Towards Scalable 3D Anomaly Detection and Localization: A Benchmark via 3D Anomaly Synthesis and A Self-Supervised Learning Network

Wenqiao Li[1]   Xiaohao Xu[2*]   Yao Gu[1*]   Bozhong Zheng[1]   Shenghua Gao[1†]   Yingna Wu[1†]

[1]ShanghaiTech University   [2]University of Michigan, Ann Arbor

{liwq2022}@shanghaitech.edu.cn

Abstract

*Recently, 3D anomaly detection, a crucial problem involving fine-grained geometry discrimination, is getting more attention. However, the lack of abundant real 3D anomaly data limits the scalability of current models. To enable scalable anomaly data collection, we propose a 3D anomaly synthesis pipeline to adapt existing large-scale 3D models for 3D anomaly detection. Specifically, we construct a synthetic dataset, i.e., **Anomaly-ShapeNet**, based on ShapeNet. Anomaly-ShapeNet consists of 1600 point cloud samples under 40 categories, which provides a rich and varied collection of data, enabling efficient training and enhancing adaptability to industrial scenarios. Meanwhile, to enable scalable representation learning for 3D anomaly localization, we propose a self-supervised method, i.e., Iterative Mask Reconstruction Network (**IMRNet**). During training, we propose a geometry-aware sample module to preserve potentially anomalous local regions during point cloud down-sampling. Then, we randomly mask out point patches and sent the visible patches to a transformer for reconstruction-based self-supervision. During testing, the point cloud repeatedly goes through the Mask Reconstruction Network, with each iteration's output becoming the next input. By merging and contrasting the final reconstructed point cloud with the initial input, our method successfully locates anomalies. Experiments show that IMRNet outperforms previous state-of-the-art methods, achieving 66.1% in I-AUC on our Anomaly-ShapeNet dataset and 72.5% in I-AUC on Real3D-AD dataset. Our benchmark will be released at https://github.com/Chopper-233/Anomaly-ShapeNet.*

1. Introduction

The 3D anomaly detection task is crucial in object-centered industrial quality inspection, where high accuracy
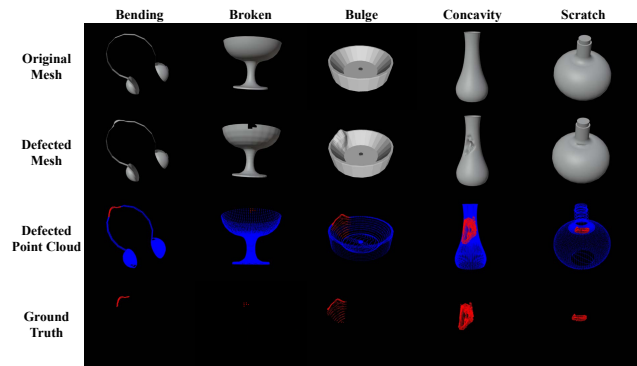

Figure 1. **Examples of the proposed Anomaly-ShapeNet**. The first and second rows are the original mesh and the subdivided mesh. The third row is synthetic defected point cloud. The fourth row is the Ground-Truth of the anomalous region.

and detection rates are often required. The goal is to identify anomaly regions and locate abnormal point clouds within the 3D context. Image-based anomaly detection algorithms under fixed perspectives [1,7,15–17,19,24,26,32–35,35,43, 46] have limitations due to blind spots and do not perform as desired in object-centered scenarios. Consequently, researchers have increasingly focused on 3D information for anomaly detection [2].

With advancements in 3D sensor technology, several datasets such as MVTecAD-3D [2], Real3D-AD [25], and MAD [45] have been created to meet the increasing demand for 3D anomaly detection. MVTecAD-3D is designed for anomaly detection in scenarios with a single-angle camera, while the MAD dataset focuses on multi-pose anomaly detection. Only Real3D-AD specifically addresses anomaly detection on complete point clouds. Since 3D point clouds obtained from a 3D scanner generally contain more morphological information compared to data from multiple cameras, our primary objective is to advance the field of point cloud-based anomaly detection tasks.

In the field of point cloud anomaly detection, there are two main issues that need to be addressed: the lack of diverse distribution datasets and the need for more ef-

fective deep learning-based approaches. Firstly, the current high-quality real-world 3D point anomaly detection dataset, Real3D-AD, has limitations in terms of the variety of normal and abnormal samples and the excessive variation in point sizes. These factors make it challenging to apply reconstruction-based [3] or deep learning feature extraction-based [1] algorithms widely. Secondly, the prevailing approaches for point cloud anomaly detection heavily rely on traditional feature processing operators and models that are pretrained to extract features or transform data to 2D for processing [1, 8]. This approach fails to fully exploit the potential of point cloud data and leads to significant feature domain misalignment.

To enable the scalable data synthesis [42], we create the Anomaly-ShapeNet dataset by synthesizing data based on ShapeNet [12], which is a widely used dataset for point cloud processing. The Anomaly-ShapeNet dataset consists of 1600 samples belonging to 40 classes. It includes six typical anomaly types: bulge, concavity, hole, break, bending, and crack. The number of points in each sample ranges from 8000 to 30,000. The anomalous portion constitutes between 1% and 10% of the entire point cloud. Importantly, our synthetic anomaly dataset contains realistic and diverse examples of anomalies. Figure 1 showcases some examples from the Anomaly-ShapeNet dataset.

To address the challenges in 3D point cloud anomaly detection, we propose a self-supervised method called IMRNet, which is based on iterative mask reconstruction. IMRNet comprises three main components: (1) the Geometry-aware Point-cloud Sampling module (GPS), (2) the Point-Patch Mask Reconstruction network (PMR), and (3) the Dense Feature Concatenation and Comparison module (DFC). In the IMRNet framework, before passing the anomalous point clouds through the reconstruction network, we employ the GPS module to extract keypoints based on the geometry structure of the point clouds. This ensures that we sample the anomaly points as comprehensively as possible. These sampled keypoints are then transformed into point-patch groups using the K-nearest neighbor (KNN) operation and sent to the PMR network in an iterative manner. The PMR network performs the mask reconstruction process on these point-patch groups. After the PMR network, both the reconstructed normal sample and the input point cloud are sent to the DFC module. The DFC module compares the features and points of both the reconstructed normal sample and the input point cloud to obtain the anomaly score. This comparison is performed in both the feature space and the point space, enabling a comprehensive evaluation of the anomaly. Overall, the IMRNet method combines the GPS module, the PMR network, and the DFC module to achieve effective self-supervised 3D point cloud anomaly detection through iterative mask reconstruction and feature-based comparison.

