VUOD: A Versatile Unsupervised Outlier Detection Framework for Natural, Industrial, Medical Images and Beyond

Zhonghang Liu[1], Siyuan Chen[1], Jingwen Yu[2], Changshuo Wang[3], Kunyang Li[4‡], and Jiangbo Lu[1✉]

[1] SmartMore Corporation, China
[2] The Hong Kong University of Science and Technology, China
[3] University College London, UK
[4] Rice University, USA
jiangbo.lu@gmail.com

Abstract. Unsupervised outlier detection that automatically identifies whether visual systems involve anomalous images is a significant research topic. However, most approaches are limited to natural images because they rely on frozen and pre-trained feature extractors. So that their performance cannot be maintained in practical scenarios, especially for industrial inspection and medical imaging. In this work, we first revisit this task and then introduce a versatile unsupervised outlier detection framework to enrich the application domains. The core idea of this framework is to improve feature discriminativess via exploiting intrinsic distribution priors. Evaluated on 3 domains and 14 benchmark datasets, our proposed solution achieves state-of-the-art performance and significantly outperforms existing methods. More importantly, we show its plug-and-play property that can be integrated into diverse visual applications to improve their robustness, such as image classification and 3D reconstruction. Our code is available at https://github.com/zhliu-uod/VUOD.

Keywords: Unsupervised · Outlier Detection · Versatile Applications

1 Introduction

In the realm of image data, outliers are inevitable. For example, the out-of-distribution data points (e.g., trucks within the "car" category) and defective manufacturing products (e.g., "screw" with a scratch). Unsupervised outlier detection (UOD), aiming to automatically identify anomalous images, is a significant research topic. However, since recent methods [40, 46, 47] rely on natural image pretrained feature extractors [27, 57], their performance will be dramatically affected in various practical domains. In this work, we introduce a versatile unsupervised outlier detection (VUOD) framework, which enables UOD methods to be effective on scene recognition [75], industrial inspection [7, 8] and medical

‡ Work done during an internship at SmartMore, ✉ Corresponding author.

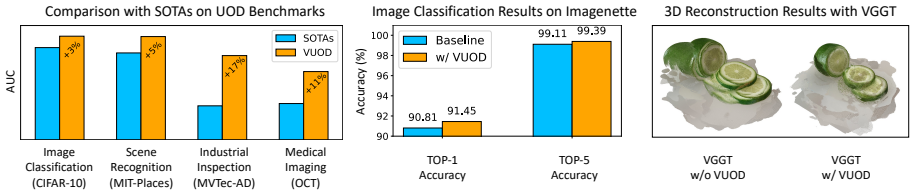

Fig. 1: The highlights of our proposed VUOD framework. Left plot: VUOD significantly outperforms existing SOTAs on UOD benchmark datasets and real-world applications. More importantly, VUOD can be seamlessly integrated into various downstream visual tasks, such as image classification (middle plot) and 3D reconstruction with VGGT [66] (right plot), to improve their performance and robustness.

imaging [21], also improving the robustness of image classification [16] and 3D reconstruction [66], as illustrated in Fig. 1.

To this end, the core idea is to improve the feature discriminativeness among different domains. From this perspective, we will mainly focus on exploiting reliable pseudo labels, i.e., self-supervision that transforms the current unsupervised task into a supervised problem, and leverage contrastive learning to enhance the separability of image feature representation. Previously, this was usually generated by threshold learners [40, 46] that learn a decision boundary on a series of outlier scores given by an external unsupervised outlier detector. However, this paradigm suffers from a significant inconsistency problem caused by the independence between threshold learners and outlier scoring methods.

In this study, we present a novel perspective by leveraging the prior knowledge of the target dataset without bringing in any external supervision signals. Two intrinsic priors, local separability and global separability, derived from our theoretical and empirical observations, are introduced. Specifically, in the local region, we utilize a fine-grained clustering approach to split the target set into a series of clusters, while in the global space, we explore a global separability prior that performs as a binary classifier to annotate each local cluster.

To evaluate the performance of our proposed solution, we subject it to rigorous and broad testing, which involves 3 domains (natural, industrial and medical) and 14 benchmark datasets. Our experimental results demonstrate that it achieves the overall state-of-the-art (SOTA) results and is also promising in terms of efficiency. The main contributions of this work can be summarized as:

- First, we comprehensively revisit and analyze the UOD task, including its definition and real-world applications. This could be beneficial for further fair evaluation and inspires future research perspectives.
- We propose a versatile UOD framework that offers several advantages:
  (i) Evaluated on standard UOD benchmarks, it achieves SOTA performance and significantly outperforms competitors on challenging datasets.
  (ii) Evaluated on practical scenarios, it significantly outperforms other UOD baselines, e.g., 16.7% improvement on MVTec-AD, and achieves competitive performance compared with specific defect detectors.

(iii) It helps improve the robustness of visual tasks like image classification and 3D reconstruction via its significant outlier filtering ability.

2 Related Work

2.1 Unsupervised Outlier Detection

UOD methods can be divided into two major categories: deep learning-based and statistical-based. Deep approaches [37, 38, 67, 76, 78] attempt to compress image samples into a low-dimensional space and reconstruct them. The underlying assumption is that the model will successfully learn the patterns of inliers, resulting in low reconstruction errors. In contrast, statistical-based methods utilize discrimination [40, 43, 46, 47] or density [35, 44] strategies to model the inlier manifold. However, most of them significantly rely on a natural image pretrained feature extractor, e.g., ResNet-50 [27], and CLIP [57], which limits their ability to generalize into diverse applications [3, 7, 8, 32, 51]. To address this limitation, we will focus on enhancing feature discriminativeness across varied domains.

2.2 Feature Discriminativeness Learning

Mainstream end-to-end self-supervised models [29, 67, 68] typically enhance feature discriminativeness through contrastive-based or pretext-based strategies. They often require carefully designed augmentations and domain-specific heuristics that may not align with the subtle nuances of outlier detection. In contrast, self-supervision generation (pseudo-labeling) in UOD is typically achieved through the thresholding (threshold learning) [1,2,4,4,9–11,23,26,30,33,50,54,55, 65, 74] process, which determines a decision boundary based on a series of given outlier scores. However, because conventional threshold learning is independent of the underlying outlier detector, and the discriminative quality of the resulting outlier scores is not guaranteed, i.e., threshold-based methods often perform sub-optimally in practice. To address these limitations, our approach predicts pseudo labels by leveraging intrinsic distribution priors within the feature space.

3 Revisiting Unsupervised Outlier Detection

Before proposing our methodology, we discuss the foundational concepts for the UOD task. We first formally define the problem, including its notations and the core objective. Following this, we explore how UOD can be applied across diverse real-world domains and integrated into broader visual perception applications.

