on the standard one-class CIFAR10 benchmark.

From Tab. 2(left), MkNN outperforms the kNN scoring by 5.9 AUC and MahaAD by 1.6 AUC, showcasing its practical usefulness over top-performing U-OOD detectors [2]. Tab. 2(right) shows that equipping MDiff with our confidence scaling, which we label as s^-_{Maha}, already improves few-shot results. We improve the results by a further 0.6 and 0.8 AUC using MkNN. These results on natural images also confirm the usefulness of our method beyond medical settings.

4 Conclusion

We introduced the setting of IDE for U-OOD detection, which reflects the iterative process of real-world model deployment, along with new metrics and benchmarks for the task. Furthermore, we presented CSO, a novel method for IDE that gradually transforms the base U-OOD detector into a binary classifier. In doing so, we additionally introduced an OOD scoring function that uses the Mahalanobis distance to compute a nearest-neighbors score, and a few-shot OOD detector that takes into account the confidence of the distributions involved. Extensive experiments showed that our simple approach outperforms methods from several related fields.

Acknowledgments. This work was funded by the Swiss National Science Foundation (SNSF), research grant 200021_192285 "Image data validation for AI systems".

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Bergman, L., Cohen, N., Hoshen, Y.: Deep nearest neighbor anomaly detection. arXiv preprint arXiv:2002.10445 (2020)
2. Doorenbos, L., Sznitman, R., Márquez-Neila, P.: Data invariants to understand unsupervised out-of-distribution detection. In: European Conference on Computer Vision. pp. 133–150. Springer (2022)
3. Du, X., Fang, Z., Diakonikolas, I., Li, Y.: How does unlabeled data provably help out-of-distribution detection? International Conference on Learning Representations (2024)
4. Fan, K., Wang, Y., Yu, Q., Li, D., Fu, Y.: A simple test-time method for out-of-distribution detection. arXiv preprint arXiv:2207.08210 (2022)
5. González, C., Gotkowski, K., Fuchs, M., Bucher, A., Dadras, A., Fischbach, R., Kaltenborn, I.J., Mukhopadhyay, A.: Distance-based detection of out-of-distribution silent failures for covid-19 lung lesion segmentation. Medical image analysis **82**, 102596 (2022)
6. Graham, B.: Kaggle diabetic retinopathy detection competition report. University of Warwick **22** (2015)
7. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. International Conference on Learning Representations (2017)
8. Huber, P.J.: Robust estimation of a location parameter. In: Breakthroughs in statistics: Methodology and distribution, pp. 492–518. Springer (1992)
9. Katz-Samuels, J., Nakhleh, J.B., Nowak, R., Li, Y.: Training ood detectors in their natural habitats. In: International Conference on Machine Learning. pp. 10848–10865. PMLR (2022)
10. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. International Conference on Learning Representations (2015)
11. Ledoit, O., Wolf, M.: A well-conditioned estimator for large-dimensional covariance matrices. Journal of multivariate analysis **88**(2), 365–411 (2004)

12. Linmans, J., Raya, G., van der Laak, J., Litjens, G.: Diffusion models for out-of-distribution detection in digital pathology. Medical Image Analysis **93**, 103088 (2024)
13. Liznerski, P., Ruff, L., Vandermeulen, R.A., Franks, B.J., Müller, K.R., Kloft, M.: Exposing outlier exposure: What can be learned from few, one, and zero outlier images. Transactions on Machine Learning Research (2022)
14. Márquez-Neila, P., Sznitman, R.: Image data validation for medical systems. In: Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part IV 22. pp. 329–337. Springer (2019)
15. Naval Marimont, S., Tarroni, G.: Implicit field learning for unsupervised anomaly detection in medical images. In: Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part II 24. pp. 189–198. Springer (2021)
16. Rajpurkar, P., Irvin, J., Bagul, A., Ding, D., Duan, T., Mehta, H., Yang, B., Zhu, K., Laird, D., Ball, R.L., et al.: Mura: Large dataset for abnormality detection in musculoskeletal radiographs. arXiv preprint arXiv:1712.06957 (2017)
17. Reiss, T., Cohen, N., Bergman, L., Hoshen, Y.: Panda: Adapting pretrained features for anomaly detection and segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2806–2814 (2021)
18. Reiss, T., Hoshen, Y.: Mean-shifted contrastive loss for anomaly detection. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 2155–2162 (2023)
19. Rippel, O., Mertens, P., Merhof, D.: Modeling the distribution of normal data in pre-trained deep features for anomaly detection. In: 2020 25th International Conference on Pattern Recognition (ICPR). pp. 6726–6733. IEEE (2021)
20. Schlüter, H.M., Tan, J., Hou, B., Kainz, B.: Natural synthetic anomalies for self-supervised anomaly detection and localization. In: European Conference on Computer Vision. pp. 474–489. Springer (2022)
21. Schölkopf, B., Smola, A.J.: Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT press (2002)
22. Sehwag, V., Chiang, M., Mittal, P.: Ssd: A unified framework for self-supervised outlier detection. International Conference on Learning Representations (2021)
23. Steinwart, I., Hush, D., Scovel, C.: A classification framework for anomaly detection. Journal of Machine Learning Research **6**(2) (2005)
24. Sun, Y., Ming, Y., Zhu, X., Li, Y.: Out-of-distribution detection with deep nearest neighbors. In: International Conference on Machine Learning. pp. 20827–20840. PMLR (2022)
25. Tan, J., Hou, B., Day, T., Simpson, J., Rueckert, D., Kainz, B.: Detecting outliers with poisson image interpolation. In: Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part V 24. pp. 581–591. Springer (2021)
26. Tang, Y.X., Tang, Y.B., Peng, Y., Yan, K., Bagheri, M., Redd, B.A., Brandon, C.J., Lu, Z., Han, M., Xiao, J., et al.: Automated abnormality classification of chest radiographs using deep convolutional neural networks. NPJ digital medicine **3**(1), 70 (2020)
27. Zhang, O., Delbrouck, J.B., Rubin, D.L.: Out of distribution detection for medical images. In: Uncertainty for Safe Utilization of Machine Learning in Medical

Imaging, and Perinatal Imaging, Placental and Preterm Image Analysis: 3rd International Workshop, UNSURE 2021, and 6th International Workshop, PIPPI 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, October 1, 2021, Proceedings 3. pp. 102–111. Springer (2021)
28. Zhang, Y., Wang, X., Zhou, T., Yuan, K., Zhang, Z., Wang, L., Jin, R., Tan, T.: Model-free test time adaptation for out-of-distribution detection. arXiv preprint arXiv:2311.16420 (2023)
29. Zimmerer, D., Full, P.M., Isensee, F., Jäger, P., Adler, T., Petersen, J., Köhler, G., Ross, T., Reinke, A., Kascenas, A., et al.: Mood 2020: A public benchmark for out-of-distribution detection and localization on medical images. IEEE transactions on medical imaging **41**(10), 2728–2738 (2022)