Towards Zero-shot 3D Anomaly Localization

Yizhou Wang[1*] Kuan-Chuan Peng[2] Yun Fu[1]
[1]Northeastern University [2]Mitsubishi Electric Research Laboratories
wyzjack990122@gmail.com kpeng@merl.com yunfu@ece.neu.edu

Abstract

3D anomaly detection and localization is of great significance for industrial inspection. Prior 3D anomaly detection and localization methods focus on the setting that the testing data share the same category as the training data which is normal. However, in real-world applications, the normal training data for the target 3D objects can be unavailable due to issues like data privacy or export control regulation. To tackle these challenges, we identify a new task – zero-shot 3D anomaly detection and localization, where the training and testing classes do not overlap. To this end, we design 3DzAL *, a novel patch-level contrastive learning framework based on pseudo anomalies generated using the inductive bias from task-irrelevant 3D xyz data to learn more representative feature representations. Furthermore, we train a normalcy classifier network to classify the normal patches and pseudo anomalies and utilize the classification result jointly with feature distance to design anomaly scores. Instead of directly using the patch point clouds, we introduce adversarial perturbations to the input patch xyz data before feeding into the 3D normalcy classifier for the classification-based anomaly score. We show that* 3DzAL *outperforms the state-of-the-art anomaly detection and localization performance.*

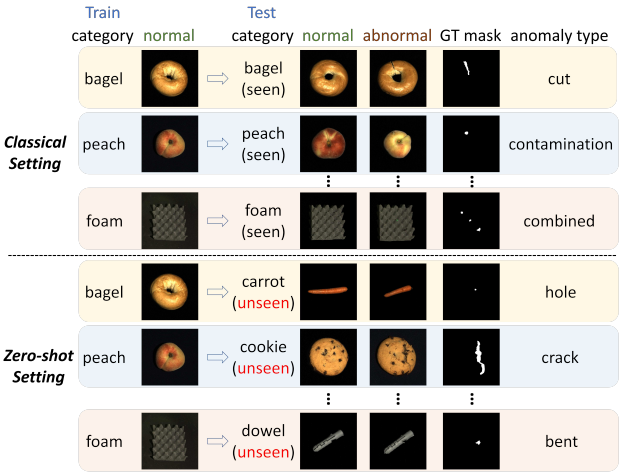

Figure 1. **Problem overview.** Current 3D anomaly detection and localization works entail training on the normal data of one class and testing on the normal and abnormal data of the same class. We extend such setting by testing on other classes without the corresponding normal training data. This zero-shot setting is practical when such data are unavailable (*e.g.*, due to data privacy, export control laws, *etc.*). GT denotes ground truth.

1. Introduction

3D anomaly detection and localization methods have been highly demanded in real-world circumstances, including industrial inspection and autonomous driving [3,6,16,17,30,43,45–47,49,50]. The main difference between 3D and 2D image anomaly detection and localization lies in that 3D data contain not only RGB information but also point location information [3,8,9,30]. Lots of shape anomalies of objects are readily identified as distinct sharp deformations from the point locations, in which cases, color information is less effective and the anomalies remain undetectable in top-down 2D views, as recognized in [23]. For instance, it is extraordinarily hard to identify a bent or cut location in a 2D image of a dowel, but such anomaly type can be very obvious in the 3D point cloud data. Recently, various 3D anomaly detection and localization methods have been introduced [5,23,35,42,54]. All these existing works concentrate on the setting that the testing data (including both normal and abnormal data) are from the same class as the training data. However, in real-world industrial 3D anomaly detection and localization applications, the normal training data of the target objects can be unavailable due to many possible reasons, *e.g.*, data privacy, export control regulations, *etc*. Sometimes the normal data of the target objects on the client side are sensitive, and the client may not want to share the data but only want an anomaly detection and localization method that can perform well "off-the-shelf." Therefore, a 3D anomaly detection and localization method able to generalize to unseen classes in the testing phase is needed.

Problem Statement. To address the aforesaid issues, we define a new problem in zero-shot 3D anomaly detection and localization, which involves identifying anomalies within a particular target class without any access to training data for

*This work was done when Yizhou Wang was an intern at Mitsubishi Electric Research Laboratories.

that class or prior knowledge of its specific type of anomaly pattern. To be more specific, our goal is to localize abnormal locations in the 3D data in the target class's testing set, with no need of target class training data. Fig. 1 illustrates this problem.

Proposed framework. To solve the aforesaid new problem, we propose a novel framework, namely "3D zero-shot anomaly localization" (3DzAL). To achieve satisfactory zero-shot performance in 3D anomaly detection and localization, we add a learnable 3D feature extraction network on top of the 3D FPFH [36] features and encourage the learned features to be complementary to the features captured in FPFH. To regularize the 3D feature extraction network, we use a patch-level pseudo anomaly-based contrastive learning scheme. We propose a pseudo anomaly generation module to synthesize anomalies since the training data only include the normal data without any abnormal data. When designing the pseudo anomaly generation module, we find that a randomly initialized and untrained CNN is able to locate the places of interest in three-dimensional point cloud data in its feature activation maps, *i.e*., if we feed the three-dimensional point cloud data as the input of a random CNN, the highly activated areas in this CNN's feature activation maps usually cover the locations of interest, *e.g*., crack, hole, *etc*. Based on this finding, we use the places of interest identified by the random CNN to synthesize pseudo abnormal patches in the pseudo anomaly generation module, which is the first attempt to use such inductive bias of random networks in 3D anomaly localization and detection.

In our proposed zero-shot 3D anomaly localization and detection setting, since the training data of the target objects are unavailable, we incorporate the 3D data from other objects (which we refer to as task-irrelevant data, *i.e*., the objects belonging to the categories different from the testing category) to synthesize the pseudo anomaly patches. We extract the 3D features of both the normal patches and pseudo abnormal patches and use a contrastive learning objective to further regularize the learned 3D feature extraction network. To enhance the anomaly localization and detection ability, we also introduce a normalcy classifier to distinguish the normal patches from the pseudo-abnormal patches to gain the discriminative ability between general normal and abnormal 3D objects. We add adversarial perturbations to the input point cloud patch utilizing the gradient of the negative log-likelihood loss applied to the testing data. Eventually, we combine the normalcy classification output score of the perturbed data and the distance-based score of the original using the RGB and FPFH features plus our learned 3D features to formulate the final anomaly score. We demonstrate that our proposed method 3DzAL outclasses the SOTA 3D anomaly localization and detection method within the zero-shot framework. In summary, our key **contributions** are as follows:

1. We formally introduce a new problem in 3D anomaly detection and localization where the model undergoes training using the normal data to detect anomalies (during testing) in a varied class without undergoing any adaptation through the target-class training data.
2. We propose a novel zero-shot 3D anomaly detection and localization method, 3DzAL , where our designed network learns the relative and general difference between the normal and abnormal 3D object data in the training class and generalizes to the target class without needing the target class training data or any models pre-trained by 3D data.
3. Intriguingly and notably, for the very first time (as far as we are aware), we show that a randomly initialized and untrained CNN has the inductive bias to localize places of interest on three-dimensional point cloud data, and its localization ability is better than an ImageNet-pretrained CNN.
4. As far as we are aware, this is the first attempt to incorporate the input perturbation technique into 3D anomaly detection and localization problems and show its efficacy.