By combining the Anomaly-ShapeNet dataset and the IMRNet method, our work aims to contribute to the field of scalable 3D anomaly detection. We summarize our contributions as follows:

- We present a diverse and high-quality synthetic 3D anomaly dataset, Anomaly-ShapeNet, with accurate 3D annotations.

- We propose the IMRNet, a novel 3D point-cloud anomaly detection method, with a Geometry-aware Point-cloud Sample module, an iterative Point-patch Mask Reconstruction network and a dense feature comparison module, which outperforms the SoTA methods on Anomaly-ShapeNet and Real3D-AD.

- We demonstrate that the proposed Geometry-aware Point-cloud Sampling module can help extract important anomaly points in 3D point clouds more effectively and the proposed PMR network can learn better representation of the 3D anomaly datasets.

2. Related works

2D anomaly detection. Recent advances in unsupervised anomaly detection [10] for two-dimensional data, such as RGB or grayscale images, have shown significant progress [18, 27, 40]. Traditional approaches often rely on autoencoders [3, 20, 38] and generative adversarial networks (GANs) [11, 29, 37], typically employing random weight initialization. Alternatively, methods leveraging pretrained network descriptors for anomaly detection often surpass those trained from scratch [1, 6, 7, 15–17, 19, 24, 26, 32–35, 35, 36, 41, 43, 46]. Noteworthy examples include Self-Taught anomaly detection (ST) [1], which aligns features from a pre-trained teacher network with a randomly initialized student network to identify anomalies, PatchCore [35] that utilizes a memory-bank approach for modeling normal data distributions, and Cutpaste [24], introducing a novel data augmentation strategy for self-supervised anomaly detection. The emergence of large-scale foundation models [23, 31] has spurred new methodologies that capitalize on the robust zero-shot generalization capabilities of these models for anomaly detection tasks [9, 14, 22]. In our work, we extend the principles of reconstruction-based self-supervised learning from 2D to 3D contexts. We introduce a novel self-supervised network employing a masked reconstruction mechanism to advance scalable 3D anomaly detection.

3D anomaly detection. The domain of 3D anomaly detection remains less advanced compared to its 2D counterpart, hindered by intrinsic challenges such as data sparsity, increased dimensionality, and prevalent noise. Bergmann *et al.* [4] introduced a seminal approach wherein a teacher-student network paradigm is employed during training to

Datasets	Year	Type	Modality	#Class	#Anomaly Types	Number	Point Range
MVTecAD-3D [2]	2021	Real	RGB/D	10	3-5	3604	$10K$-$30K$
Eyecandies [5]	2022	Syn	RGB/D/N	10	3	15500	-
MAD [45]	2023	Syn+Real	RGB	20	3	10133	-
Real3D-AD [25]	2023	Real	Point Cloud	12	2	1200	$35K$-$780K$
Anomaly-ShapeNet (Ours)	2023	Syn	Point Cloud	40	6	1600	$8K$-$30K$

Table 1. **Comparison between the proposed Anomaly-ShapeNet and existing mainstream 3D anomaly detection datasets.**

align point-cloud features, which are then compared in testing to identify anomalies. Horwitz *et al.* [21] leveraged classical 3D descriptors coupled with K-Nearest Neighbors (KNN) for anomaly detection. Although AST [36] demonstrated efficacy in certain scenarios, its primary focus on background suppression via depth information led to the omission of finer detail anomalies. The M3DM framework [39] innovatively combines 3D point data with conventional imaging for enhanced decision-making. CPMF [8] introduced a novel methodology that integrates a memory bank approach with KNN and enriches the detection process by rendering 3D data into multi-view 2D images. Conversely, EasyNet [13] presents a straightforward mechanism for 3D anomaly detection, circumventing the need for pre-training. Nonetheless, the scarcity of robust 3D anomaly detection datasets [2, 25, 45] constrains the scalability and generalizability of these models across varied 3D anomaly detection contexts. In light of this, our work aims to devise a 3D anomaly synthesis pipeline, enhancing the volume and diversity of data necessary for the development of more generalized 3D anomaly detection models.

3. Anomaly-ShapeNet 3D Dataset Synthesis

Pipeline overview. As is shown in Figure 2, the overall pipeline for constructing our Anomaly-ShapeNet dataset consists of three components: mesh subdivision, defects carving, and ground truth generation.

Normal data sampling. To generate our dataset's original normal samples, we utilized the ShapeNet dataset [12], which is renowned for its diversity and high quality and is commonly employed in tasks such as point cloud segmentation, completion, and classification. For our data source, we specifically selected normal samples from the sub-dataset ShapeNetcoreV2 of the ShapeNet.

Anomaly data synthesis. We developed a point cloud refinement module to enhance the limited number of points and faces found in certain point clouds from the ShapeNet dataset. To introduce more realistic defects, we utilized *Blender*, a widely-used software in the industrial design domain to sculpt various defects. Blender is an open-source industrial design software offering extensive features such as sculpting, refining, cropping, and various editing modes, which contributes to its popularity in this field. After acquir-
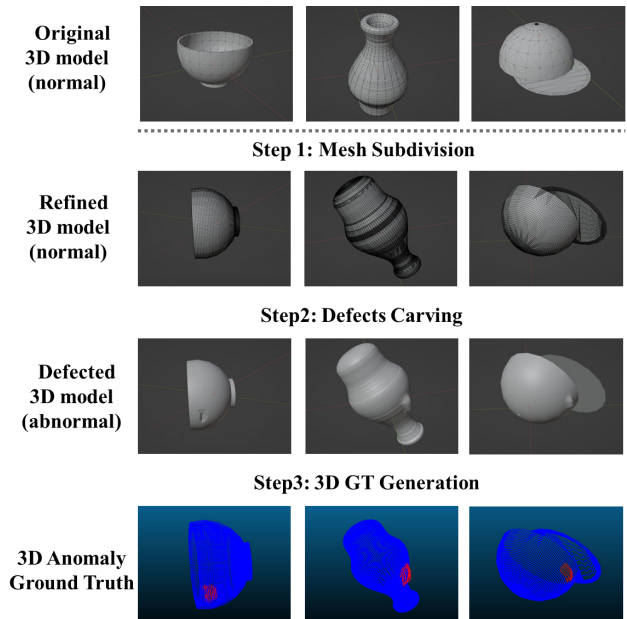

Figure 2. **Pipeline for anomaly synthesis built upon the Anomaly-ShapeNet Dataset.** Selected normal samples are processed through a mesh subdivision module to attain a more uniform point cloud distribution. We employ *Blender* to introduce defects into the refined samples, and utilize *CloudCompare* software to acquire 3D anomaly ground truth.

ing the abnormal samples, we employed *CloudCompare*, point editing software, to obtain the ground truths.