3.1 Problem Definition

The "unsupervised" term in the outlier detection field mainly refers to no labeled data being involved in the training phase. In other words, UOD methods are trained and tested on the same *target dataset* [47], which involves n unlabeled

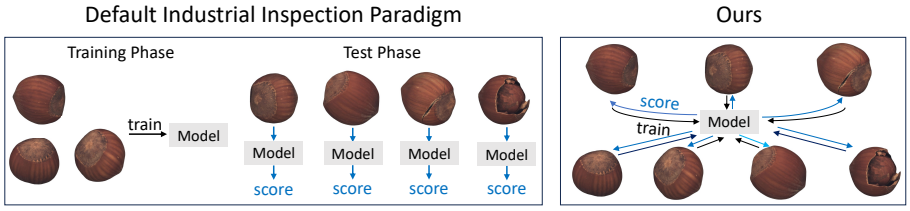

Fig. 2: Comparison between our paradigm and default industrial inspection. Left plot: The default unsupervised paradigm for industrial inspection, requiring a training phase on defect-free samples before sequential inference. Right plot: Our introduced paradigm eliminates the split between training and testing, enabling flexible, simultaneous detection of multiple objects within a single scene.

images $\mathbf{I} = \mathbf{I}_{in} \cup \mathbf{I}_{out}$, where $\mathbf{I}_{in}$, $\mathbf{I}_{out}$ refer to inliers and outliers, respectively. The outlier ratio (contamination factor) $\gamma \in (0, 1)$ is $\frac{\#\mathbf{I}_{out}}{\#\mathbf{I}_{in}+\#\mathbf{I}_{out}}$, which is also unknown. By default, we utilize some pretrained feature extractors, e.g., ResNet-50 [27] and CLIP [57], to obtain the target features $\mathbf{X} = \mathbf{X}_{in} \cup \mathbf{X}_{out} \in \mathbb{R}^{n \times d}$, where d refers to the dimension. Although UOD typically is a binary classification that includes both scoring (predicting outlier scores for each sample) and thresholding (deciding the boundary to predict labels), the thresholding phase will be an ill-posed problem without a separable outlier score distribution [46]. The core objective of UOD approaches is to learn an outlier score function $f(\cdot)$ to quantifies the anomalous probability of each $x_{i \in [1,n]} \in \mathbf{X}$, and the ideal $f(\cdot)$ should achieve a 100% ranking accuracy, i.e., $\max(\{f(x_i)|x_i \in \mathbf{X}_{in}\}) < \min(\{f(x_i)|x_i \in \mathbf{X}_{out}\})$.

3.2 Real-world Applications

To date, as previously discussed, most UOD methods have focused on detecting anomalous samples in natural image datasets, such as trucks that appear in the "car" class. However, UOD has significant value beyond this setting. It plays an important role in various real-world domains, including industrial inspection [7, 8, 32, 45, 51, 59, 79] and medical imaging [12]. Remarkably, the paradigm of applying UOD methods into industrial inspection (defect detection) is different from default approaches, as shown in Fig. 2, and our solution demonstrates more flexibility since its unsupervised property. Moreover, UOD can be integrated as a complementary component in broader visual perception pipelines. For example, in image classification tasks, UOD can be used to remove out-of-distribution samples from the training set, thereby improving model robustness. Similarly, for feed-forward 3D reconstruction pipelines [66] that rely on consistent scene or object imagery, UOD can help filter mismatched or corrupted inputs, leading to more accurate reconstruction results. This will be evaluated in Sec. 5.4.

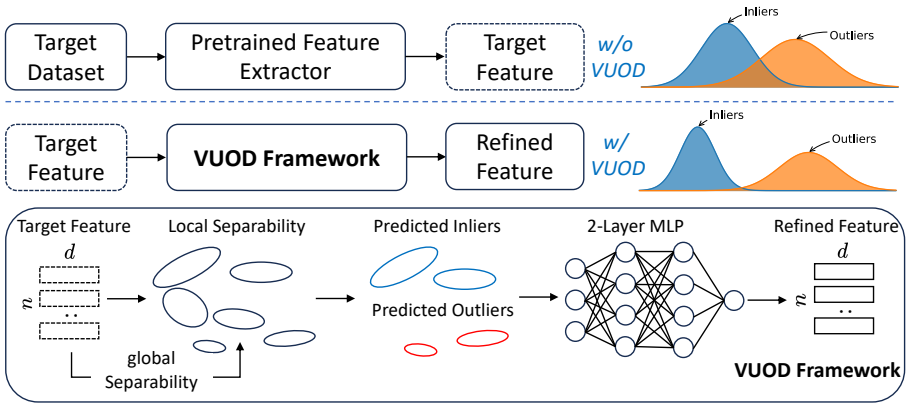

Fig. 3: The core idea of our proposed VUOD framework. An unlabeled target dataset is first processed by natural image pretrained feature extractors. We then investigate the intrinsic priors of the feature space (local and global separability), and generate pseudo labels. Based on predicted inliers and outliers, we apply contrastive learning to enhance the discriminativeness of the feature representations.

4 Method

4.1 Overview

To apply UOD approaches into diverse domains, the core idea is to enhance the discriminativess of feature representation via *pseudo labeling* (self-supervision generation), i.e., splitting both inlier features $\mathbf{X}'_{in}$ and outlier features $\mathbf{X}'_{out}$ from $\mathbf{X}$, and thereby leveraging *contrastive learning* strategy to enlarge the similarity within inliers and the difference between inliers and outliers in the enhanced feature space. The overview of our framework is illustrated in Fig. 3.

4.2 Separability Priors

We decompose pseudo labeling into splitting and classifying. Specifically, in the local region, we utilize a fine-grained clustering approach to split the target set into a series of clusters, while in the global space, we explore a separability prior that performs as a supervision signal to classify each local cluster.

Local Separability. In the first splitting phase, we adopt Affinity Propagation (AP) [63], which will split the target dataset's features $\mathbf{X}$ into a series of fine-grained clusters $\{\mathbf{C}_t\}_{t=1}^{T}$ without pre-defining the clustering number T, where $\mathbf{C}_t$ refers to t-th cluster. It proceeds by alternatively updating two matrices: responsibility and availability. The responsibility is defined as:

$$r(x_i, x_k) \leftarrow s(x_i, x_k) - \max_{k' \neq k} \{a(x_i, x_{k'}) + s(x_i, x_{k'})\}, \tag{1}$$

which quantifies how well-suited x_k is to serve as the exemplar (centroid) for x_i, relative to other candidate exemplars $x_{k'}$. Here $s(\cdot) = -||\cdot||_2^2$ refers to

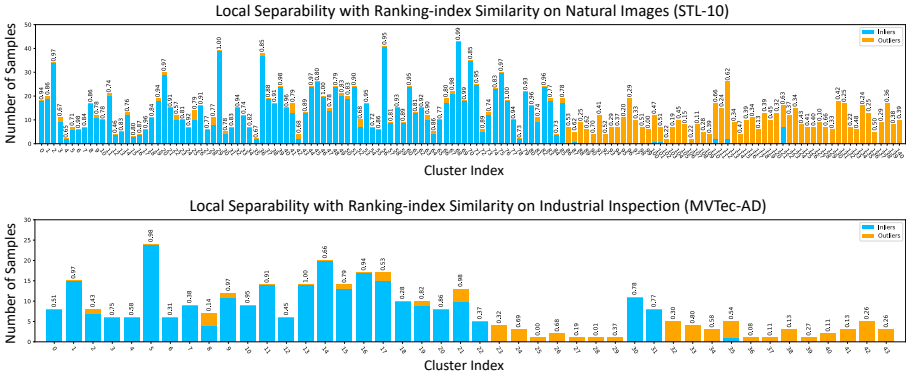

Fig. 4: The visualization of local separability with ranking-index similarity (Eq. 5). Intuitively, the fine-grained clustering algorithm Affinity Propagation [63] is capable of effectively separating outliers, even when the pretrained feature embeddings exhibit limited separability. In practice, uncontaminated inlier clusters typically show high ranking-index similarity (often larger than 0.9), while uncontaminated outlier clusters tend to exhibit substantially lower similarity values (often less than 0.5).

