SiM3D: Single-instance Multiview Multimodal and Multisetup 3D Anomaly Detection Benchmark

Alex Costanzino[1] Pierluigi Zama Ramirez[1] Luigi Lella[1] Matteo Ragaglia[2] Alessandro Oliva[2]
Giuseppe Lisanti[1] Luigi Di Stefano[1]
[1]CVLab, University of Bologna, Italy [2]SACMI Imola, Italy
https://alex-costanzino.github.io/SiM3D/

Abstract

We propose SiM3D, the first benchmark considering the integration of multiview and multimodal information for comprehensive 3D anomaly detection and segmentation (ADS), where the task is to produce a voxel-based Anomaly Volume. Moreover, SiM3D focuses on a scenario of high interest in manufacturing: single-instance anomaly detection, where only one object, either real or synthetic, is available for training. In this respect, SiM3D stands out as the first ADS benchmark that addresses the challenge of generalising from synthetic training data to real test data. SiM3D includes a novel multimodal multiview dataset acquired using top-tier industrial sensors and robots. The dataset features multiview high-resolution images (12 Mpx) and point clouds (∼7M points) for 333 instances of eight types of objects, alongside a CAD model for each type. We also provide manually annotated 3D segmentation GTs for anomalous test samples. To establish reference baselines for the proposed multiview 3D ADS task, we adapt prominent singleview methods and assess their performance using novel metrics that operate on Anomaly Volumes.

1. Introduction

In computer vision, anomaly detection and segmentation (ADS) involves identifying anomalous samples and localising their defects. This task is challenging in industrial contexts due to the variability and rarity of defects. Typically, ADS in the industry follows a *cold-start* approach, where training is unsupervised and uses only defect-free (nominal) samples. Before the release of the MVTec AD dataset in 2019 [4], only a limited number of ADS methods were proposed, with just 9 publications in major computer vision conferences (e.g., CVPR, ECCV, ICCV, WACV) over the five years before. The introduction of MVTec AD marked a pivotal moment in the field, fostering the publication of roughly 172 papers in these venues over the next 5 years. This dataset not only stimulated the development of several ADS methods [14, 30, 36] but also promoted the creation of novel and more challenging benchmarks. For instance,

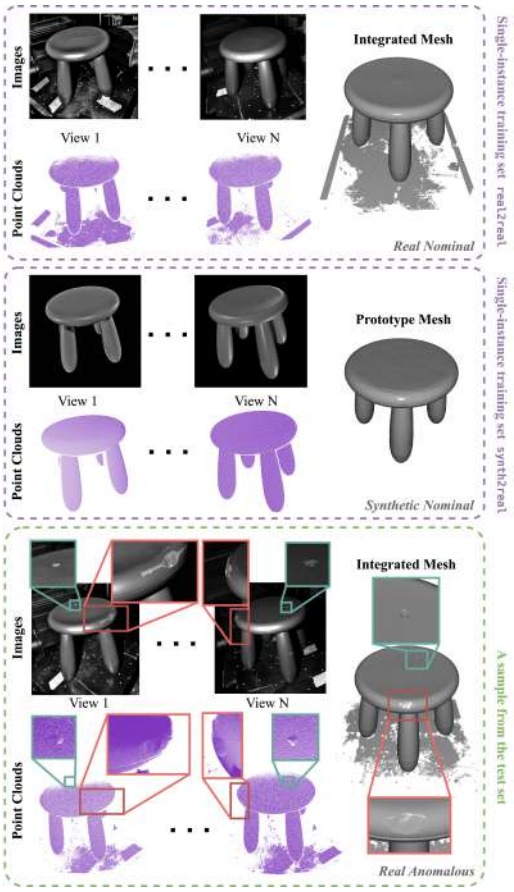

Figure 1. **SiM3D dataset overview.** From top to bottom: the single-instance real and synthetic training samples for object *Plastic Stool*, one of the anomalous samples from the test set.

MVTec 3D-AD and [6] and Eyecandies [7] introduced the first benchmarks deploying multimodal inputs to enhance ADS performance. Then, PAD [45] and Real-IAD [35] introduced novel benchmarks that leverage multiple RGB views of objects. Although these benchmarks tackle various settings, two real industrial issues remain unaddressed.