Dataset statistics. Anomaly-ShapeNet comprises training folders with four normal samples per class and testing folders with 28 to 40 samples, including both normal and abnormal instances. Sample point clouds range from 8,000 to 30,000 points, compatible with prevalent point cloud models and minimizing the need for down-sampling. This dataset provides a broader category variety and a more application-friendly number of points compared to Real3D-AD [39], the sole 3D point cloud anomaly detection dataset from 3D scans. It expands the scope of Real3D-AD, offering increased diversity and a wider range of anomaly types. It aligns with the data volume preferred by previous algorithms, thereby facilitating advancements in anomaly detection and reducing computational overhead. The comparison with leading 3D anomaly datasets is detailed in Table 1.
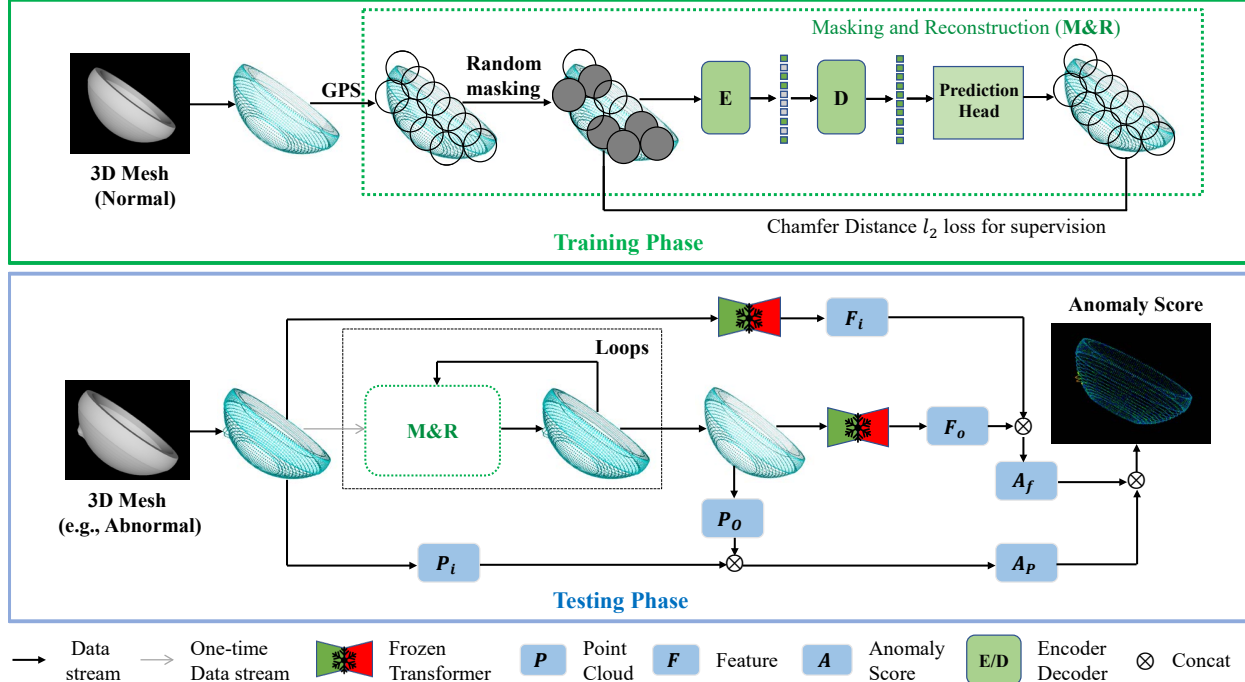
Figure 3. **Overview of the Iterative Mask Reconstruction Network (IMRNet) Pipeline.** (a) *Training Phase:* The standard training point cloud is initially converted to point-patch format using the Geometry-aware Point-cloud Sampling (GPS) module. Following this, random masking is applied to the point-patches, which are then reconstructed by a network comprising an Autoencoder-based Transformer and a lightweight prediction head, operating in a self-supervised paradigm. (b) *Testing Phase:* The input point cloud is subjected to a reconstruction process mirroring the training procedure. The reconstructed point cloud is cyclically fed back into the reconstruction network as input for several iterations. Ultimately, a comparative analysis is performed between the reconstructed and the original point clouds at both the point cloud and feature levels to derive the final anomaly score map.

4. Self-Supervised Representation Learning for 3D Anomaly Detection: IMRNet

Figure 3 shows the overall architecture of our IMRNet. Our IMRNet, which consists of three modules: GPS, PMR, and DFC, successfully detect and locate the anomaly in the abnormal samples. The details of each module will be illustrated in the following sections.

4.1. Geometry-aware Point Cloud Sampling

In point cloud processing, uniform sampling or farthest point sampling is commonly used. However, when it comes to point cloud anomaly detection, arbitrary sampling methods can lead to ambiguous representations of the anomaly structures. In our geometry aware sampling module, we address this issue by first calculating the geometry features of the points to adaptively sample the point cloud.

Geometry feature extraction. Given a point cloud (P) as input, we define the set of neighboring points of a certain point P_i within a radius r as $\mathcal{N}_i$ (where $|\mathcal{N}_i| = N$). To compute the normal vector $\mathbf{N}_i$ for each point (P_i), we employ a local surface fitting method. The normal vector $\mathbf{N}_i$

can be obtained by solving the following equation:

$$\min_{\mathcal{N}_i} \sum_{P_j \in \mathcal{N}_i} |\mathbf{N}_i \cdot \mathbf{v}_j - d|^2 \quad (1)$$

where ($\mathbf{v}_j$) is the vector from P_i to P_j, $\mathbf{N}_i \cdot \mathbf{v}_j$ denotes the dot product between the normal vector and the vector $\mathbf{v}_j$, and d represents the signed distance from the plane defined by the normal vector $\mathbf{N}_i$ to the point P_j.

The curvature of a point can be calculated using the normal vector and the eigenvalues of the covariance matrix $\mathbf{C}_i$ of its neighboring points $\mathcal{N}_i$. The curvature value (K_i) is given by:

$$K_i = \frac{\min(\lambda_1, \lambda_2)}{\lambda_1 + \lambda_2 + \epsilon} \quad (2)$$

where (λ_1) and (λ_2) are the eigenvalues of $\mathbf{C}_i$ and ϵ is a small positive constant to avoid division by zero.