the similarity (negative squared euclidean distance), the availability $a(x_i, x_k) \leftarrow \min(0,\, r(x_k, x_k) + \sum_{i' \notin \{i,k\}} \max(0,\, r(x_{i'}, x_k)))$ represents how proper it would be for x_i to pick x_k as its exemplar, considering other points' preference for x_k as an exemplar. Based on some common sense in the UOD field [18, 19, 31, 64, 71], outliers are typically sparse, with large variance between each other and low similarity to inliers (normal samples). For example, if the inliers are good "capsules", outliers could be "capsules" with cracks, pokes, scratches, etc. [8]. Such that we could hypothesize that

$$\mathbb{E}_{x_i \in \mathbf{X}_{out}, x_j \in \mathbf{X}_{in}}[s(x_i, x_j)] \ll \mathbb{E}_{x_j, x_{j'} \in \mathbf{X}_{in}}[s(x_j, x_{j'})], \tag{2}$$

where $\mathbb{E}[\cdot]$ is the exception. In the responsibility $r(x_i, x_k)$ update step of AP, for an outlier $x_i \in \mathbf{X}_{out}$, there might exist no x_k (whether it is an inlier sample or another outlier) that can provide sufficiently high similarity. As a result, it may not be strongly attracted to any inlier cluster; or if there exists weak similarity among outliers, they may form a small cluster; or if completely isolated, it becomes a singleton cluster.

Consequently, all these clusters of AP can be divided into three categories: (i) inlier clusters that each cluster mostly contains inliers; (ii) outliers clusters that each cluster mostly contains outliers; (iii) mixed clusters that contain both inliers and outliers. The visualization can be found in Fig. 4, and more visualizations can be found in the Appendix. Next, we will explore a supervision signal to predict the labels of each cluster and split category (i) and (ii) from $\mathbf{X}$.

Discussion: Why not k-Means? In this work, we apply Affinity Propagation instead of the widely used k-Means [43, 48, 49, 77] to measure local separability. This is because k-Means implicitly assumes that the data are generated from several compact and spherical clusters, and it enforces each sample to be assigned

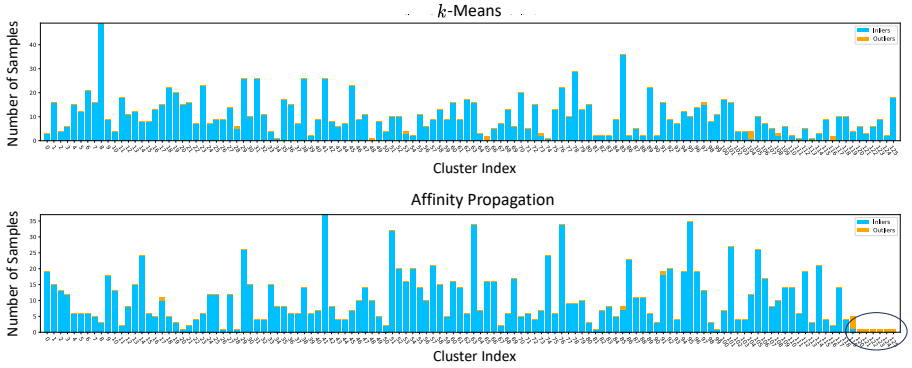

Fig. 5: The clustering comparison between Affinity Propagation and k-Means. Since k-Means requires a predefined number of clusters, which is impractical in unsupervised settings, we adopt the automatically determined number of clusters from Affinity Propagation for a fair comparison. As illustrated, Affinity Propagation consistently performs better than k-Means, particularly when outliers constitute only a small fraction of the target dataset.

to one of them. In other words, there is no mechanism to reject an assignment or leave a sample un-clustered. Consequently, outliers that even deviate significantly from the inlier distribution tend to be absorbed into the nearest cluster, which makes it difficult to separate them. The comparison is visualized in Fig. 5. Besides, one of the most challenging problems that utilizes k-Means is learning k in k-Means without any supervision, which is a challenging problem [43].

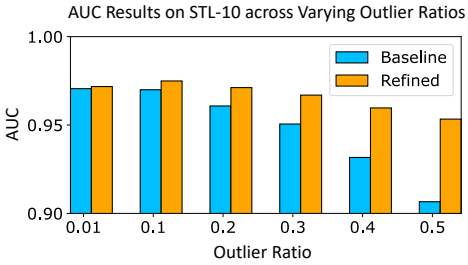

Fig. 6: The AUC results comparison between the refined distance (Eq. 6) and the baseline (Eq. 4). By removing the outlier impact from the mean computation, the performance of global separability can be maintained across different outlier ratios, i.e., the challenging mean-shift problem is well addressed.

Global Separability. Inspired by some theoretical analysis about the distribution of the target high-dimensional features [40–42], the distance from each sample to the centroid of inliers $m_{\mathbf{X}_{in}}$ could serve as a naturally discriminative outlier score function, i.e., for most $x_i \in \mathbf{X}_{in}$ and $x_j \in \mathbf{X}_{out}$, $||x_i - m_{\mathbf{X}_{in}}||2 <$

$||x_j - m_{\mathbf{X}_{in}}||2$. This is because inliers $\mathbf{X}_{in}$ are contained within a d-dimensional hypersphere of radius r, the volume is denoted as: $\mathbf{V}_d = (\pi^{d/2} r^d)/\Gamma(1 + d/2)$, where $\Gamma(\cdot)$ refers to the Gamma function. Shrinking r by a small value δ ($\delta > 0$) yields the following limit:

$$\lim_{d\to\infty} \frac{\mathbf{V}_d((1-\delta)r)}{\mathbf{V}_d} = \lim_{d\to\infty} (1-\delta)^d \leq \lim_{d\to\infty} e^{-\delta d} = 0. \tag{3}$$