2. Related works

3D anomaly localization and detection is crucial in industrial scenarios [3, 23]. With the emergence of the first 3D anomaly localization and detection dataset MVTec 3D-AD [3], a great number of anomaly detection and localization methods for three-dimensional point cloud data have been introduced. [3] proposed to use generative adversarial networks, autoencoders, and variational models in both voxel-level and depth-level modeling. [4, 35] adopted student-teacher frameworks for anomaly detection and take advantage of the distance between the student and teacher model output as anomaly score. [23] proposed the 3D version of Patchcore [34], which utilizes a core-set assisted memory bank for normal feature storage and employs the distance between the testing sample feature and the normal memory bank as an anomaly score. More recently, [5] proposed a collaborative discrepancy optimization method with the help of synthetic anomalies, [42] came up with a new position-encoding-augmented feature mapping for anomaly detection, and [44] suggested a hybrid feature fusion technique for multimodal industrial anomaly detection. [10] developed a method using a dual-expert framework that combines 3D geometric information and 2D color features, but it required training the expert models using the training data of all categories, which is not feasible under our proposed zero-shot setting. [53] proposed a novel method called 3DSR, which utilizes a Depth-Aware Discrete Autoencoder architecture and a simulated depth data generation process to jointly model RGB and 3D data, achieving the best anomaly localization and detection performance so far. Despite the efficacy of the above solutions, they require the testing samples to share the same class as the training samples. Once the training and testing data distributions differ, the performance will be largely compromised. In contrast, 3DzAL learns more generally discriminative features and aims to find the essential difference between normal and abnormal 3D data. Specifically, we propose to employ the inductive bias to generate pseudo-abnormal examples and use contrastive learning on top of them.

Low-shot anomaly detection which is composed of few-shot anomaly detection and zero-shot anomaly detection, has been

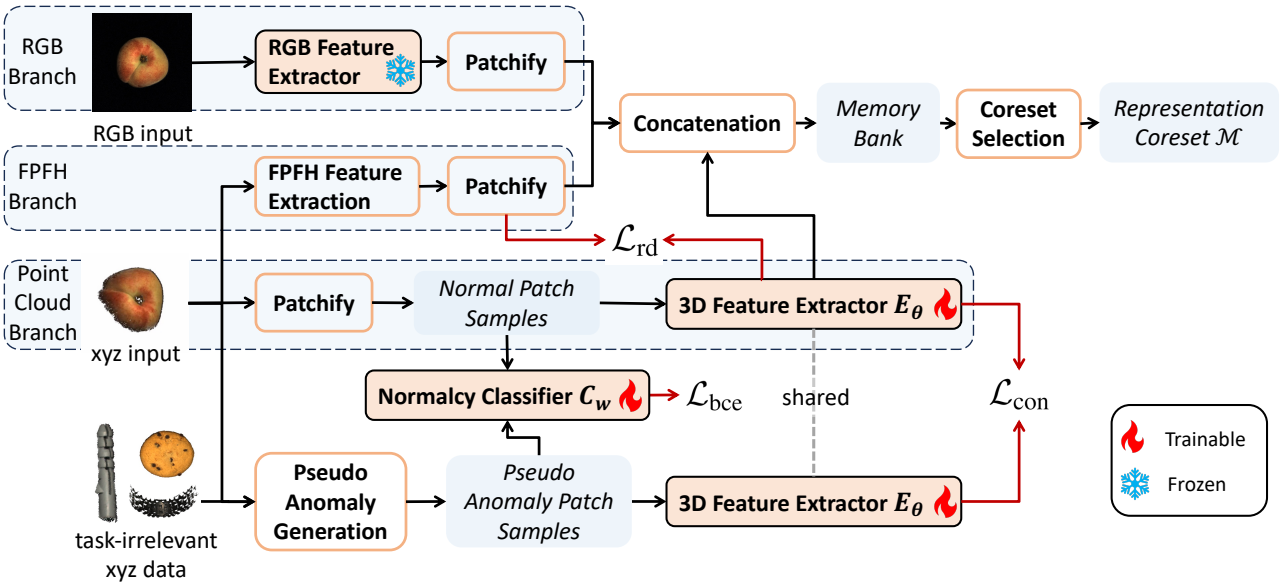

Figure 2. **Framework overview.** Our proposed 3DzAL framework mainly adopts three branches to extract features given both 2D and 3D data of an object. The RGB branch extracts feature from 2D image data of the object using ResNet pre-trained on ImageNet. The FPFH branch extracts handcrafted FPFH features from 3D point cloud data. The point cloud branch employs a learnable network (PointNet++) to extract features. The network is trained by a patch-level contrastive learning loss, which takes inductive bias-based pseudo anomaly patches as negative samples and normal patches as positive samples and a representation disentanglement loss which pushes the FPFH features and the learned 3D features away. The features of the three branches are concatenated to store in the memory bank where a coreset selection is performed. In addition, a normalcy classifier is trained to classify the pseudo anomaly patch and the normal patch using the binary cross-entropy loss.

attracting attention in anomaly detection research recently. For few-shot anomaly detection, some works [11, 12, 24, 34, 39, 48] reflected the notion of "few-shot" in only using a much smaller number of normal training samples, and others [14, 21, 27, 29, 31, 41] explored the setting that a few abnormal samples can be accessed during testing. In the context of zero-shot anomaly detection, the current dedication to such research direction is still limited. [15, 28] exploited the transfer learning power of the pretrained CLIP models [33] for image-level out-of-distribution detection or anomaly detection without the normal training data. [37] investigated the capacity of ImageNet-pretrained masked autoencoder [19] for zero-shot image anomaly detection via adopting the reconstruction discrepancy as anomaly score, and [1] tackled the zero-shot setting in video anomaly detection. More recently, [56] leverages text prompts that are not tied to specific objects, allowing it to identify general patterns of normality and abnormality, making it effective for zero-shot anomaly detection across different domains. [25] relies on custom-designed text prompts to map image features to abnormal areas, utilizing CLIP's capabilities for zero-shot anomaly recognition. Better than all the existing works, 3DzAL needs no pre-trained model and makes the first attempt to execute zero-shot 3D anomaly detection and localization.

3. Proposed 3DzAL framework

Method overview. Given the normal training data from one particular class, our aim is to learn the representation that can ideally transfer across different classes without the need for the normal data of the testing class. To achieve this goal, we introduce an innovative 3DzAL framework which is depicted in Fig. 2. 3DzAL is built on the basis of a memory bank restoration and feature distance calculation paradigm. Specifically, 3DzAL is composed of a random CNN-based pseudo 3D anomaly sample generation module with the assistance of task-irrelevant data, 3D feature contrastive learning using pseudo anomaly, and a 3D point cloud sample normalcy classifier trained using the normal training sample and the synthesized pseudo anomaly sample. Finally, the distance-based score using the contrastive-learned features and the normalcy classifier output score using perturbed patch inputs are weighted and integrated to form the final anomaly score.