To quantify the rate of change in normal vectors and curvature between two points (P_i and P_j), we can define the following formulas:

$$R_{\text{norm}}(P_i, P_j) = \frac{|\mathbf{N}_i - \mathbf{N}j|}{|\mathbf{v}_{ij}|} \quad (3)$$

$$R_{\text{curv}}(P_i, P_j) = |K_i - K_j| \quad (4)$$

where $\mathbf{v}_{ij}$ is the vector from P_i to P_j, $\mathbf{N}_i$ and $\mathbf{N}_j$ are the normal vectors at each point, and K_i and K_j are the curvature values at each point.

Geometry-aware sampling. During the sampling process of the input point cloud, we employ a geometry-aware point cloud sampling approach. To achieve this, we introduce a rate of change memory bank $\mathcal{M}$ that captures the local and global rates of change in the point cloud. For each point P_i, the memory bank stores the rate of change value R_i computed based on the differences in normal vectors and curvature values between neighboring points. Given the rate of change in normal vectors and curvature (R_{norm} and R_{curv}), the overall rate of change value R is computed as the average of the rate of change values with its neighboring points:

$$R_i = \frac{1}{|\mathcal{N}_i|} \sum_{P_j \in N_i} \left(R_{\text{norm}}(P_i, P_j) + R_{\text{curv}}(P_i, P_j) \right) \quad (5)$$

To prioritize high-rate-of-change points, we sort them based on values in the memory bank $\mathcal{M}$. Lower ranks indicate higher rates of change, capturing significant anomalous structures. We select points with greater rates of change using a threshold τ. By sampling points with ranks from 1 to $\lfloor \tau \cdot N \rfloor$, where N is the total point count, we capture more points with significant rates of change. The sampling process to derive the final sampled point set ($\mathcal{S}$) can be represented by the following equation:

$$\mathcal{S} = \{P_k | \text{Rank}(P_k) \leq \lfloor \tau \cdot N \rfloor\} \quad (6)$$

By incorporating this geometry-aware point cloud sampling strategy, we enhance the accuracy and effectiveness of point cloud anomaly detection by ensuring that the sampled points represent the underlying anomaly structures with their distinctive geometric characteristics.

4.2. Iterative Mask Reconstruction

In 2D anomaly detection field, splitting normal images into patches of the same size and randomly masking them before reconstructing the masked patches back to normal is common. The underlying idea is the reconstruction network only receives the normal patches and learns only the normal feature distribution during training. During testing, input anomalous images are masked and can be restored to normal images. Different from images, point cloud can not be divided into regular patches like images. Furthermore, due to the unordered nature of point clouds, the reconstruction error cannot be directly computed using mean squared error (MSE) or structural similarity (SSIM). Based on these properties, we propose the PMR module, which mainly consists of three components: point-patch generation, random masking and embedding, and reconstruction target.

Point-patch generation. Following Point-MAE [28], the input point cloud is divided into irregular overlapping patches via our Geometry-aware Point Sampling (GPS) and K-Nearest Neighborhood (KNN). Formally, given an point cloud P_i with N points, $P_i \in \mathbb{R}^{N \times 3}$, GPS is applied to sample C points as the patch centers. Based on C, KNN selects k nearest points around each center and formulate the point patches P. C and P are formulated as follows:

$$C = GPS(P_i), C \in \mathbb{R}^{n \times 3} \quad (7)$$

$$P = KNN(P_i, C), P \in \mathbb{R}^{n \times k \times 3} \quad (8)$$

It should be noted that, in our point patches, each center point represents its neighborhood like 2D patchcore [35]. This not only leads to better convergence but also facilitates the detection and localization of anomalies.

Random masking and embedding. With a masking ratio m at 40%, we randomly select a set of masked patches $M \in \mathbb{R}^{mn \times k \times 3}$, which is also used as the ground truth for reconstruction. After ramdomly masking, the visible points could be illustrated as:

$$P_{vis} = P \odot (1 - M) \quad (9)$$

where $\odot$ denotes the spatial element-wise product.

To transform the visible patches and masked patches to tokens, we implement the PointNet [30], which mainly consists of MLPs and max pooling layers. Considering our point patches are represented by their coordinates, a simple Position Embedding (PE) map the centers C to the embedding P_c. Setting the dimension as d, the visible tokens T_v is defined as:

$$T_{vis} = PointNet(P_{vis}), T_{vis} \in \mathbb{R}^{(1-m)n \times d} \quad (10)$$

Reconstruction target. Our reconstruction backbone is entirely based on Point-Transformer(PT) [44], with an asymmetric encoder-decoder architecture. In order to predict the masked points, we add a prediction head with a simple fully connected layer to the last layer of the decoder.

During the training phase, both the visible tokens T_{vis} and the mask tokens T_m are sent to the Transformer PT with the global position embedding P_c. At the last layer of the decoder, the prediction head F_c tries to output the reconstructed points P_{pre}. The process can be expressed as follows:

$$P_{pre} = F_c\{PT(T_{vis}, T_m, P_c)\}, P_{pre} \in \mathbb{R}^{mn \times k \times 3} \quad (11)$$

The target of our reconstruction net is to restore the masked point patches M, also called P_{gt}. After obtaining the predicted point patches P_{pre} and the ground truth P_{gt}, we use l_2 Chamfer Distance as our reconstruction loss.

$$L = \frac{1}{|P_{pre}|} \sum_{a \in P_{pre}} \min_{b \in P_{gt}} \|a-b\|_2^2 + \frac{1}{|P_{gt}|} \sum_{b \in P_{gt}} \min_{a \in P_{pre}} \|a-b\|_2^2 \quad (12)$$

Algorithm 1 Iterative Mask Reconstruction

Require: Point Transformer PT, Prediction head F_c, Geometry feature extractor G, Feature related sample function S, masking ratio m, iteration number I, data loader $\mathcal{D}$

1: Load model PT
2: Set PT to evaluation mode
3: **for** each P^* in $\mathcal{D}$ **do**
4:     $(P, C) \leftarrow \text{GPS}(P^*)$
5:     **for** index in 1 to I **do**
6:         $P_{\text{vis}} \leftarrow P(1-m)$
7:         $T \leftarrow PT(P_{\text{vis}}, C)$
8:         $P_{pre} \leftarrow F_c(T, C)$
9:         $P = P_{pre} \oplus P_{vis}$
10:     **end for**
11: **end for**
12: **function** GPS(P)
13:     $M \leftarrow G(P)$
14:     $C \leftarrow S(M, P)$
15:     $P \leftarrow \text{KNN}(C, P, k)$
16:     **return** (P, C)
17: **end function**

During testing, each restored point-patches will be concatenated with the visible point-patches and sent back to the Transformer for a few times until the anomaly parts have been masked and restored to normal surface. Given the anomaly samples as P^*, the iterative reconstruction pseudocode is shown in Algorithm 1.