As $d \to \infty$, almost all the volume concentrates on the hypersphere's surface. Given a low outlier ratio γ (e.g. $< 1\%$), the total feature mean $m_{\mathbf{X}}$ remains highly close to the inlier mean $m_{\mathbf{X}_{in}}$. Therefore,

$$f_{base}(x_i) = ||x_i - m_{\mathbf{X}}||^2 \tag{4}$$

provides reliable outlier scores. In contrast, when γ is relatively high, the gap between $m_{\mathbf{X}}$ and $m_{\mathbf{X}_{in}}$ becomes larger, which is considered a classic mean-shift problem [58]. To address this issue, we tend to remove some outliers based on an intuitive conclusion: $f(x_i) = ||x_i - m_{\mathbf{X}_{out}}||^2$ is an invalid outlier score function. Likewise, if some local $\mathbf{C}_t \in \mathbf{X}_{out}$ are obtained from the above local separability section, $f_t(x_i) = ||x_i - m_{\mathbf{C}_t}||^2$ is also invalid. This contrastive property inspires us to the ranking-index results between $\{f_t(x_i)\}_{i=1}^n$ and $\{f_{base}(x_i)\}_{i=1}^n$, which can be an indicator of the possibility of a cluster to be an outlier cluster [47]. Specifically, we adopt the Spearman's rank correlation coefficient [22] to compute the ranking-index similarity [47] between $\{f_t(x_i)\}_{i=1}^n$ and $\{f_{base}(x_i)\}_{i=1}^n$:

$$\phi_{t,base} = |\texttt{Spearman}(\text{rank-index}(\{f_t(x_i)\}_{i=1}^n), \text{rank-index}(\{f_{base}(x_i)\}_{i=1}^n)|, \tag{5}$$

where $|\cdot|$ represents the absolute value function. To align the value of $\phi_{t,base}$ across different domains, we apply the min-max normalization to it. If $\phi_{t,base}$ is less than a small positive value ϵ^+, we will remove $\mathbf{C}_t$ from $\mathbf{X}$, and the refined outlier score function is defined as:

$$f_{refine}(x_i) = ||x_i - m_{\mathbf{X}'_{in}}||^2, \mathbf{X}'_{in} = \{\mathbf{X} \setminus \mathbf{C}_t | \phi_{t,base} < \epsilon^+\}, \tag{6}$$

and ϵ^+ is fixed to 0.9 to obtain uncontaminated inlier features. As visualized in Fig. 6, the refined distance transform (Eq. 6) significantly outperforms the baseline (Eq. 4) across varying outlier ratios.

4.3 Implementation

Pseudo Labeling. Inspired by the above analysis, we can further compare each $f_t(x_i) = ||x_i - m_{\mathbf{C}_t}||^2$ with f_{refine}:

$$\phi_{t,refine} = |\texttt{Spearman}(\text{rank-index}(\{f_t(x_i)\}_{i=1}^n), \text{rank-index}(\{f_{refine}(x_i)\}_{i=1}^n)|. \tag{7}$$

If $\phi_{t,refine}$ is large, we consider the t-th cluster as an inlier cluster; if $\phi_{t,refine}$ is relatively small, we consider it an outlier cluster. Based on the basic definition of the UOD task, outliers may derive from various classes of data distribution

that are out of the normal range. Besides, although inliers usually come from one single class, we argue that it also demonstrates high within-class variation. For example, the "airplane" class not only involves standard airplanes but also has some wings, passengers, or cockpits. Such that in this work, we define two decision boundaries, ϵ^+ is set to 0.9 while ϵ^- is set to 0.5, respectively, to maintain the balance between sample quantity and purity of both inliers and outliers. The predicted inlier and outlier supervision in the second classifying phase is denoted as $\mathbf{X}'_{in} = \{\mathbf{C}_t | \phi_{t,refine} > \epsilon^+\}$ and $\mathbf{X}'_{out} = \{\mathbf{C}_t | \phi_{t,refine} < \epsilon^-\}$.

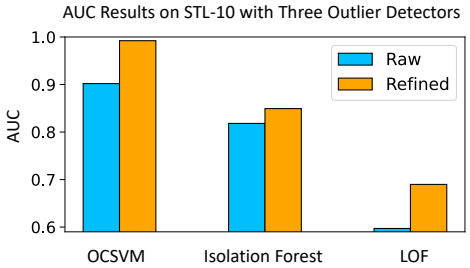

Fig. 7: The AUC results comparison between the refined feature and the raw feature with three outliers. The refined feature can significantly improve the detection accuracy of three outlier detectors.

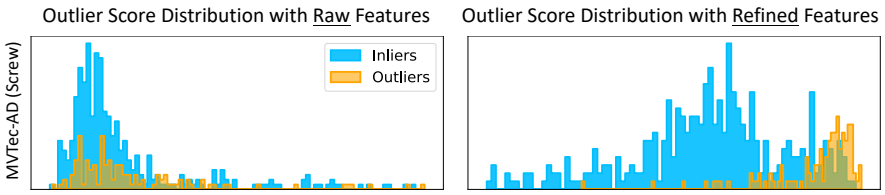

Fig. 8: The histogram visualization of outlier score distribution with both raw features and enhanced features. The enhanced features exhibit substantially improved separability between inliers and outliers compared with the raw features.

Contrastive Learning. Then, we are able to leverage the contrastive loss to enlarge the gap between inlier and outlier features as well as reduce the variation within inlier features. During each training epoch, we randomly sample positive pairs (x_i, x_j) from $\mathbf{X}'_{in}$ and negative pairs (x_i, x_j) from the combination of $\mathbf{X}'_{in}$ and $\mathbf{X}'_{out}$, corresponding to similar pairs with label $y = 0$ and dissimilar pairs with label $y = 1$, respectively. To avoid memory explosion and improve training stability, pairs are dynamically resampled at each epoch rather than generating all possible pairs at once. We define a simple two-layer perceptron (MLP) contrastive learning model $g_\theta : \mathbb{R}^d \to \mathbb{R}^{d'}$ consisting of two linear layers with ReLU

activation, followed by l_2 normalization of the output embedding:

$$g_\theta(x) = \texttt{normalize}\Big(\mathbf{W}_2, \mathrm{ReLU}(\mathbf{W}_1 x + b_1) + b_2\Big). \tag{8}$$

Given a sample pair (x_i, x_j) and its predicted label $y \in \{0, 1\}$, we adopt the following contrastive loss:

$$\begin{aligned}\mathcal{L}(x_i, x_j, y) = (1-y) \cdot ||g_\theta(x_i) - g_\theta(x_j)||_2^2 \\ + y \cdot \max\big(0, m - ||g_\theta(x_i) - g_\theta(x_j)||_2\big)^2,\end{aligned} \tag{9}$$

where m is a margin (1.0) hyperparameter that enforces a minimum distance between negative pairs. During training, the model performs forward propagation on the dynamically sampled positive and negative pairs to compute the loss $\mathcal{L}$, and the parameters $\theta \leftarrow \{\mathbf{W}_1, b_1, \mathbf{W}_2, b_2\}$ are updated via the Adam optimizer [34]. In inference, all feature samples $\mathbf{X}$ are passed through the trained model to obtain embeddings $\mathbf{Z} = g_\theta(\mathbf{X})$, which are then normalized. Consequently, we feed $\mathbf{Z}$ into the classic OCSVM [61] to predict the outlier scores $f(x_i) = \mathrm{OCSVM}(z_i \in \mathbf{Z}) \leftarrow g_\theta(x_i \in \mathbf{X})$. We select OCSVM as the outlier scoring head because OCSVM leverages the kernel trick to define complex, non-linear decision boundaries in high-dimensional feature spaces, making it more sensitive to subtle anomalousness. As validated in Fig. 7, our refined feature can significantly improve the detection accuracy (AUC score) of three classic outlier detectors, while OCSVM shows the best performance. As demonstrated in Fig. 8, for a hard case, the outlier score discriminativeness of our VUOD framework will be more separable than VUOD with the raw feature representation.