Our paper focuses on the 3D anomaly location detection, so the directions of existing zero-shot AD works in RGB images are complementary to what we propose. We intentionally do not make use of any existing zero-shot AD work and show that our proposed method still outperforms the SOTA. If we use existing zero-shot AD works, then we won't be able to claim such novelty in our method. We leave the integration of zero-shot RGB anomaly detection techniques [15, 25, 28, 37, 56] for performance gain for future work.

Notation. We denote a 3D point cloud sample as $X \in \mathbb{R}^{N\times 3}$ and its ordered version as $\mathcal{X} \in \mathbb{R}^{H\times W\times 3}$, where N is the number of points and H,W is the height and width of the corresponding 2D ordered map, either 3D data or 2D RGB values

and $H \times W = N$. Here "ordered" means the 3D point cloud data are in the ordered 2D image form but the three-channel values of each pixel are xyz instead of RGB values. The object $\mathcal{X}$ is partitioned into patches along the width and height. We denote each patch as x and $\mathcal{X} = \square_x$, where $\square$ refers to the practice of "realigning" x "based on their respective spatial location" as defined in [34]. The 3D representation extraction network is denoted as E_θ with parameter θ, and the normalcy classifier network is represented as C_w with parameter w.

3.1. Normal feature extraction

The Patchcore [34] is one of the SOTA methods for 2D industrial anomaly localization and detection on the MVTec-2D dataset [2]. BTF [23] is the 3D data version of the PatchCore and achieves the SOTA anomaly detection and localization result on the dataset MVTec-3D [3]. Following these works, we also first extract features from the normal data samples and store them in the memory bank. In particular, we extract RGB features from the 2D RGB image of the 3D object and handcrafted 3D FPFH [36] features from the corresponding 3D point cloud sample. As illustrated in the point cloud branch of Fig. 2, we add an additional 3D network to extract learnable features. The network is learned using contrastive learning and a feature disentanglement objective.

3.2. Learning discriminative 3D representations

Pseudo anomaly generation with inductive bias. To generate satisfactory anomaly detection and localization performance, we require our point cloud branch to reflect a clear distinction between the testing anomalies and the testing normal samples. However, considering that the training samples belong to different categories compared to the testing class within the zero-shot setting, if we want to regularize the training of E_θ, we need to mimic the disparities between the normal and the abnormal 3D samples regardless of the class prior. This motivates us to synthesize pseudo anomalies and perform contrastive learning between the pseudo anomalies and the normal samples. [7] showed that a randomly initialized and untrained convolutional neural network (CNN) inherently possesses an inductive bias to focus on objects, *i.e.*, even a randomly initialized CNN can generate biased activation maps towards objects of interest on a 2D image. Aich *et al.* [1] are the first to utilize such inductive bias of an untrained CNN to extract objects from task-irrelevant data and attach to the normal data to synthesize 2D pseudo anomaly image samples.

In this work, we discover that **such inductive bias also exists for 3D point cloud data**. More specifically, given an ordered point cloud data $\mathcal{X} \in \mathbb{R}^{H \times W \times 3}$, we employ an untrained ResNet-50 randomly initialized using the He initialization [20], which is denoted as $R(\cdot)$. Here the channel values of the input are xyz location values instead of RGB values when feeding the inputs. We choose to use the reciprocal second, third, and fourth layer output of $R(\mathcal{X}) \in \mathbb{R}^{d \times h \times w}$ to generate and fuse the resulting activation maps. Specifically,

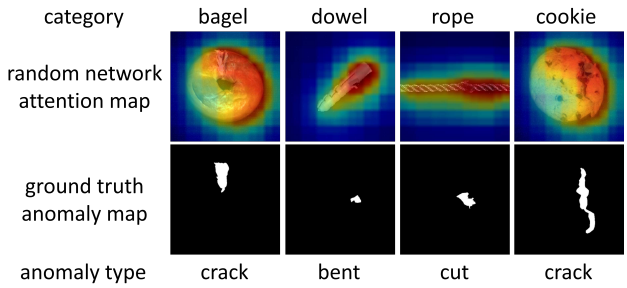

Figure 3. **Inductive bias of random networks.** We feed the xyz data of abnormal examples as the input of a randomly initialized and untrained ResNet-50, and visualize the attention maps. These maps show that the random network has the inductive bias of covering the locations of interest, including the locations shown in the ground truth.

the output sizes of the reciprocal second, third, and fourth layers are 14×14, 28×28, and 56×56, respectively. We first sum the values of all the channels in the feature map and then normalize them to the range $[0,1]$. Then we resize the output activation map of the reciprocal second and third layer output to the same spatial size as the reciprocal fourth layer, *i.e.*, 56×56. Then the three activation map values are added, averaged, and resized to the original input size to generate the soft mask, which we dub as $A \in \mathbb{R}^{H \times W}$. For the top τ (percentage) $A_{(i,j)}$ value points, we set mask $M_{(i,j)} = 1$, and for the rest, we set $M_{(i,j)} = 0$. Here (i,j) represents the position information in $H \times W$ locations. Finally, the ordered points localized out are $M_X = M \odot \mathcal{X}$, where $\odot$ is element-wise multiplication. As mentioned in [1, 7], randomly initialized CNN is able to localize objects in that the background is comparitively less textured compared to the foreground object and the regions of foreground tend to exhibit higher activation values under activation functions like ReLU [18]. However, our method is different in that we use multi-scale attention values for information fusing, and also surprising because our fed input is ordered xyz tensor (the point location information), not RGB values. In our experiments, we find that the highly activated areas correspond to the locations with shapes that are locations of interest (as illustrated in Fig. 3) and should be detected. This is intriguing because it means that there also exists inductive bias for 3D xyz input data, which shows that at the spatial level, the point clusters that exhibit abrupt deviations or alternations can be highly activated under a series of activation functions like ReLU. After selecting the points that show the places of interest from the task-irrelevant data, we attach the points of interest to the normal training sample. Then we move our anchor point (the center point around which the points are selected or sampled) to the geometry center of the anomaly points part plus some surrounding points, and use KD-tree [55] search algorithm to pick out the nearest point cloud part as the generated pseudo abnormal patch. The anomaly points part is attached to the surface by taking the anchor point as the geometric center. The

anchor point serves as the query point in the KD tree search. The KD tree search is conducted within the normal point cloud plus the anomaly points part with the anchor point as the query point. This can guarantee that the synthesized abnormal patch can contain both normal points and pseudo-generated points.

Besides such "adding-point" type pseudo abnormal patches which mimic anomaly types like bulging, contamination, or bent, we also involve another type of pseudo abnormal patch by setting the anchor at a random point of the surface of the normal sample, randomly sampling point cloud part and then randomly removing some ratio of points. Such kind of "removing-point" anomaly aims to resemble abnormal part types including cuts or holes. Our generated pseudo abnormal patches consist of both "adding-point" and "removing-point" types with the quantity ratio $1:1$. Fig. 4 illustrates the above process.