First, existing benchmarks have primarily addressed the problem of 2D ADS. Consequently, current methods gen-

Dataset	Venue & Year	2D	No. Pixels	3D	No. Points	Scenario	Train	Test	Setup	Task	No. Classes
MVTec AD	CVPR 2019	RGB Images	490k – 1M	—	—	Multi-instance	Singleview	Singleview	`real2real`	2D	15
MVTec 3D-AD	VISAPP 2021	RGB Images	160k – 810k	XYZ Images	10k – 195k	Multi-instance	Singleview	Singleview	`real2real`	2D	10
Eyecandies	ACCV 2022	RGB Images	262k	Depths + Normals	262k	Multi-instance	Singleview	Singleview	`synth2synth`	2D	10
MVTec LOCO AD	IJCV 2022	RGB Images	1.28M – 2.05M	—	—	Multi-instance	Singleview	Singleview	`real2real`	2D	5
VisA	ECCV 2022	RGB Images	1.5M	—	—	Multi-instance	Singleview	Singleview	`real2real`	2D	12
PAD	NeurIPS 2023	RGB Images	12M	—	—	Single-instance	Multiview	Singleview	`real2real` + `synth2synth`	2D	20
Real3D-AD	NeurIPS 2023	—	—	Point Clouds	43k – 2.68M	Multi-instance	Multiview	Singleview	`real2real`	3D	12
Real-IAD	CVPR 2024	RGB Images	10M	—	—	Multi-instance	Singleview	Singleview	`real2real`	2D	30
SiM3D	**2025**	**Grayscale Images**	**12M**	**Point Clouds + Mesh**	**5M – 7M**	**Single-instance**	**Multiview**	**Multiview**	**`real2real` + `synth2real`**	**3D**	8

Table 1. **Popular ADS benchmarks.** Chronologically sorted, the sequence hints at a growing interest in multimodal and multiview data.

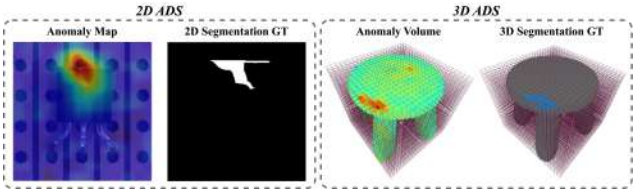

Figure 2. **2D vs. 3D anomaly detection and segmentation.** Typical ADS benchmarks rely on 2D Anomaly Maps and 2D ground-truths (left) to assess segmentation performance. SiM3D proposes to do so by 3D Anomaly Volumes and 3D ground-truths (right).

erate Anomaly Maps, i.e., images that encode the likelihood of each pixel belonging to a defect, as shown in Fig. 2 (left). However, many industrial applications require precise localisation of defects in the 3D space to allow automatic intervention, reducing waste and optimising production. Benchmarks relying on multimodal input data, like those proposed in MVTec 3D-AD [6] and Eyecandies [7], represent progress toward this goal by providing 3D information for each pixel within an image. However, in these benchmarks, ADS methods are still expected to provide as output a 2D Anomaly Map, and objects are inspected from a single viewpoint. For comprehensive 3D anomaly detection, we must observe an object in its entirety, utilising captures from multiple viewpoints to detect defects across the entire 3D space. This calls for methods capable of integrating multiview information to estimate the defect probability at each 3D location within an Anomaly Volume, as shown in Fig. 2 (right). Furthermore, as demonstrated by MVTec 3D-AD and Eyecandies, these methods should process multimodal information, i.e., RGB images as well as 3D information, to achieve more reliable anomaly detection.

Secondly, current benchmarks typically provide multiple training samples per object class to capture the variability across nominal objects. However, collecting many samples – even nominal ones – can be costly and challenging. For example, line changeovers would demand acquiring new datasets and retraining models, both highly time-intensive processes. Additionally, collecting large datasets increases the risk of errors, such as misjudged nominal samples, which may degrade model performance. Moreover, manufactured objects typically closely resemble one another as they replicate the structure and materials of a synthetic CAD prototype. Thus, the variability across nominal samples of manufactured objects is frequently minimal, with a single nominal item subsuming the information required to spot anomalies in other objects.

These considerations highlight the need for ADS techniques that may be effectively trained by a single nominal sample or even a synthetic prototype (e.g., an object's CAD model) and then be able to generalise to other manufactured objects to identify anomalous ones effectively. The ability to perform training by a single object instance, either real or synthetic, would avoid the need for extensive data collection, vastly facilitating the creation and adaptation of ADS models. Yet, most current benchmarks focus on multiple training instances setups, and none of the prior works consider challenging synthetic (training) to real (test) setups.