5 Evaluation

5.1 Experimental Settings

Benchmark Datasets. The evaluated benchmark datasets involve (i) Natural images: STL-10 [14], Internet [40], Caltech-101 [20], CIFAR-10 [36], CIFAR-100 [36], MIT-Places [75]; (ii) Industrial images: MVTec-AD [8], BTAD [51], MPDD [32] and MVTecLOCO [7]; (iii) Medical images: LiverCT, BrainMRI, OCT and RESC, from a medical imaging benchmark BMAD [3].
Building A Target Dataset. For natural image datasets, we adopt ResNet-50 [27] and CLIP [57] as two default feature extractors, and the target dataset is made up of all inliers from one class and some randomly selected outliers from other classes. There are six outlier ratios ranging from 0.01 to 0.5 [47]. For other benchmarks, we combine all samples from both training and test set and utilize ResNet-18 [27] and Wide ResNet-101 [73], followed by the existing work [5].
Competing Baselines. We mainly compare with a series of existing UOD methods: Deep SVDD [60], RSRAE [37], GOAD [6], Shell-Re [40], NeuTraL [56], ICL [62], LUNAR [37], ECOD [39], LVAD [43], SLAD [70], DeepIF [69], Multi-T [46] and FlexUOD [47]. And for the industrial MVTec-AD evaluation, we also

compare with some specific industrial defect detectors: PaDiM [15], FRE [52], and EfficientAD [5]. These approaches are also trained and tested on the same target dataset (involving both normal data and defective data).

Evaluation Metric. We mainly evaluate the ranking accuracy of the predicted outlier scores using the AUROC (AUC) score.

Table 1: AUC results on a series of natural image datasets. **Red** and **Blue** indicates the best and second-best results, respectively.

Method	Venue	STL-10	Internet	Caltech-101	CIFAR-10	CIFAR-100	MIT-Places
Deep SVDD [60]	ICML-2018	0.593	0.674	0.745	0.533	0.575	0.539
RSRAE [37]	ICLR-2019	0.903	0.916	**0.986**	0.816	0.889	0.778
GOAD [6]	ICLR-2020	0.946	0.945	0.981	0.862	0.886	0.871
Shell-Re [40]	TPAMI-2021	0.866	0.896	0.905	0.867	0.834	0.822
NeuTraL [56]	ICML-2021	0.828	0.877	0.633	0.748	0.815	0.711
ICL [62]	ICLR-2022	0.929	0.937	0.964	0.859	0.911	0.832
LUNAR [24]	AAAI-2022	0.781	0.776	0.923	0.766	0.859	0.698
ECOD [39]	TKDE-2022	0.932	0.925	0.976	0.888	0.905	0.846
LVAD [43]	ECCV-2022	0.948	0.923	0.977	0.864	0.907	0.860
SLAD [70]	ICML-2023	0.923	0.919	0.888	0.861	0.899	0.846
DeepIF [69]	TKDE-2023	0.858	0.866	0.874	0.801	0.808	0.765
Multi-T [46]	ECCV-2024	0.957	0.956	0.985	0.888	0.917	0.859
FlexUOD [47]	CVPR-2025	**0.980**	**0.981**	0.983	**0.942**	**0.947**	**0.928**
VUOD (Ours)	–	**0.994**	**0.991**	**0.994**	**0.972**	**0.981**	**0.971**

5.2 Result Analysis

Table 1 shows the significant improvements compared with several competitive baselines across a diverse range of outlier ratios, two feature extractors, and various benchmark datasets. Notably, even with the challenging datasets such as CIFAR-10 and CIFAR-100 [53], our method outperforms current SOTA AUC scores by margins of about 3% and 4%, respectively. Besides, we observe a 5% performance improvement on the scene understanding/recognition dataset, MIT-Places, increasing the AUC score from 0.928 to 0.971, which indicates the potential application of VUOD in enhancing performance on related tasks.

In Table 2, we demonstrate the results on benchmark datasets that go beyond natural images, such as industrial defect detection datasets (e.g., MVTec-AD, MVTec-LOCO) and medical disease detection datasets (e.g., LiverCT and BrainMRI) that are usually patch/pixel-level contaminated. To maintain the evaluation consistency with Table 1, we still use the default feature embedding of the entire image, which does not seem like the optimal choice, but VUOD still significantly outperforms the baselines. Without any supervision, its AUC score exceeds 0.910 on industrial datasets. In Table 3, we compare VUOD with several recent industrial defect detection methods on MVTec-AD. Remarkably, although VUOD is not specifically designed for this task, it still delivers competitive results. Moreover, it supports ResNet-18 as a reliable backbone, which offers a significant advantage for practical deployment.

Table 2: AUC results on industrial inspection and medical imaging (disease detection). The best results are highlighted in bold.

Type	Dataset	ResNet-18					Wide ResNet-101				
		RS	RAE	VAD	Multi-T	FlexUOD	VUOD	RS	RAE	VAD	Multi-T	FlexUOD	VUOD
Industrial	MVTec-AD	0.709	0.779	0.757	0.752	**0.911**		0.770	0.799	0.780	0.778	**0.931**	
	BTAD	0.873	0.894	0.885	0.882	**0.902**		0.846	0.884	0.886	0.883	**0.928**	
	MPDD	0.618	0.573	0.563	0.564	**0.906**		0.592	0.602	0.603	0.595	**0.926**	
	MVTec-LOCO	0.655	0.693	0.661	0.663	**0.812**		0.629	0.695	0.657	0.660	**0.832**	
Medical	LiverCT	0.666	0.663	0.696	0.708	**0.813**		0.703	0.571	0.652	0.567	**0.838**	
	BrainMRI	0.541	0.623	0.583	0.584	**0.804**		0.676	0.647	0.652	0.578	**0.790**	
	OCT	0.524	0.778	0.730	0.612	**0.880**		0.713	0.811	0.725	0.588	**0.877**	
	RESC	0.712	0.810	0.823	**0.948**	0.901		0.662	0.746	0.644	0.589	**0.931**	

Table 3: AUC results for each category on MVTec-AD.

Type	Category	PaDiM [15]	FRE [52]	EfficientAD [5]	VUOD	VUOD
	Backbone	ResNet-18	ResNet-50	Wide ResNet-101	ResNet-18	Wide ResNet-101
Textures	Carpet	0.986	0.968	0.969	0.875	0.901
	Grid	0.865	0.869	0.996	0.816	0.859
	Leather	0.997	1.000	0.976	0.841	0.971
	Tile	0.939	0.994	0.960	0.924	0.972
	Wood	0.987	0.992	0.903	0.939	0.991
Objects	Bottle	0.996	0.993	1.000	0.927	0.882
	Cable	0.789	0.892	0.852	0.929	0.908
	Capsule	0.851	0.744	0.753	0.931	0.948
	Hazelnut	0.852	0.940	0.766	0.949	0.946
	Metal Nut	0.928	0.914	0.977	0.949	0.948
	Pill	0.764	0.838	0.892	0.891	0.885
	Screw	0.811	0.789	0.940	0.903	0.921
	Toothbrush	0.958	0.931	0.970	0.956	0.977
	Transistor	0.908	0.894	0.853	0.929	0.908
	Zipper	0.896	0.942	0.956	0.909	0.944
	Average	0.902	0.913	0.917	0.911	**0.931**

Efficiency Analysis. We first conduct the efficiency comparison between AP and k-Means in Table 4 since it is the most important component of VUOD. Notably, the running time of k-Means depends on the cluster number k, which is known to be challenging to determine without any supervision. When k is set to the same as AP, AP runs more quickly. This occurs because our clustering tasks involve small sample sizes, where AP's quadratic message updates remain inexpensive. In contrast, k-Means performs multiple random initializations, each requiring high-dimensional distance computations, and often needs to be executed with several candidate values of k. AP avoids these costs by running a single deterministic optimization and automatically determining the number of clusters. Consequently, AP becomes empirically faster than k-Means in small-

Table 4: Timing (Seconds) is measured with 10,000 ResNet-50 feature samples. #AP refers to the automatic defined clustering number of AP.