Contrastive learning with pseudo anomalies. To learn the representations that can robustly distinguish between the intrinsically abnormal and normal 3D samples, we use a contrastive learning objective that takes normal 3D patches as positive samples and the pseudo-abnormal patches as negative samples. As shown in Fig. 2, we add an additional 3D network PointNet++ [32] E_θ in the patch level to extract features. We adopt the contrastive learning loss as:

$$\mathcal{L}_{\text{con}} = \sum_{x_j \in \mathcal{X}_p} \frac{-1}{|\mathcal{P}(x_j)|} * \sum_{x_p \in \mathcal{P}(x_j)} \log \frac{\exp\left(\frac{E_\theta(x_j)\cdot E_\theta(x_p)}{T\cdot \|E_\theta(x_j)\|_2 \cdot \|E_\theta(x_p)\|_2}\right)}{\sum_{x_n \in \mathcal{N}(x_j)} \exp\left(\frac{E_\theta(x_j)\cdot E_\theta(x_n)}{T\cdot \|E_\theta(x_j)\|_2 \cdot \|E_\theta(x_n)\|_2}\right)}, \quad (1)$$

where $\mathcal{X}_p$ is the positive patch sample set, $\mathcal{P}(x_j)$ is the positive patch sample set besides x_j, $\mathcal{N}(x_j)$ is the negative patch sample set, and T is the temperature parameter. The purpose of Eq. (1) is to maximize the similarity between the learned feature representations of the positive samples while minimizing the similarity between the positive sample set and the negative sample set. Since the positive sample patches exhibit normal patterns while the negative samples are pseudo abnormal patches which we use task-irrelevant data from multiple categories to generate, in the testing phase, the network learned by Eq. (1) has the capacity to induce features that are far away from the normal feature memory bank when encountering abnormal samples during testing.

Representation disentanglement loss. To ensure that the learned point cloud branch output features are complementary to the handcrafted FPFH features, we design the representation disentanglement loss $\mathcal{L}_{\text{rd}}$, which aims at minimizing the cosine similarity $\cos(\cdot)$ between the extracted learnable feature $E_\theta(x)$ and the FPFH feature $F(x)$:

$$\mathcal{L}_{\text{rd}} = \cos(F(x), E_\theta(x)) = \frac{F(x)\cdot E_\theta(x)}{\|F(x)\|_2 \cdot \|E_\theta(x)\|_2}. \quad (2)$$

Therefore the loss function for training the network E_θ is the combination of the contrastive learning loss and the disentanglement loss: $\mathcal{L} = w_{\text{con}}\cdot\mathcal{L}_{\text{con}} + w_{\text{rd}}\cdot\mathcal{L}_{\text{rd}}$, where w_{con} and w_{rd} are the weights.

3.3. 3D normalcy classifier

For the positive and negative samples, we use an additional PointNet++ [32] network C_w for classification training. The normalcy classifier aims to distinguish between the normal sample and the synthetic abnormal ones and is formulated as a conventional binary classification problem. We adopt the binary cross-entropy loss $\mathcal{L}_{\text{bce}}$:

$$\mathcal{L}_{\text{bce}} = -\frac{1}{N}\sum_{i=1}^{N} \log(p(x_i|w))\cdot y_i + \log(1-p(x_i|w))\cdot(1-y_i), \quad (3)$$

where x_is are the training data composed of positive (normal training sample patch) and negative (pseudo abnormal patch) samples of the contrastive learning paradigm, and $p(x_i|w)$ is the softmax output probability of class 1 of C_w. y_is are binary labels and have value 0 for positive samples and value 1 for negative samples. In the testing phase, for each test sample patch, we use $p(x_i|w)$ as the patch-level anomaly score. The motivation is that we assign higher anomaly score values to the testing patch that is classified as abnormal (class 1) and lower score values to the testing patch that is classified as normal (class 0). This is because our classifier has been trained to discriminate between normal and pseudo-abnormal patches which are synthesized using task-irrelevant data belonging to multiple categories, which has been able to distinguish between the normal and abnormal 3D patches regardless of the class information.

Training and memory bank. After the training of E_θ, we extract the features of the training patches with it. The learned feature from the point cloud branch, concatenated with the RGB feature from RGB branch and the FPFH feature from FPFH branch, becomes the final feature representation of the patch x, and we denoted the concatenated feature of x as $f(x)$. Inspired by PatchCore [34], we store the extracted features of the training samples into a memory bank and run a minimax facility location-based coreset selection [38, 40] algorithm to reduce the computation burden. We use notion $\mathcal{M}$ for the reduced patch-level feature memory bank.

3.4. Anomaly score design

Distance-based score. Given a testing object sample X^{test}, we extract the three branch patch-level features in the same way as the training process using trained E_θ, and we denote the collected patch-level feature set as $f(\mathcal{X}^{\text{test}})$ and the features as $f(x^{\text{test}})$. Following [34], we utilize the maximum distance score S^* from $f(\mathcal{X}^{\text{test}})$ to the corresponding nearest neighbour f^* of the memory bank:

$$f(x^{\text{test},*}), f^* = \underset{f(x^{\text{test}})\in f(\mathcal{X}^{\text{test}})}{\operatorname{argmax}} \ \underset{f\in\mathcal{M}}{\operatorname{argmin}} \left\| f(x^{\text{test}}) - f \right\|_2, \quad (4)$$

$$S^* = \left\| f(x^{\text{test},*}) - f^* \right\|_2. \quad (5)$$

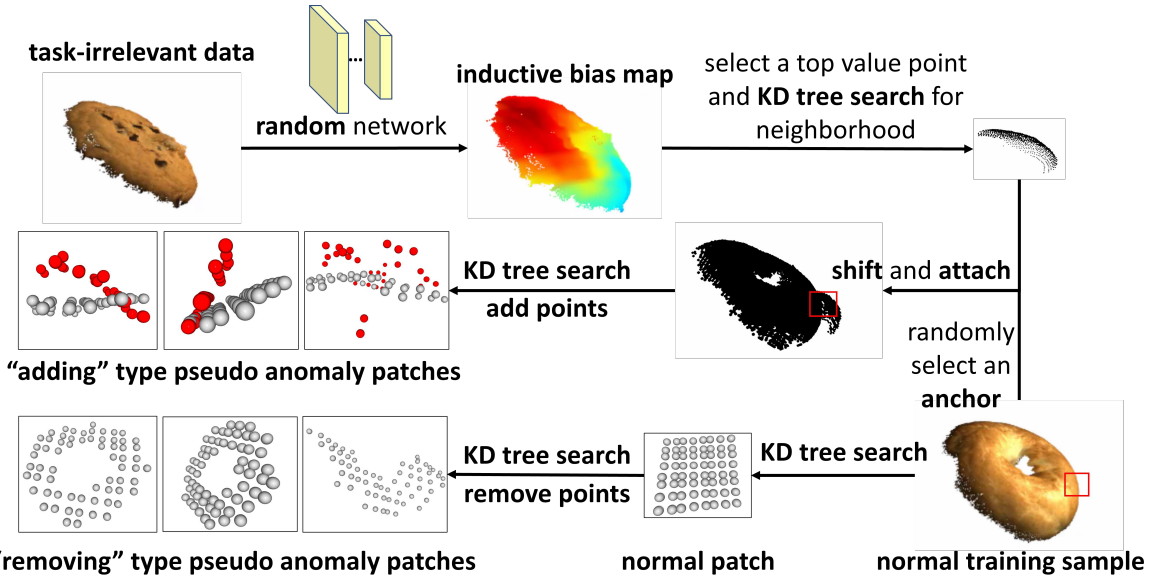

Figure 4. **Pseudo anomaly generation.** Overview of our proposed patch-level 3D pseudo anomaly sample generation process for both "adding" and "removing" type anomalies.