4.3. Dense feature concatenation and comparision

Following the iterative reconstruction module, the anomaly score is determined by comparing the reconstructed point cloud with the original input. To enhance detection of subtle anomalies, we combine features from both the input and reconstructed point clouds. Additionally, to reduce false positives caused by excessive iterations in normal point cloud generation, we employ a single template from the training phase to regularize the output.

Point-patch comparison. Leveraging the point-patch subdivision approach inspired by PatchCore [35], we address the challenges of the unordered nature of point clouds by facilitating patch-wise comparison between the reconstructed and original point clouds. Each point's anomaly score is derived by transforming the Chamfer distance-based comparison scores of its corresponding patches, computed directly from the training phase. The comparison process is formalized as:

$$P_i = KNN(p_i, k), P_i \in \mathbb{R}^{n \times k \times 3} \tag{13}$$

$$P_o = KNN(p_o, k), P_o \in \mathbb{R}^{n \times k \times 3} \tag{14}$$

$$A_p = \mathcal{L}_{\text{c}}(P_i, P_o) \tag{15}$$

where p_i and p_o denote the input and output point clouds, respectively, with n representing the aggregation count of neighboring points around each point. P_i and P_o are the corresponding point-patches. The Chamfer loss is given by $\mathcal{L}_{\text{c}}$, and A_p is the anomaly score in the point domain.

Feature fusion and comparision. It is commonly recognized that features extracted from different layers of a neural network represent information at varying levels. Consequently, the fusion of information from different layers can effectively enhance the capability of anomaly detection. In our DFC (Dense Feature Concatenation and Comparison) module, we utilize the decoder of a transformer previously employed for reconstruction as a feature extractor. Following the decoding of input and reconstructed point clouds, we extract their 1st, 2nd, and 3rd layer features (f^1, f^2, f^3) and subsequently fuse and compare them, resulting in feature-level anomaly scores. Feature level anomaly scores A_f are formulated as follows:

$$f^1, f^2, f^3 = \phi(p) \tag{16}$$

$$F = f^1 \oplus f^2 \oplus f^3 \tag{17}$$

$$A_f = F_i \Theta F_o \tag{18}$$

where p are input and reconstructed point clouds, and f are the extracted features. $\oplus$ represents the fusion operation and F are the fused features. Θ is the comparision operation and A_f is the feature anomaly score.

Template regularization. Excessively iterative reconstruction may induce "normal point drift", potentially increasing the positive false rate. To mitigate this, we employ a feature template T_f saved during the training phase to regularize the features F_o of our reconstructed point cloud. For each vector z_i within F_o, we compute its distance to the template's corresponding vector z_i and save the distance to a memory bank M.

$$\forall z_i \in \mathcal{F}_{\mathcal{O}}, M = ||\hat{z}_i^{(l)} - z_i^{(l)}|| \tag{19}$$

By Setting a distance threshold τ, we access each distance d_i within the set M. If d_i exceeds τ, we replace the corresponding vector z_i with $\hat{z}_i$. If using template regularization, the regularized F_o will be used for calculating A_f.

After obtaining the anomaly score A_p and A_f, we interpolate them to a uniform dimension. The final anomaly score is the result of concatenating two scores.

$$A = A_p \oplus A_f \tag{20}$$

Method	Airplane	Car	Candy	Chicken	Diamond	Duck	Fish	Gemstone	Seahorse	Shell	Starfish	Toffees	Mean
BTF(Raw) [21]	0.730	0.647	0.539	0.789	0.707	**0.691**	0.602	**0.686**	0.596	0.396	0.530	0.703	0.635
BTF(FPFH)	0.520	0.560	0.630	0.432	0.545	0.784	0.549	0.648	**0.779**	**0.754**	0.575	0.462	0.603
M3DM [39]	0.434	0.541	0.552	0.683	0.602	0.433	0.540	0.644	0.495	0.694	0.551	0.450	0.552
PatchCore(FPFH) [35]	**0.882**	0.590	0.541	0.837	0.574	0.546	0.675	0.370	0.505	0.589	0.441	0.565	0.593
PatchCore(PointMAE)	0.726	0.498	0.663	0.827	0.783	0.489	0.630	0.374	0.539	0.501	0.519	0.585	0.594
CPMF [8]	0.701	0.551	0.552	0.504	0.523	0.582	0.558	0.589	0.729	0.653	**0.700**	0.390	0.586
RegAD [25]	0.716	0.697	0.685	**0.852**	0.900	0.584	**0.915**	0.417	0.762	0.583	0.506	0.827	0.704
Ours	0.762	**0.711**	**0.755**	0.780	**0.905**	0.517	0.880	0.674	0.604	0.665	0.674	0.774	**0.725**

Table 2. **I-AUROC score for anomaly detection of 12 categories of Real3D-AD**. Bold numbers represent the current highest metrics. Our method clearly outperforms the baseline; For pure 3D point setting, we get 0.725 mean I-AUROC score.

Method	cap0	cap3	helmet3	cup0	bowl4	vase3	headset1	eraser0	vase8	cap4	vase2	vase4	helmet0	bucket1
BTF(Raw) [21]	0.668	0.527	0.526	0.403	0.664	**0.717**	0.515	0.525	0.424	0.468	0.410	0.425	0.553	0.321
BTF(FPFH)	0.618	0.522	0.444	0.586	0.609	0.699	0.490	**0.719**	**0.668**	0.520	0.546	0.510	0.571	0.633
M3DM [39]	0.557	0.423	0.374	0.539	0.464	0.439	0.617	0.627	0.663	**0.777**	0.737	0.476	0.526	0.501
Patchcore(FPFH) [35]	0.580	0.453	0.404	0.600	0.494	0.449	0.637	0.657	0.662	0.757	0.721	0.506	0.546	0.551
Patchcore(PointMAE)	0.589	0.476	0.424	0.610	0.501	0.460	0.627	0.677	0.663	0.727	**0.741**	0.516	0.556	0.561
CPMF [8]	0.601	0.551	0.520	0.497	**0.683**	0.582	0.458	0.689	0.529	0.553	0.582	0.514	0.555	0.601
RegAD [25]	0.693	0.725	0.367	0.510	0.663	0.650	0.610	0.343	0.620	0.643	0.605	0.500	**0.600**	0.752
Ours	**0.737**	**0.775**	**0.573**	**0.643**	0.676	0.700	0.676	0.548	0.630	0.652	0.614	**0.524**	0.597	**0.771**