AP	k-Means ($k = 10$)	k-Means ($k = 100$)	k-Means ($k =$#AP)
84.60	9.588	61.20	207.3

Table 5: Efficiency between VUOD and EfficientAD on MVTec-AD. We evaluate efficiency on the "bottle" category (input size: 256×256). Both methods use Wide ResNet-101 as the backbone and are conducted on one NVIDIA RTX 3090 GPU.

Method	Timing (Seconds)	FLOPs (G)	GPU Memory (MB)
EfficientAD	42.10	75.91	102.0
VUOD	**17.18**	**22.80**	**39.23**

scale and high-dimensional feature clustering, as evaluated in Table 4. In Table 5, our solution not only excels in detection accuracy but also efficiency on the industrial inspection dataset MVTec-AD, compared with EfficientAD [5].

Table 6: Ablation study about VUOD. The results are AUC scores of OCSVM with the raw and enhanced features.

Method	STL-10	Internet	Caltech-101	CIFAR-10	CIFAR-100	MIT-Places	MVTec-AD
OCSVM (Raw Feat.)	0.899	0.903	0.955	0.824	0.845	0.824	0.754
OCSVM (Re. Feat.)	0.994	0.991	0.994	0.972	0.981	0.971	0.921
Improvement	10.6%	9.75%	4.08%	18.0%	16.1%	17.8%	22.1%
Method	BTAD	MPDD	MVTec-LOCO	LiverCT	BrainMRI	OCT	RESC
OCSVM (Raw Feat.)	0.874	0.575	0.662	0.597	0.605	0.747	0.701
OCSVM (Re. Feat.)	0.915	0.916	0.822	0.825	0.797	0.879	0.916
Improvement	4.69%	59.3%	24.2%	38.2%	31.7%	17.7%	30.7%

Table 7: Ablation study about parameter setting (ϵ^+, ϵ^-) in Sec. 4.3. The results are AUC scores of VUOD on CIFAR-10.

Method	(0.9, 0.7)	(0.9, 0.6)	(0.9, 0.5)	(0.9, 0.4)	(0.8, 0.6)	(0.8, 0.5)	(0.8, 0.4)	(0.8, 0.3)
VUOD	0.969	0.971	**0.972**	0.970	0.968	0.969	0.968	0.965
Method	(0.7, 0.5)	(0.7, 0.4)	(0.7, 0.3)	(0.7, 0.2)	(0.6, 0.4)	(0.6, 0.3)	(0.6, 0.2)	(0.6, 0.1)
VUOD	0.970	0.968	0.967	0.964	0.967	0.966	0.964	0.963

5.3 Ablation Study

To measure the feature discriminativeness enhancement contribution of the VUOD framework, we conduct ablation studies on all benchmark datasets, and AUC results are averaged on two feature extractors. Compared with the default OCSVM, our feature enhancement techniques improve the performance up to 59.3%, as shown in Table 6. For all evaluated benchmarks, we used a consistent set of default parameters (ϵ^+, ϵ^-) in Sec. 4.3 without per-category tuning. To further validate this, we conducted an additional robustness study (Table 7), shifting the default thresholds by 0.2 resulted in only a marginal performance fluctuation ($\leq 0.5\%$ in AUC score), suggesting that they are not sensitive to specific values.

Table 8: Image classification results on Imagenette [28], a smaller set of ImageNet [17]. The image classifier baseline we employed is ResNet-18.

Method	Top-1 Accuracy	Top-5 Accuracy
Baseline	90.81%	99.11%
w/ VUOD	**91.45%**	**99.39%**

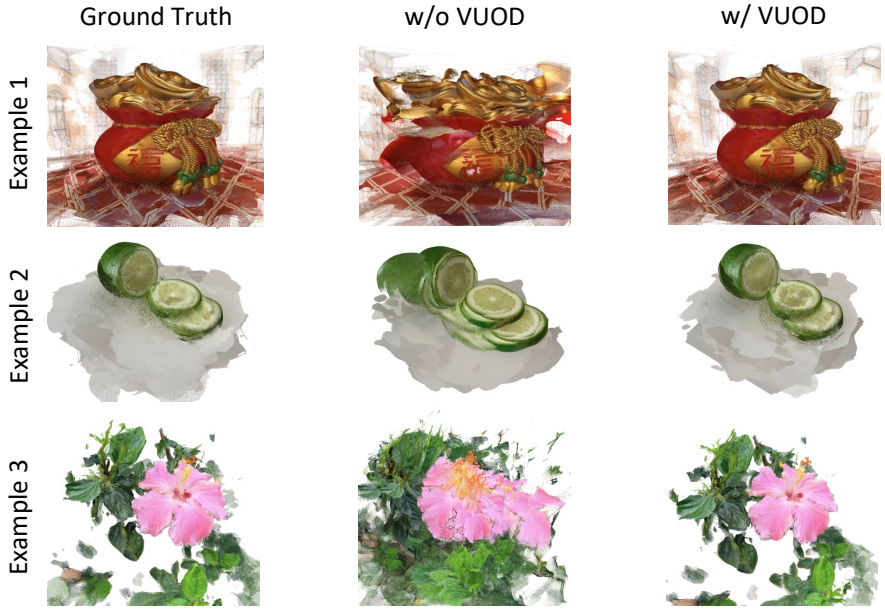

Fig. 9: 3D Dense Point Map Reconstruction Results of VGGT [66]. Ground Truth refers to the reconstruction obtained using VGGT on an uncontaminated image set. Three target datasets are visualized in the Appendix.

5.4 Diverse Visual Applications

The above standard UOD evaluation focuses on detecting outliers from various domains, but lacks applying this mechanism to other topics. To further validate the versatile ability of our VUOD framework, we consider it as an image-level data pre-processor to filter anomalous samples from the training/input set of two important visual tasks: image classification and 3D reconstruction. And we compute the threshold τ based on the predicted outlier ratio γ from FlexUOD [47], i.e., $\tau = \text{sort}(\{f(x_i)\}_{i=1}^{n})[n \cdot (1-\gamma)]$. If the predicted outlier score $f(x_i) > \tau$, x_i will be classified as an outlier. Otherwise, it belongs to the inliers.

Image Classification. By integrating VUOD into the classification pipeline, VUOD can effectively identify and remove outliers from the default training set, which naturally involves some noisy images [72], thereby improving the classification accuracy of ResNet-18, as shown in Table 8.