To obtain the object-level anomaly score S_{dist}, we impose an additional weight in the following form:

$$S_{\text{dist}}(X^{\text{test}}) = \left(1 - \frac{\exp\|f(x^{\text{test},*}) - f^*\|_2}{\sum_{f \in \mathcal{N}_b(f^*)} \exp\|f(x^{\text{test},*}) - f\|_2}\right) \cdot S^*, \quad (6)$$

where $\mathcal{N}_b(f^*)$ is the b nearest patch-level features in the memory bank for the test patch-feature f^*. The rationale of adopting the reweighting strategy is that we ought to elevate the anomaly score when the nearest memory bank features to $f(x^{\text{test},*})$ and f^* are themselves far away from the surrounding samples. To design the anomaly map score for pixel-level anomaly localization, we simply compute the L_2 distances between the test patch features and the nearest patch features in $\mathcal{M}$, and realign them according to their respective spatial positions over the whole object. Then we resize the score map to the original ordered 3D data resolution $H \times W$ via Bilinear Interpolation $\mathcal{R}$ and apply KNN Gaussian Blurring $\mathcal{B}$ to the anomaly score map:

$$S_{\text{dist,map}}(X^{\text{test}}) = \mathcal{B}\left(\mathcal{R}\left(\square_{x^{\text{test}} \in \mathcal{X}^{\text{test}}} \min_{f \in \mathcal{M}} \|f(x^{\text{test}}) - f\|_2\right)\right), \quad (7)$$

Classification-based score with 3D input perturbation. We apply adversarial perturbation to the input patch xyz point cloud data based on the gradient of the negative log of the softmax score from the anticipated class, as determined by our trained classifier C_w in relation to the input patch. Mathematically, for any 3D point cloud patch x^{test},

$$\widetilde{x}^{\text{test}} = x^{\text{test}} + \eta(-\nabla_{x^{\text{test}}} \log(\widehat{p}(x^{\text{test}}|w))), \quad (8)$$

where $\widehat{p}(x^{\text{test}}|w) = \max\{p(x^{\text{test}}|w), 1 - p(x^{\text{test}}|w)\}$, and η is the perturbation magnitude. Given that C_w has been effectively trained to classify between the normal and pseudo anomalous patch, this approach seeks to lower the softmax score of the class predicted with the highest likelihood. This means it aims to make the abnormality harder to categorize with respect to the testing sample. We denote the 3D object and its ordered version after perturbation as $\widetilde{X}^{\text{test}}$ and $\widetilde{\mathcal{X}}^{\text{test}}$ respectively, *i.e.*, $\widetilde{\mathcal{X}}^{\text{test}} = \square_{x^{\text{test}} \in \mathcal{X}^{\text{test}}} \widetilde{x}^{\text{test}}$. We design the object-level classification-based score as

$$S_{\text{cls}}(\widetilde{X}^{\text{test}}) = \max_{x^{\text{test}} \in \widetilde{\mathcal{X}}^{\text{test}}} p(\widetilde{x}^{\text{test}}|w), \quad (9)$$

meaning that the abnormality extent of the whole object is decided by the most abnormal patch considered by C_w. The classification-based anomaly score map is designed by realigning the patch-level softmax probabilities based on their overall spatial locations and applying the same interpolation and blurring techniques as in our distance-based score map:

$$S_{\text{cls,map}}(\widetilde{X}^{\text{test}}) = \mathcal{B}\left(\mathcal{R}\left(\square_{x^{\text{test}} \in \widetilde{\mathcal{X}}^{\text{test}}} p(\widetilde{x}^{\text{test}}|w)\right)\right). \quad (10)$$

Final anomaly score. To design the final anomaly score, we combine both the distance-based score and the classification-based score after the adversarial perturbabtion. We denote the weight of the distance-based score as w_d and the weight of the classification-based score as w_c. The final pixel-level anomaly score map is computed as:

$$S_{\text{map}}(X^{\text{test}}) = w_d \cdot S_{\text{dist,map}}(X^{\text{test}}) + w_c \cdot S_{\text{cls,map}}(\widetilde{X}^{\text{test}}), \quad (11)$$

and the object-level final anomaly score is designed as

$$S(X^{\text{test}}) = w_d \cdot S_{\text{dist}}(X^{\text{test}}) + w_c \cdot S_{\text{cls}}(\widetilde{X}^{\text{test}}). \quad (12)$$

train\test	bagel	cable	carrot	cookie	dowel	foam	peach	potato	rope	tire	mean (3DzAL)	BTF	3DSR
bagel	-	78.6	91.9	89.3	81.6	48.3	91.2	96.5	82.0	86.7	**82.9** (↑ 2.4)	80.5	7.9
cable	31.5	-	87.1	48.6	81.0	56.6	65.4	89.4	81.8	80.7	**69.1** (↑ 0.9)	68.2	6.6
carrot	45.4	77.2	-	52.9	82.3	46.3	67.3	91.3	84.4	89.3	**70.7** (↑ 2.1)	68.6	21.7
cookie	70.6	76.8	91.5	-	81.0	46.5	82.9	91.7	84.4	89.2	**79.4** (↑ 5.5)	73.9	8.3
dowel	15.1	76.7	89.8	20.8	-	46.4	49.0	84.5	82.3	89.3	**61.5** (↑ 4.1)	57.4	35.4
foam	25.6	77.6	86.0	9.4	80.1	-	57.0	79.4	79.8	83.9	**64.3** (↑ 3.3)	61.0	0.5
peach	81.3	78.8	92.4	84.8	82.7	51.8	-	97.9	82.9	89.2	**82.4** (↑ 3.5)	78.9	14.2
potato	78.0	78.1	96.7	80.0	81.5	46.4	88.8	-	82.7	88.1	**80.0** (↑ 1.7)	78.3	13.2
rope	13.4	76.3	87.7	9.0	80.5	45.8	47.7	82.7	-	89.4	**59.2** (↑ 7.2)	52.0	19.3
tire	14.9	76.7	87.8	6.5	80.8	47.0	48.4	83.7	80.8	-	**58.5** (↑ 4.7)	53.8	23.8

Table 1. The detailed pixel-level AUPRO (%) of 3DzAL under the zero-shot setting. The best and second-best performances are highlighted in **bold** and underline (the gain of 3DzAL over the best baseline is also reported). 3DzAL outperforms BTF and 3DSR in all of the categories.