Method	bottle3	vase0	bottle0	tap1	bowl0	bucket0	vase5	vase1	vase9	ashtray0	bottle1	tap0	phone	cup1
BTF(Raw) [21]	0.568	0.531	0.597	0.573	0.564	**0.617**	0.585	0.549	0.564	0.578	0.510	0.525	0.563	0.521
BTF(FPFH)	0.322	0.342	0.344	0.546	0.509	0.401	0.409	0.219	0.268	0.420	0.546	0.560	0.671	0.610
M3DM [39]	0.541	0.423	0.574	0.739	0.634	0.309	0.317	0.427	**0.663**	0.577	0.637	**0.754**	0.357	0.556
Patchcore(FPFH) [35]	0.572	0.455	**0.604**	**0.766**	0.504	0.469	0.417	0.423	0.660	0.587	0.667	0.753	0.388	0.586
Patchcore(PointMAE)	**0.650**	0.447	0.513	0.538	0.523	0.593	0.579	0.552	0.629	0.591	0.601	0.458	0.488	0.556
CPMF [8]	0.405	0.451	0.520	0.697	**0.783**	0.482	0.618	0.345	0.609	0.353	0.482	0.359	0.509	0.499
RegAD [25]	0.525	0.533	0.486	0.641	0.671	0.610	0.520	0.702	0.594	0.597	0.695	0.676	0.414	0.538
Ours	0.640	**0.533**	0.552	0.696	0.681	0.580	**0.676**	**0.757**	0.594	**0.671**	**0.700**	0.676	**0.755**	**0.757**

Method	vase7	helmet2	cap5	shelf0	bowl5	bowl3	helmet1	bowl1	headset0	bag0	bowl2	jar	Mean
BTF(Raw) [21]	0.448	0.602	0.373	0.164	0.417	0.385	0.349	0.264	0.378	0.410	0.525	0.420	0.493
BTF(FPFH)	0.518	0.542	0.586	0.609	0.699	0.490	**0.719**	0.668	0.520	0.546	0.510	0.424	0.528
M3DM [39]	0.657	0.623	0.639	0.564	0.409	0.617	0.427	0.663	0.577	0.537	0.684	0.441	0.552
Patchcore(FPFH) [35]	**0.693**	0.425	**0.790**	0.494	0.558	0.537	0.484	0.639	0.583	0.571	0.615	0.472	0.568
Patchcore(PointMAE)	0.650	0.447	0.538	0.523	0.593	0.579	0.552	0.629	0.591	0.601	0.458	0.483	0.562
CPMF [8]	0.397	0.462	0.697	0.685	0.685	**0.658**	0.589	0.639	0.643	0.643	0.625	0.610	0.559
RegAD [25]	0.462	0.614	0.467	**0.688**	0.593	0.348	0.381	0.525	0.537	**0.706**	0.490	0.592	0.572
Ours	0.635	**0.641**	0.652	0.603	**0.710**	0.599	0.600	**0.702**	**0.720**	0.660	**0.685**	**0.780**	**0.661**

Table 3. **I-AUROC score for anomaly detection of 40 categories of our Anomaly-ShapeNet dataset**. Our method clearly outperforms other methods. Last line is the average result of 40 classes. The results can be regarded as the baseline of Anomaly-ShapeNet.
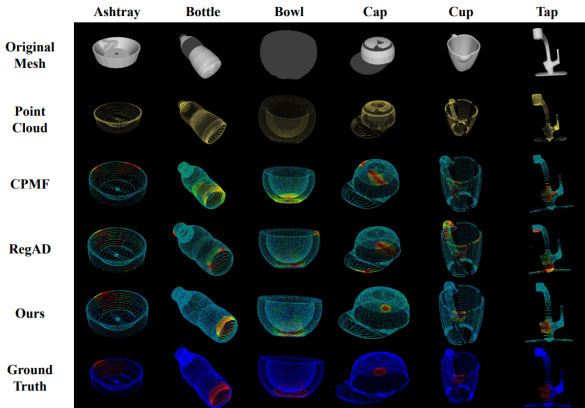

Figure 4. **Qualitative results visualization of anomaly localization performance on Anomaly-ShapeNet.**

5. Experiments

5.1. Datasets

Anomaly-ShapeNet. The Anomaly-ShapeNet is our newly proposed 3D synthesised point cloud anomaly detection dataset. The Anomaly-ShapeNet dataset offers 40 categories, with around 1600 positive and negative samples. Each training set for a category contains only four samples, which is similar to the few-shot scenario. Each test set for a category contains normal and various defected samples. Meanwhile, we evaluated M3DM [39], RegAD [25], and other methods on our designed dataset, providing a benchmark for future researchers.

Real3D-AD. Real3D-AD [25] is a new large-scale 3D point anomaly dataset captured with the PMAX-S130 high-resolution binocular 3D scanner. It contains 1,254 samples across 12 categories and includes the Reg-AD as a baseline.

5.2. Evaluation metrics

Image-level anomaly detection is measured using I-AUROC (Area Under the Receiver Operator Curve), with higher values indicating superior detection capabilities. Pixel-level anomalies are evaluated via the same curve for segmentation accuracy. Besides, the experimental results for the other metrics are listed in the appendix.

5.3. Implementation details

The backbone architecture used in our experiments is directly adopted from Point-MAE [28]. Instead of training directly on our dataset, we first train the backbone using ShapeNet-55 [12] with a point size of 8192. Regarding the geometry-aware sample module, we set the threshold τ to 0.3, and the number of points sampled in salient regions is twice that of non-salient regions. When fine-tuning

our model on both the Anomaly-ShapeNet and Real3D-AD datasets, we convert the point clouds into a 256×64 point-patch structure. Here, 256 represents the number of central points sampled, and 64 represents the number of neighborhood points selected using K-nearest neighbors (KNN). During training and testing, we set the mask rate to 0.4, and the number of iterations for testing is 3. For anomaly scoring, we utilize the $1st$, $2nd$, and $3rd$ intermediate layers of the backbone's decoder for comparison.