3D Reconstruction. We further extend VUOD to 3D reconstruction, specifically within the VGGT [66] framework. As VGGT requires a sequence of images,

it is inherently sensitive to image contamination. The absence of an outlier filtering mechanism leads to significant reconstruction artifacts, illustrated in Fig. 9. We compare our solution against the latest RobustVGGT [25], which also employs image-level rejection but is tailored for VGGT. Notably, RobustVGGT requires a time-consuming preliminary pass through the VGGT pipeline to extract internal representations. In contrast, VUOD is feature-agnostic, supporting flexible backbones such as ResNet and CLIP. In experiments across three target datasets in Fig. 9, both two approaches successfully identified all outliers. However, RobustVGGT requires 118 ms per image, VUOD reduces processing time to just 27 ms, achieving a 4.4× speedup on an NVIDIA RTX 4090D GPU.

5.5 Limitation

Since VUOD aims to be available for various UOD domains, it operates at the whole-image level and is designed to rank samples based on their outlier scores, it may be less effective for patch-level or pixel-level outlier segmentation tasks [13].

6 Conclusion

In this work, we aim to break the application limitation of traditional unsupervised outlier detection by introducing the VUOD framework. It is designed to improve feature discriminativess via pseudo labeling and contrastive learning. Our evaluation demonstrates that VUOD offers promising performance and stability across not only standard evaluation but also diverse applications such as industrial inspection and medical imaging. Additionally, VUOD is distinguished by its plug-and-play advantage that facilitates easy integration with other critical computer vision tasks like image classification and 3D reconstruction, to improve their classification accuracy and reconstruction robustness.

Acknowledgement

This work is partially supported by Shenzhen Science and Technology Program KQTD20210811090149095.

References

1. Afsari, B.: Riemannian l_p center of mass: existence, uniqueness, and convexity. PAMS pp. 655–673 (2011)
2. Aggarwal, C.C.: An Introduction to Outlier Analysis. Springer (2017)
3. Bao, J., Sun, H., Deng, H., He, Y., Zhang, Z., Li, X.: Bmad: Benchmarks for medical anomaly detection. In: CVPR. pp. 4042–4053 (2024)
4. Barbado, A., Corcho, Ó., Benjamins, R.: Rule extraction in unsupervised anomaly detection for model explainability: Application to oneclass svm. Epert Syst. Appl. p. 116100 (2022)

5. Batzner, K., Heckler, L., König, R.: Efficientad: Accurate visual anomaly detection at millisecond-level latencies. In: WACV. pp. 127–137 (2024)
6. Bergman, L., Hoshen, Y.: Classification-based anomaly detection for general data. In: ICLR (2020)
7. Bergmann, P., Batzner, K., Fauser, M., Sattlegger, D., Steger, C.: Beyond dents and scratches: Logical constraints in unsupervised anomaly detection and localization. IJCV pp. 947–969 (2022)
8. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In: CVPR. pp. 9592–9600 (2019)
9. Boente, G., Pires, A.M., Rodrigues, I.M.: Influence functions and outlier detection under the common principal components model: A robust approach. Biometrika pp. 861–875 (2002)
10. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: Lof: identifying density-based local outliers. In: SIGMOD. pp. 93–104 (2000)
11. Van den Burg, G.J., Williams, C.K.: An evaluation of change point detection algorithms. arXiv preprint arXiv:2003.06222 (2020)
12. Cai, Y., Zhang, W., Chen, H., Cheng, K.T.: Medianomaly: A comparative study of anomaly detection in medical images. Medical Image Analysis p. 103500 (2025)
13. Chan, R., Lis, K., Uhlemeyer, S., Blum, H., Honari, S., Siegwart, R., Fua, P., Salzmann, M., Rottmann, M.: Segmentmeifyoucan: A benchmark for anomaly segmentation. In: NeurIPS (2021)
14. Coates, A., Ng, A., Lee, H.: An analysis of single-layer networks in unsupervised feature learning. In: AISTATS. pp. 215–223 (2011)
15. Defard, T., Setkov, A., Loesch, A., Audigier, R.: Padim: a patch distribution modeling framework for anomaly detection and localization. In: ICPR. pp. 475–489 (2021)
16. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A large-scale hierarchical image database. In: CVPR. pp. 248–255 (2009)
17. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR. pp. 248–255 (2009)
18. Du, X., Wang, Z., Cai, M., Li, Y.: Vos: Learning what you don't know by virtual outlier synthesis. In: ICLR (2022)
19. Du, X., Wang, Z., Cai, M., Li, Y.: Dream the impossible: Outlier imagination with diffusion models. In: NeurIPS. vol. 36, pp. 60878–60901 (2023)
20. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. TPAMI pp. 594–611 (2006)
21. Fernando, T., Gammulle, H., Denman, S., Sridharan, S., Fookes, C.: Deep learning for medical anomaly detection–a survey. CSUR (2021)
22. Fieller, E.C., Pearson, E.S.: Tests for rank correlation coefficients: Ii. Biometrika pp. 29–40 (1961)
23. Friendly, M., Monette, G., Fox, J.: Elliptical insights: understanding statistical methods through elliptical geometry. STAT SCI pp. 1–39 (2013)
24. Goodge, A., Hooi, B., Ng, S.K., Ng, W.S.: Lunar: Unifying local outlier detection methods via graph neural networks. In: AAAI. pp. 6737–6745 (2022)
25. Han, J., Hong, S., Jung, J., Jang, W., An, H., Wang, Q., Kim, S., Feng, C.: Emergent outlier view rejection in visual geometry grounded transformers. In: CVPR. pp. 427–437 (2026)
26. Hashemi, N., German, E.V., Ramirez, J.P., Ruths, J.: Filtering approaches for dealing with noise in anomaly detection. In: CDC. pp. 5356–5361 (2019)

27. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
28. Howard, J.: Imagenette: A smaller subset of the imagenet dataset (2019)
29. Huyan, N., Quan, D., Zhang, X., Liang, X., Chanussot, J., Jiao, L.: Unsupervised outlier detection using memory and contrastive learning. TIP pp. 6440–6454 (2022)
30. Iouchtchenko, D., Raymond, N., Roy, P.N., Nooijen, M.: Deterministic and quasi-random sampling of optimized gaussian mixture distributions for vibronic monte carlo. arXiv preprint arXiv:1912.11594 (2019)
31. Isaac-Medina, B.K.S., Pérez-Carrasco, J.A., Galasso, F.: Towards open-world object-based anomaly detection via self-supervised outlier synthesis. In: ECCV (2024)
32. Jezek, S., Jonak, M., Burget, R., Dvorak, P., Skotak, M.: Deep learning-based defect detection of metal parts: evaluating current methods in complex conditions. In: ICUMT. pp. 66–71 (2021)
33. Keyzer, M.A., Sonneveld, B.: Using the mollifier method to characterize datasets and models: the case of the universal soil loss equation. ITC Journal pp. 263–272 (1997)
34. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
35. Kriegel, H.P., Schubert, M., Zimek, A.: Angle-based outlier detection in high-dimensional data. In: KDD. pp. 444–452 (2008)
36. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Master's thesis, Department of Computer Science, University of Toronto (2009)
37. Lai, C.H., Zou, D., Lerman, G.: Robust subspace recovery layer for unsupervised anomaly detection. In: ICLR (2020)
38. Li, T., Wang, Z., Liu, S., Lin, W.Y.: Deep unsupervised anomaly detection. In: WACV. pp. 3636–3645 (2021)
39. Li, Z., Zhao, Y., Hu, X., Botta, N., Ionescu, C., Chen, G.H.: Ecod: Unsupervised outlier detection using empirical cumulative distribution functions. TKDE pp. 9091–9104 (2023)
40. Lin, D., Liu, S., Li, H., Cheung, N.M., Ren, C., Matsushita, Y.: Shell theory: A statistical model of reality. TPAMI (2021)
41. Lin, W.Y., Liu, S., Dai, B.T., Li, H.: Distance based image classification: A solution to generative classification's conundrum? IJCV pp. 177–198 (2023)
42. Lin, W.Y., Liu, S., Lai, J.H., Matsushita, Y.: Dimensionality's blessing: Clustering images by underlying distribution. In: CVPR. pp. 5784–5793 (2018)
43. Lin, W.Y., Liu, Z., Liu, S.: Locally varying distance transform for unsupervised visual anomaly detection. In: ECCV. pp. 354–371 (2022)
44. Liu, F.T., Ting, K.M., Zhou, Z.H.: Isolation forest. In: ICDM. pp. 413–422 (2008)
45. Liu, J., Xie, G., Wang, J., Li, S., Wang, C., Zheng, F., Jin, Y.: Deep industrial image anomaly detection: A survey. Machine Intelligence Research pp. 104–135 (2024)
46. Liu, Z., Lu, P., Xie, G., Lu, Z., Lin, W.Y.: Rethinking unsupervised outlier detection via multiple thresholding. In: ECCV. pp. 258–275 (2024)
47. Liu, Z., Zhou, K., Wang, C., Lin, W.Y., Lu, J.: Flexuod: The answer to real-world unsupervised image outlier detection. In: CVPR. pp. 15183–15193 (2025)
48. Lloyd, S.: Least squares quantization in pcm. TIT pp. 129–137 (1982)
49. MacQueen, J.: Classification and analysis of multivariate observations. In: 5th Berkeley Symp. Math. Statist. Probability. pp. 281–297 (1967)
50. Martin, M.A., Roberts, S.: An evaluation of bootstrap methods for outlier detection in least squares regression. J. Appl. Stat pp. 703–720 (2006)

51. Mishra, P., Verk, R., Fornasier, D., Piciarelli, C., Foresti, G.L.: Vt-adl: A vision transformer network for image anomaly detection and localization. In: ISIE. pp. 01–06 (2021)
52. Ndiour, I.J., Ahuja, N.A., Genc, U., Tickoo, O.: Fre: A fast method for anomaly detection and segmentation. In: BMVC (2023)
53. Perera, P., Nallapati, R., Xiang, B.: Ocgan: One-class novelty detection using gans with constrained latent representations. In: CVPR. pp. 2898–2906 (2019)
54. Perini, L., Bürkner, P.C., Klami, A.: Estimating the contamination factor's distribution in unsupervised anomaly detection. In: ICML. pp. 27668–27679 (2023)
55. Qi, Z., Jiang, D., Chen, X.: Iterative gradient descent for outlier detection. IJWMIP p. 2150004 (2021)
56. Qiu, C., Pfrommer, T., Kloft, M., Mandt, S., Rudolph, M.: Neural transformation learning for deep anomaly detection beyond images. In: ICML. pp. 8703–8714 (2021)
57. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML. pp. 8748–8763 (2021)
58. Reiss, T., Hoshen, Y.: Mean-shifted contrastive loss for anomaly detection. In: AAAI. pp. 2155–2162 (2023)
59. Roth, K., Pemula, L., Zepeda, J., Schölkopf, B., Brox, T., Gehler, P.: Towards total recall in industrial anomaly detection. In: CVPR. pp. 14318–14328 (2022)
60. Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S.A., Binder, A., Müller, E., Kloft, M.: Deep one-class classification. In: ICML. pp. 4393–4402 (2018)
61. Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C.: Estimating the support of a high-dimensional distribution. Neural Computation pp. 1443–1471 (2001)
62. Shenkar, T., Wolf, L.: Anomaly detection for tabular data with internal contrastive learning. In: ICLR (2021)
63. Shoemaker, D., Schadt, E., Armour, C., He, Y., Garrett-Engele, P., McDonagh, P., Loerch, P., Leonardson, A., Lum, P., Cavet, G., et al.: Experimental annotation of the human genome using microarray technology. Nature pp. 922–927 (2001)
64. Tao, L., Du, X., Wang, Z., Li, Y.: Non-parametric outlier synthesis. In: ICLR (2023)
65. Thanammal, K., Jayasudha, J., Vijayalakshmi, R., Arumugaperumal, S.: Effective histogram thresholding techniques for natural images using segmentation. JOIG pp. 113–116 (2014)
66. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. In: CVPR. pp. 5294–5306 (2025)
67. Wang, S., Zeng, Y., Liu, X., Zhu, E., Yin, J., Xu, C., Kloft, M.: Effective end-to-end unsupervised outlier detection via inlier priority of discriminative network. In: NeurIPS (2019)
68. Wang, S., Zeng, Y., Yu, G., Cheng, Z., Liu, X., Zhou, S., Zhu, E., Kloft, M., Yin, J., Liao, Q.: $E^3$3outlier: a self-supervised framework for unsupervised deep outlier detection. TPAMI pp. 2952–2969 (2023)
69. Xu, H., Pang, G., Wang, Y., Wang, Y.: Deep isolation forest for anomaly detection. TKDE (2023)
70. Xu, H., Wang, Y., Wei, J., Jian, S., Li, Y., Liu, N.: Fascinating supervisory signals and where to find them: Deep anomaly detection with scale learning. In: ICML. pp. 38655–38673 (2023)
71. Ye, H., Liu, Z., Shen, X., Cao, W., Zheng, S., Gui, X., Zhang, H., Chang, Y., Bian, J.: Uadb: Unsupervised anomaly detection booster. In: ICDE. pp. 2593–2606 (2023)

72. Yun, S., Oh, S.J., Heo, B., Han, D., Choe, J., Chun, S.: Re-labeling imagenet: from single to multi-labels, from global to localized labels. In: CVPR. pp. 2340–2350 (2021)
73. Zagoruyko, S., Komodakis, N.: Wide residual networks. In: BMVC (2016)
74. Zhao, Y., Rossi, R.A., Akoglu, L.: Automating outlier detection via meta-learning. arXiv preprint arXiv:2009.10606 (2020)
75. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: A 10 million image database for scene recognition. TPAMI pp. 1452–1464 (2017)
76. Zhou, C., Paffenroth, R.C.: Anomaly detection with robust deep autoencoders. In: KDD. pp. 665–674 (2017)
77. Zhu, J., Ding, C., Tian, Y., Pang, G.: Anomaly heterogeneity learning for open-set supervised anomaly detection. In: CVPR. pp. 17616–17626 (2024)
78. Zong, B., Song, Q., Min, M.R., Cheng, W., Lumezanu, C., Cho, D., Chen, H.: Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In: ICLR (2018)
79. Zou, Y., Jeong, J., Pemula, L., Zhang, D., Dabeer, O.: Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In: ECCV. pp. 392–408 (2022)