train\test	bagel	cable	carrot	cookie	dowel	foam	peach	potato	rope	tire	mean (3DzAL)	BTF	3DSR
bagel	-	57.9	71.8	68.9	57.5	58.1	56.1	69.0	47.3	55.9	**60.3** (↑ 6.9)	53.4	46.1
cable	52.5	-	52.3	49.6	51.6	72.4	46.4	48.2	44.2	59.9	**53.0** (↑ 2.9)	50.1	47.7
carrot	51.6	54.1	-	53.5	57.9	54.9	52.5	48.8	48.1	47.1	**52.1** (↑ 1.3)	50.8	46.0
cookie	40.2	50.0	55.5	-	55.3	60.1	46.0	47.3	35.9	59.6	50.0 (↓ 0.6)	49.5	**50.6**
dowel	54.9	57.0	42.4	51.8	-	57.6	50.4	55.9	58.9	50.4	**53.3** (↑ 2.0)	51.3	47.3
foam	60.9	48.8	46.7	47.0	52.7	-	49.0	48.5	50.0	62.4	51.8 (↓ 3.8)	49.9	**55.6**
peach	46.5	49.2	67.5	49.3	55.0	54.8	-	79.7	58.1	52.6	**57.0** (↑ 1.4)	55.6	45.0
potato	43.8	50.5	73.8	43.2	52.4	55.6	49.7	-	46.2	48.3	51.5 (↓ 0.6)	**52.1**	49.7
rope	48.8	47.6	54.6	42.0	41.9	52.8	46.7	45.7	-	61.8	49.1 (↓ 1.0)	49.1	**50.1**
tire	48.0	52.1	48.8	45.5	51.6	63.5	53.6	50.0	56.3	-	**52.2** (↑ 0.8)	50.5	51.4

Table 2. The detailed image-level AUROC of 3DzAL under the zero-shot setting. The best and second-best performances are highlighted in **bold** and underline (the gain of 3DzAL over the best baseline is also reported). 3DzAL outperforms BTF and 3DSR in most of the categories.

4. Experiments

Dataset. We conduct our zero-shot setting experiments on the MVTec 3D-AD dataset [3], which is the most commonly used 3D anomaly detection and localization dataset for industrial inspection. The dataset MVTec 3D-AD is a collection of high-resolution 3D models and corresponding 2D images. The dataset contains more than 800 3D models of everyday objects from 10 different classes.

Experimental setting. For our proposed zero-shot setting, we iteratively use the normal training data of one class for the training of 3DzAL , and then test on a different class. To ensure that the auxiliary data for pseudo anomaly generation is task-irrelevant, we adopt the leave-one-out strategy. Specifically, we use one class chosen from the remaining 9 classes for testing and the rest 8 classes as the task-irrelevant data in turn. There are 10×9=90 individual experiments in total.

Implementation details. For the RGB branch feature extraction, we use the Wide ResNet-50 [51] pre-trained on the ImageNet [13]. For all the ordered 3D data, we resize the original data resolution to H=W=224 and use the 8×8 patch size, so for each sample, we have 28×28=784 patches. For both E_θ and C_w we adapt the input resolution of PointNet++ [32] network architecture to 64. We use the percentage τ=0.1% when choosing pseudo abnormal points, and we choose the ratio of negative patch samples and positive patch samples as 16 : 1 for contrastive learning. We set the temperature T=0.07 in our contrastive learning paradigm. We set the weights of the loss functions as w_{con}=1, w_{rd}=100 to make the range of each loss comparable. The Adam optimizer [26] is employed for training. We train the two networks E_θ and C_w for 5 epochs and use the last-epoch model in the testing phase. At testing time, we set the nearest neighbor parameter b=3 and the perturbation magnitude η=0.1.

Baselines and evaluation metric. Since the problem of "zero-shot 3D anomaly localization" is defined by us, we are unable to identify alternative methods specifically designed for this setup. The most recent and closely related baselines we find are BTF [23] and 3DSR [53], which work under the classical setting, *i.e.*, the training and the testing class are the same. We adopt the commonly used evaluation metrics: pixel-level AUPRO and image-level AUROC [3, 23, 53]. 3DSR is the current best SOTA work under the classical setting, which achieves the highest image-level AUROC up to 0.978 and the highest pixel-level AUPRO up to 0.972 (mean taken over all the classes in the classical setting). We adapt the publicly released code of 3DSR [52] and BTF [22] to our zero-shot setting to report their results respectively for a fair comparison.

Result and analysis. We summarize the pixel-level AUPRO of 3DzAL and compare them with the mean results of BTF

baseline	$\mathcal{L}_{\text{con}}$	$\mathcal{L}_{\text{rd}}$	C_w	IP	P-AUPRO (%)	I-AUROC (%)
✓	✗	✗	✗	✗	80.5 / 57.4 / 61.0	53.4 / 51.3 / 49.9
✓	✓	✗	✗	✗	80.6 / 57.7 / 62.0	53.7 / 51.5 / 50.2
✓	✓	✓	✗	✗	80.8 / 57.8 / 62.5	54.0 / 51.8 / 50.5
✓	✓	✓	✓	✗	82.7 / 61.3 / 63.8	57.5 / 52.4 / 50.9
✓	✓	✓	✓	✓	**82.9 / 61.5 / 64.3**	**60.3 / 53.3 / 51.8**

Table 3. Ablation study of the 3DzAL components. IP denotes input perturbation, P-AUPRO denotes pixel-level AUPRO, and I-AUROC denotes image-level AUROC. For each cell, the numbers correspond to the cases when the training class is bagel/dowel/foam.

pseudo anomaly generation type		pixel-level	image-level
adding points	removing points	AUPRO (%)	AUROC (%)
✓	✗	80.3 / 79.3 / 58.8	53.8 / 52.6 / 49.1
✗	✓	81.0 / 79.6 / 58.9	54.0 / 52.8 / 49.1
✓	✓	**82.9 / 80.0 / 59.2**	**60.3 / 51.5 / 49.1**

Table 4. Ablation study of the pseudo anomaly generation type. For each cell, the numbers correspond to the cases when the training class is bagel/potato/rope.

and 3DSR in Tab. 1. Tab. 1 shows that 3DzAL outperforms BTF/3DSR in all the categories by a considerable margin, which shows the efficacy of 3DzAL . It also outperforms BTF/3DSR in **9/7 out of 10** categories in image-level AUROC, as shown in Tab. 2. Despite performing well in the classic setting, 3DSR performs particularly badly in anomaly localization in our zero-shot setting, which shows its poor generalization ability.

Comparison of the memory bank size and the model parameters size. As to the memory bank size after coreset selection, when trained on the class bagel and testing on the class cable gland, the size of the memory bank of baseline BTF is 229M (19129 patch features with dimension size 1569) and that of 3DzAL is 234M (19129 patch features with dimension size 1601). Therefore, 3DzAL has a slightly larger memory bank than BTF (because we have additionally concatenated learned 3D features for each memory bank patch-level feature), and the size ratio is the same for other training and testing cases. Although the memory bank size of 3DzAL increases 2.2% compared with BTF, the pixel-level AUPRO of 3DzAL improves over BTF by about 5.7% on average. For the model parameters, the model parameter size of 3DSR is 38M, and the model size of 3DzAL is 13.6M (6.8M for the learned 3D feature network and 6.8M for the normalcy classifier). Therefore, 3DzAL has nearly $\frac{1}{3}$ of the model parameters of 3DSR. BTF method does not involve model training.