5.4. Anomaly detection on Anomaly-ShapeNet and Real3D-AD Datasets

Anomaly detection results on Anomaly-ShapeNet and Real3D-AD are shown in table 2 and table 3. Compared to other methods, our IMR-Net performs better on the average I-AUROC, which achieves 72.5% on Real3D-AD and 66.1% on Anomaly ShapeNet. In table 3, our IMR-Net achieves the highest score for 19 out of 40 classes. We visualize some representative samples of Anomaly-ShapeNet for anomaly detection and localization in Figure 4 .

5.5. Ablation study

Effectiveness of geometry aware sampling. FPS (farthest point sampling) and RS (random samping) are widely used in 3D point processing. However, when the sampling ratio is too low, FPS (Farthest Point Sampling) and RS (Random Sampling) may result in the sampled defects too slightly to be detect. Therefore, we employ the geometry aware sampling. To demonstrate the superiority of the GPS algorithm in anomaly detection, we conducted ablation experiments alongside RS , FPS , and Voxel Down-sampling methods. As shown in table 4 , when using our GPS, the higher I-AUC (0.66) and P-AUC (0.65) were achieved.

Analysis of masking ratio. Figure 5 shows the influence of masking ratio. The optimal ratio is 0.4, which is good both for reconstruction and feature representation. When the masking rate is reduced, the anomaly regions may not be covered during the iteration process. Conversely, when the masking ratio increases, the limited data may prevent the model from convergence. Figure 5b analyzes the relationship between the size of the anomalous region and the masking rate, where we find that point clouds with larger anomalous areas require higher masking rates.

Feature discrimination ability. Previous methods like M3DM [39] and RegAD [25] primarily employed pretrained models on other datasets to extract features, leading to domain bias. In contrast, our IMRNet, as a self-supervised network, effectively extracts features from the abnormal and normal point clouds in the Real3D-AD and Anomaly-ShapeNet datasets. This is directly proved in Figure 6. Our extracted features have a more compacted feature space and this property makes the features more suitable for memory bank building and feature distance calculation.

# Sample	Metrics	
	I-AUC	P-AUC
RS	0.55	0.61
FPS	0.64	0.62
Voxel	0.62	0.55
GPS	**0.66**	**0.65**

Table 4. **Ablation study on the sample methods.** The bold number represents the sample method corresponding to the highest index. RS represents random sampling and FPS represents farthest random sampling. Voxel denotes voxel down-sample and GPS is our geometry aware sampling.
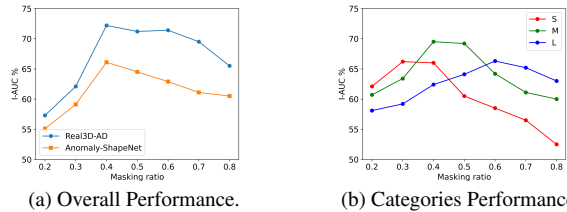

(a) Overall Performance. (b) Categories Performance.

Figure 5. Ablation study of masking ratio. (S,M,L) represent (small,middle,large) size anomaly. At the masking ratio 0.4, the overall performance is best. Objects with larger anomaly correspond to higher optimal masking ratios.
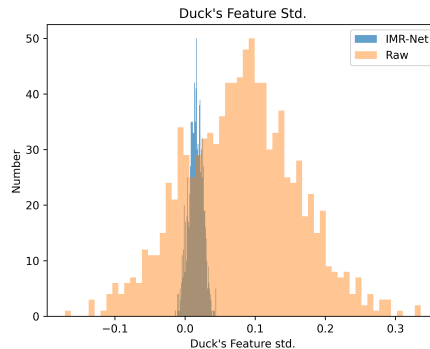

Figure 6. Histogram of standard deviation along each dimension of feature of extracted by pretrained model and our IMRNet model. We show a case of the *Duck* class in Real3D-AD dataset.

6. Conclusion

In this work, we propose Anomaly-ShapeNet, a synthetic 3D point dataset for anomaly detection, containing realistic and challenging samples. The diverse point clouds with high accuracy and reasonable quantity in Anomaly-ShapeNet make it more suitable for various 3D algorithms. Moreover, we firstly introduce IMRNet, a self-supervised model based on 3D point mask reconstruction, achieving the state-of-the-art performance on both the Anomaly-ShapeNet and Real3D-AD datasets.

References

[1] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4182–4191, 2020. 1, 2

[2] Paul Bergmann, Xin Jin, David Sattlegger, and Carsten Steger. The MVTec 3D-AD Dataset for Unsupervised 3D Anomaly Detection and Localization. In *17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, volume 5: VISAPP, Setúbal, 2022. Scitepress. 1, 3

[3] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving Unsupervised Defect Segmentation by Applying Structural Similarity to Autoencoders. In Alain Tremeau, Giovanni Farinella, and Jose Braz, editors, *14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, volume 5: VISAPP, pages 372–380, Setúbal, 2019. Scitepress. 2

[4] Paul Bergmann and David Sattlegger. Anomaly detection in 3d point clouds using deep geometric descriptors. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2613–2623, 2023. 2

[5] Luca Bonfiglioli, Marco Toschi, Davide Silvestri, Nicola Fioraio, and Daniele De Gregorio. The eyecandies dataset for unsupervised multimodal anomaly detection and localization. In *Proceedings of the Asian Conference on Computer Vision*, pages 3586–3602, 2022. 3

[6] Yunkang Cao, Yanan Song, Xiaohao Xu, Shuya Li, Yuhao Yu, Yifeng Zhang, and Weiming Shen. Semi-supervised knowledge distillation for tiny defect detection. In *2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, pages 1010–1015. IEEE, 2022. 2

[7] 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

[8] Yunkang Cao, Xiaohao Xu, and Weiming Shen. Complementary pseudo multimodal feature for point cloud anomaly detection. *arXiv preprint arXiv:2303.13194*, 2023. 2, 3, 7

[9] Yunkang Cao, Xiaohao Xu, Chen Sun, Yuqi Cheng, Zongwei Du, Liang Gao, and Weiming Shen. Segment any anomaly without training via hybrid prompt regularization. *arXiv preprint arXiv:2305.10724*, 2023. 2

[10] Yunkang Cao, Xiaohao Xu, Jiangning Zhang, Yuqi Cheng, Xiaonan Huang, Guansong Pang, and Weiming Shen. A survey on visual anomaly detection: Challenge, approach, and prospect. *arXiv preprint arXiv:2401.16402*, 2024. 2

[11] Fabio Carrara, Giuseppe Amato, Luca Brombin, Fabrizio Falchi, and Claudio Gennaro. Combining GANs and AutoEncoders for efficient anomaly detection. In *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021. 2