Ablation study. We carry out ablation studies on the components of 3DzAL including the loss functions, normalcy classifier, and input perturbation. We conduct experiments on the 3 diverse training categories: bagel (round and relatively big), dowel (strip-shaped), and foam (irregular shape). Tab. 3 shows that $\mathcal{L}_{\text{con}}$,$\mathcal{L}_{\text{rd}}$, C_w, and input perturbation all enhance the performance. Next, we study the impact of the patch-level pseudo anomaly type on the performance by getting rid of the "adding-point" and "removing-point" type anomalies, and summarize the

method	pixel-level AUPRO (%)	image-level AUROC (%)
BTF (baseline)	80.5 / 78.9 / 52.0	53.4 / 55.6 / 49.1
3DzAL (pretrained CNN WI)	81.4 / 79.6 / 57.8	56.7 / 56.0 / 49.1
3DzAL (random CNN WI)	**82.9 / 82.4 / 59.2**	**60.3 / 57.0 / 49.1**

Table 5. Ablation study on the CNN weight initialization (WI) type. For each cell, the numbers correspond to the cases when the training class is bagel/peach/rope.

training classes \ method	BTF	3DzAL
bagel + cable	80.3 / 53.4	**80.9** (↑ 0.6) / **53.7** (↑ 0.3)
carrot + cookie	79.4 / 50.8	**83.7** (↑ 4.3) / **55.4** (↑ 4.6)
dowel + foam	78.5 / 51.3	**80.3** (↑ 1.8) / **54.0** (↑ 2.7)

Table 6. The performance of training on the normal data of the specified 2 classes, and testing on the rope class. For each cell, the first / second number is pixel-level AUPRO (%) / image-level AUROC(%). We also report the performance gain of 3DzAL over BTF.

results in Tab. 4, where both pseudo anomaly types contribute to 3D anomaly localization performance. Moreover, we use the ResNet-50 model pre-trained on the ImageNet dataset instead of random initialization in Tab. 5, where we conduct experiments to train on the class bagel/peach/rope, and all the other experimental settings and hyperparameters are kept the same. Tab. 5 shows that random initialization outperforms the pre-trained one for pseudo anomaly generation in our task. We hypothesize that it is because the ImageNet pretrained weights focus on discriminative areas for classification purposes, not necessarily the abnormal areas we want. Finally, we show the experiments when training on 2 classes and testing on another unseen class for both BTF and 3DzAL in Tab. 6, where 3DzAL consistently outperforms BTF in anomaly localization, which supports that 3DzAL can generalize to the multi-class setting.

5. Conclusion and Limitation

We have defined a new task for 3D anomaly localization and detection, which involves localizing anomalies in 3D point clouds for the target class that lacks training data. To address this challenge, we have proposed a novel framework named "3D zero-shot anomaly localization" (3DzAL) that aims to learn patch-level relative normalcy using contrastive learning and normalcy classification based on pseudo abnormal 3D patch generation. We are the first to show the efficacy of input perturbation in 3D anomaly detection and localization. 3DzAL surpasses the current state-of-the-art methods in 3D anomaly detection and localization. These promising results highlight the potential of using task-irrelevant data to generate pseudo anomalies as a viable approach for tackling the zero-shot 3D anomaly detection and localization problem. In addition, our new finding that a randomly initialized untrained neural network has the inductive bias to localize places of interest on 3D data can be potentially utilized as a prior for other tasks involving 3D data. The anomaly localization performance in some cases (*e.g.*, train on the class foam and test on the class cookie) is not high.

References

[1] Abhishek Aich, Kuan-Chuan Peng, and Amit K Roy-Chowdhury. Cross-domain video anomaly detection without target domain adaptation. In *WACV*, pages 2579–2591, 2023. 3, 4

[2] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. MVTec AD–a comprehensive real-world dataset for unsupervised anomaly detection. In *CVPR*, pages 9592–9600, 2019. 4

[3] Paul Bergmann., Xin Jin., David Sattlegger., and Carsten Steger. The MVTec 3D-AD dataset for unsupervised 3D anomaly detection and localization. In *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP,*, pages 202–213. INSTICC, SciTePress, 2022. 1, 2, 4, 7

[4] Paul Bergmann and David Sattlegger. Anomaly detection in 3D point clouds using deep geometric descriptors. In *WACV*, pages 2613–2623, 2023. 2

[5] Yunkang Cao, Xiaohao Xu, Zhaoge Liu, and Weiming Shen. Collaborative discrepancy optimization for reliable image anomaly localization. *IEEE Transactions on Industrial Informatics*, 2023. 1, 2

[6] Yunkang Cao, Xiaohao Xu, and Weiming Shen. Complementary pseudo multimodal feature for point cloud anomaly detection. *arXiv preprint arXiv:2303.13194*, 2023. 1

[7] Yun-Hao Cao and Jianxin Wu. A random CNN sees objects: One inductive bias of cnn and its applications. In *AAAI*, volume 36, pages 194–202, 2022. 4

[8] Ali Cheraghian, Shafin Rahman, Dylan Campbell, and Lars Petersson. Transductive zero-shot learning for 3d point cloud classification. In *WACV*, pages 923–933, 2020. 1

[9] Ali Cheraghian, Shafin Rahman, Townim F Chowdhury, Dylan Campbell, and Lars Petersson. Zero-shot learning on 3d point cloud objects and beyond. *IJCV*, 130(10):2364–2384, 2022. 1

[10] Yu-Min Chu, Chieh Liu, Ting-I Hsieh, Hwann-Tzong Chen, and Tyng-Luh Liu. Shape-guided dual-memory learning for 3d anomaly detection. pages 6185–6194. PMLR, 2023. 2

[11] Niv Cohen and Yedid Hoshen. Sub-image anomaly detection with deep pyramid correspondences. *arXiv preprint arXiv:2005.02357*, 2020. 3

[12] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. PaDiM: a patch distribution modeling framework for anomaly detection and localization. In *Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10–15, 2021, Proceedings, Part IV*, pages 475–489. Springer, 2021. 3

[13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. Ieee, 2009. 7

[14] Choubo Ding, Guansong Pang, and Chunhua Shen. Catching both gray and black swans: Open-set supervised anomaly detection. In *CVPR*, pages 7388–7398, 2022. 3

[15] Sepideh Esmaeilpour, Bing Liu, Eric Robertson, and Lei Shu. Zero-shot out-of-distribution detection based on the pre-trained model CLIP. In *AAAI*, volume 36, pages 6568–6576, 2022. 3

[16] Zhaoxin Fan, Hongyan Liu, Jun He, Qi Sun, and Xiaoyong Du. A graph-based one-shot learning method for point cloud recognition. In *Computer Graphics Forum*, volume 39, pages 313–323. Wiley Online Library, 2020. 1

[17] Zhaoxin Fan, Hongyan Liu, Jun He, Min Zhang, and Xiaoyong Du. Mpdnet: A 3d missing part detection network based on point cloud segmentation. In *ICASSP*, pages 1810–1814. IEEE, 2021. 1

[18] Richard HR Hahnloser, Rahul Sarpeshkar, Misha A Mahowald, Rodney J Douglas, and H Sebastian Seung. Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *nature*, 405(6789):947–951, 2000. 4

[19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, pages 16000–16009, 2022. 3

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*, pages 1026–1034, 2015. 4

[21] Chih-Hui Ho, Kuan-Chuan Peng, and Nuno Vasconcelos. Long-tailed anomaly detection with learnable class names. In *CVPR*, pages 12435–12446, 2024. 3

[22] Eliahu Horwitz and Yedid Hoshen. 3D-ADS. 2022. 7

[23] Eliahu Horwitz and Yedid Hoshen. Back to the feature: Classical 3d features are (almost) all you need for 3d anomaly detection. In *CVPRW*, pages 2967–2976, June 2023. 1, 2, 4, 7

[24] Chaoqin Huang, Haoyan Guan, Aofan Jiang, Ya Zhang, Michael Spratling, and Yan-Feng Wang. Registration based few-shot anomaly detection. In *ECCV*, pages 303–319. Springer, 2022. 3

[25] Jongheon Jeong, Yang Zou, Taewan Kim, Dongqing Zhang, Avinash Ravichandran, and Onkar Dabeer. Winclip: Zero-/few-shot anomaly classification and segmentation. In *CVPR*, pages 19606–19616, 2023. 3

[26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

[27] Philipp Liznerski, Lukas Ruff, Robert A. Vandermeulen, Billy Joe Franks, Marius Kloft, and Klaus Robert Muller. Explainable deep one-class classification. In *ICLR*, 2021. 3

[28] Philipp Liznerski, Lukas Ruff, Robert A. Vandermeulen, Billy Joe Franks, Klaus Robert Muller, and Marius Kloft. Exposing outlier exposure: What can be learned from few, one, and zero outlier images. *Transactions on Machine Learning Research*, 2022. 3

[29] Yiwei Lu, Frank Yu, Mahesh Kumar Krishna Reddy, and Yang Wang. Few-shot scene-adaptive anomaly detection. In *ECCV*, pages 125–141. Springer, 2020. 3

[30] Mana Masuda, Ryo Hachiuma, Ryo Fujii, Hideo Saito, and Yusuke Sekikawa. Toward unsupervised 3D point cloud anomaly detection using variational autoencoder. In *ICIP*, pages 3118–3122. IEEE, 2021. 1

[31] 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. 3

[32] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *NeurIPS*, 30, 2017. 5, 7

[33] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. pages 8748–8763. PMLR, 2021. 3

[34] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall

in industrial anomaly detection. In *CVPR*, pages 14318–14328, 2022. 2, 3, 4, 5

[35] Marco Rudolph, Tom Wehrbein, Bodo Rosenhahn, and Bastian Wandt. Asymmetric student-teacher networks for industrial anomaly detection. In *WACV*, pages 2592–2602, 2023. 1, 2

[36] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE international conference on robotics and automation*, pages 3212–3217. IEEE, 2009. 2, 4

[37] Eli Schwartz, Assaf Arbelle, Leonid Karlinsky, Sivan Harary, Florian Scheidegger, Sivan Doveh, and Raja Giryes. Maeday: Mae for few- and zero-shot anomaly-detection. *Computer Vision and Image Understanding*, 241:103958, 2024. 3

[38] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *ICLR*, 2018. 5

[39] Shelly Sheynin, Sagie Benaim, and Lior Wolf. A hierarchical transformation-discriminating generative model for few shot anomaly detection. In *ICCV*, pages 8495–8504, 2021. 3

[40] Samarth Sinha, Han Zhang, Anirudh Goyal, Yoshua Bengio, Hugo Larochelle, and Augustus Odena. Small-gan: Speeding up gan training using core-sets. pages 9005–9015. PMLR, 2020. 5

[41] Shashanka Venkataramanan, Kuan-Chuan Peng, Rajat Vikram Singh, and Abhijit Mahalanobis. Attention guided anomaly localization in images. In *ECCV*, pages 485–503. Springer, 2020. 3

[42] Qian Wan, Yunkang Cao, Liang Gao, Weiming Shen, and Xinyu Li. Position encoding enhanced feature mapping for image anomaly detection. In *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*, pages 876–881. IEEE, 2022. 1, 2

[43] Yizhou Wang, Dongliang Guo, Sheng Li, and Yun Fu. Towards explainable visual anomaly detection. *arXiv preprint arXiv:2302.06670*, 2023. 1

[44] Yue Wang, Jinlong Peng, Jiangning Zhang, Ran Yi, Yabiao Wang, and Chengjie Wang. Multimodal industrial anomaly detection via hybrid fusion. In *CVPR*, pages 8032–8041, 2023. 2

[45] Yizhou Wang, Can Qin, Yue Bai, Yi Xu, Xu Ma, and Yun Fu. Making reconstruction-based method great again for video anomaly detection. In *2022 IEEE International Conference on Data Mining (ICDM)*, pages 1215–1220. IEEE, 2022. 1

[46] Yizhou Wang, Can Qin, Rongzhe Wei, Yi Xu, Yue Bai, and Yun Fu. Self-supervision meets adversarial perturbation: A novel framework for anomaly detection. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 4555–4559, 2022. 1

[47] Yizhou Wang, Can Qin, Rongzhe Wei, Yi Xu, Yue Bai, and Yun Fu. Sla^{2}2p: Self-supervised anomaly detection with adversarial perturbation. *IEEE Transactions on Knowledge and Data Engineering*, 36(12):9282–9293, 2024. 1

[48] Jhih-Ciang Wu, Ding-Jie Chen, Chiou-Shann Fuh, and Tyng-Luh Liu. Learning unsupervised metaformer for anomaly detection. In *ICCV*, pages 4369–4378, 2021. 3

[49] Guoyang Xie, Jinbao Wang, Jiaqi Liu, Jiayi Lyu, Yong Liu, Chengjie Wang, Feng Zheng, and Yaochu Jin. IM-IAD: Industrial image anomaly detection benchmark in manufacturing. *arXiv preprint arXiv:2301.13359*, 2023. 1

[50] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *CVPR*, pages 2636–2645, 2020. 1

[51] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *BMVC*, 2016. 7

[52] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. 3DSR. 2024. 7

[53] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Cheating depth: Enhancing 3d surface anomaly detection via depth simulation. In *WACV*, pages 2164–2172, 2024. 2, 7

[54] Ye Zheng, Xiang Wang, Yu Qi, Wei Li, and Liwei Wu. Benchmarking unsupervised anomaly detection and localization. *arXiv preprint arXiv:2205.14852*, 2022. 1

[55] Kun Zhou, Qiming Hou, Rui Wang, and Baining Guo. Real-time kd-tree construction on graphics hardware. *ACM TOG*, 27(5):1–11, 2008. 4

[56] Qihang Zhou, Guansong Pang, Yu Tian, Shibo He, and Jiming Chen. Anomalyclip: Object-agnostic prompt learning for zero-shot anomaly detection. *arXiv preprint arXiv:2310.18961*, 2023. 3