[12] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2, 3, 7

[13] Ruitao Chen, Guoyang Xie, Jiaqi Liu, Jinbao Wang, Ziqi Luo, Jinfan Wang, and Feng Zheng. Easynet: An easy network for 3d industrial anomaly detection. *arXiv preprint arXiv:2307.13925*, 2023. 3

[14] Xuhai Chen, Yue Han, and Jiangning Zhang. A Zero-/Few-Shot Anomaly Classification and Segmentation Method for CVPR 2023 VAND Workshop Challenge Tracks 1&2: 1st Place on Zero-shot AD and 4th Place on Few-shot AD. *arXiv preprint arXiv:2305.17382*, 2023. 2

[15] Niv Cohen and Yedid Hoshen. Sub-image anomaly detection with deep pyramid correspondences. arXiv preprint arXiv:2005.02357, 2020. 1, 2

[16] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: A patch distribution modeling framework for anomaly detection and localization. In Alberto Del Bimbo, Rita Cucchiara, Stan Sclaroff, Giovanni Maria Farinella, Tao Mei, Marco Bertini, Hugo Jair Escalante, and Roberto Vezzani, editors, *Pattern Recognition. ICPR International Workshops and Challenges*, pages 475–489. Springer International Publishing, 2021. 1, 2

[17] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. PaDiM: a patch distribution modeling framework for anomaly detection and localization. In *International Conference on Pattern Recognition*, pages 475–489. Springer, 2021. 1, 2

[18] Thibaud Ehret, Axel Davy, Jean-Michel Morel, and Mauricio Delbracio. Image Anomalies: A Review and Synthesis of Detection Methods. *Journal of Mathematical Imaging and Vision*, 61(5):710–743, 2019. 2

[19] Denis Gudovskiy, Shun Ishizaka, and Kazuki Kozuka. CFLOW-AD: Real-Time Unsupervised Anomaly Detection With Localization via Conditional Normalizing Flows. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 98–107, 2022. 1, 2

[20] Eungi Hong and Yoonsik Choe. Latent feature decentralization loss for one-class anomaly detection. *IEEE Access*, 8:165658–165669, 2020. 2

[21] Eliahu Horwitz and Yedid Hoshen. Back to the feature: classical 3d features are (almost) all you need for 3d anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2967–2976, 2023. 3, 7

[22] Jongheon Jeong, Yang Zou, Taewan Kim, Dongqing Zhang, Avinash Ravichandran, and Onkar Dabeer. Winclip: Zero-/few-shot anomaly classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19606–19616, 2023. 2

[23] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 2

[24] Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. CutPaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, pages 9664–9674, 2021. 1, 2

[25] Jiaqi Liu, Guoyang Xie, Ruitao Chen, Xinpeng Li, Jinbao Wang, Yong Liu, Chengjie Wang, and Feng Zheng. Real3d-ad: A dataset of point cloud anomaly detection. *arXiv preprint arXiv:2309.13226*, 2023. 1, 3, 7, 8

[26] Pankaj Mishra, Claudio Piciarelli, and Gian Luca Foresti. A neural network for image anomaly detection with deep pyramidal representations and dynamic routing. *International Journal of Neural Systems*, 30(10):2050060, 2020. 1, 2

[27] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep learning for anomaly detection: A review. *ACM Comput. Surv.*, 54(2), 2021. 2

[28] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, pages 604–621. Springer, 2022. 5, 7

[29] Kevin M. Potter, Brendan Donohoe, Benjamin Greene, Abigail Pribisova, and Emily Donahue. Automatic detection of defects in high reliability as-built parts using x-ray CT. In *Applications of Machine Learning 2020*, volume 11511, pages 120 – 136. International Society for Optics and Photonics, SPIE, 2020. 2

[30] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 5

[31] 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. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 2

[32] Tal Reiss, Niv Cohen, Liron Bergman, and Yedid Hoshen. Panda: Adapting pretrained features for anomaly detection and segmentation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2805–2813, 2021. 1, 2

[33] Oliver Rippel, Arnav Chavan, Chucai Lei, and Dorit Merhof. Transfer Learning Gaussian Anomaly Detection by Fine-Tuning Representations. arXiv preprint arXiv:2108.04116, 2021. 1, 2

[34] Nicolae-Catalin Ristea, Neelu Madan, Radu Tudor Ionescu, Kamal Nasrollahi, Fahad Shahbaz Khan, Thomas B Moeslund, and Mubarak Shah. Self-supervised predictive convolutional attentive block for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 1, 2

[35] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14318–14328, 2022. 1, 2, 5, 6, 7

[36] Marco Rudolph, Tom Wehrbein, Bodo Rosenhahn, and Bastian Wandt. Asymmetric student-teacher networks for industrial anomaly detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2592–2602, 2023. 2, 3

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

[38] Lu Wang, Dongkai Zhang, Jiahao Guo, and Yuexing Han. Image anomaly detection using normal data only by latent space resampling. *Applied Sciences*, 10(23), 2020. 2

[39] Yue Wang, Jinlong Peng, Jiangning Zhang, Ran Yi, Yabiao Wang, and Chengjie Wang. Multimodal industrial anomaly detection via hybrid fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8032–8041, 2023. 3, 7, 8

[40] 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. 2

[41] Guoyang Xie, Jingbao Wang, Jiaqi Liu, Feng Zheng, and Yaochu Jin. Pushing the limits of fewshot anomaly detection in industry vision: Graphcore. *arXiv preprint arXiv:2301.12082*, 2023. 2

[42] Xiaohao Xu, Tianyi Zhang, Sibo Wang, Xiang Li, Yongqi Chen, Ye Li, Bhiksha Raj, Matthew Johnson-Roberson, and Xiaonan Huang. Customizable perturbation synthesis for robust slam benchmarking. *arXiv preprint arXiv:2402.08125*, 2024. 2

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

[44] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16259–16268, 2021. 5

[45] Qiang Zhou, Weize Li, Lihan Jiang, Guoliang Wang, Guyue Zhou, Shanghang Zhang, and Hao Zhao. Pad: A dataset and benchmark for pose-agnostic anomaly detection. *arXiv preprint arXiv:2310.07716*, 2023. 1, 3

[46] Yang Zou, Jongheon Jeong, Latha Pemula, Dongqing Zhang, and Onkar Dabeer. SPot-the-Difference self-supervised pretraining for anomaly detection and segmentation. In *Proceedings of the European Conference on Computer Vision*, 2022. 1, 2