To fill all the gaps highlighted by our considerations and stimulate interest in the development of multimodal and multiview 3D ADS methods, we propose the Single-instance Multiview Multimodal Multisetup dataset, dubbed SiM3D. The dataset contains a series of high-resolution scans of manufactured objects, which can be nominal or anomalous. Each scan consists of multiple views, each featuring a greyscale image, its corresponding point cloud (i.e., multimodal data), and the ground-truth acquisition pose. In addition, an integrated 3D mesh is provided with each scan. For each object, a single nominal sample has been retained to be employed as a *single-instance* training set, while the remaining samples, both nominal and anomalous, compose the test set. Moreover, to foster research on novel approaches that may address the domain gap between synthetic training data and real test samples in ADS, for each object of our dataset, we also provide a *synthetic scan*, i.e., we deploy the available CAD model to obtain the same kind of data as in our high-resolution real scans, with images and point clouds rendered from the same poses. An overview of the proposed dataset is depicted in Fig. 1.

Based on our novel dataset, we create the first benchmark for 3D anomaly detection, where the task is to predict anomaly scores within a voxelized 3D Anomaly Volume rather than a 2D Anomaly Map (Fig. 2). Peculiarly, our benchmark includes two setups based on the domain of the data available for training: in the `real2real` setup, a model is given access to a single nominal instance of an object (Fig. 1, top), while in `synth2real` the same kind

of training data are obtained from the CAD model of the object (Fig. 1, centre). We adapt prominent ADS methods – both unimodal and multimodal – for the proposed task and evaluate them on our benchmark. To assess segmentation performance, we provide accurate 3D ground-truths (Fig. 2) and extend ADS metrics to operate on anomaly scores and ground-truths specified on 3D voxel grids. Our experimental findings highlight the open challenges in the field and offer insights into possible future directions.

2. Related Work

The MVTec AD [4] dataset significantly boosted anomaly detection research, inspiring new techniques and benchmarks for various challenges. MVTec AD provides a training set containing only nominal images and a test set that includes nominal and anomalous ones for several object categories. This unsupervised setup exemplifies many real-world applications in which anomalous objects are challenging to collect, and thus, it has also been adopted by more recent ADS benchmarks. Among them, MVTec LOCO [5] presents a challenging scenario with both structural anomalies (e.g., dents, holes) and logical anomalies (e.g., incorrect object combination). VisA [46] introduced high-resolution images of complex scenes containing multiple instances of the same object. However, certain defects may be visible only by analysing objects' 3D geometry. For this reason, multimodal datasets, such as MVTec 3D-AD [6] and Eyecandies [7] include 3D data alongside images. For each sample in the dataset, MVTec 3D-AD provides images and pixel-aligned XYZ information captured by a 3D sensor, while Eyecandies contains synthetic images with pixel-aligned depths and normals. Recently, PAD [45] addressed pose-agnostic ADS with the first multiview dataset (MAD), featuring images of objects from various viewpoints. In PAD, training samples are multiview RGB images with known poses, while test samples are singleview RGB images with unknown poses. MAD includes both synthetic and real objects, but its benchmarks do not explore synthetic-to-real generalisation. Real-IAD [35] introduced a large-scale multiview dataset with RGB images covering various defect sizes. Eventually, Real3D-AD [21] is the first point cloud anomaly detection benchmark where the task is to predict an anomaly score for each point. It uses front-and-back scans of nominal objects for training and a singleview point cloud for testing.

As shown in Tab. 1, the proposed SiM3D benchmark offers several outstanding features, including very-high-resolution images (12 Mpx), large point clouds/meshes containing 5-7 million elements captured with high-precision 3D sensors. Moreover, it is designed for a highly relevant manufacturing scenario: single-instance anomaly detection, where only one object is available for model training. To promote research toward even more cost-effective scenarios, ours is also the only benchmark that addresses the challenge of generalising from synthetic training data to real test data. Finally, SiM3D stands out as the first benchmark that considers integrating multiview information at test time for comprehensive 3D anomaly detection, where the task consists of outputting a voxel-based Anomaly Volume.

ADS Methods. Following the setup defined in the MVTec-AD benchmark, most ADS methods [22] are designed to estimate 2D anomaly maps from a single image [2, 3, 9, 10, 13, 15, 16, 19, 20, 23, 26, 27, 27–29, 31, 34, 37–41, 43, 44]. One of the most popular methods is PatchCore [30], which uses feature extractors pre-trained on large datasets [8, 17, 24] to gather features from nominal samples into a memory bank. At test time, input sample features are compared to this bank to detect anomalies. Another more recent yet popular approach, EfficientAD [1], uses a lightweight feature extractor and a student-teacher model to identify anomalies based on feature prediction discrepancy. The introduction of the MVTec 3D-AD benchmark has led to the development of several multimodal approaches that use images and 3D data to generate 2D anomaly maps [14, 18, 32, 36]. Among them, BTF [18] is a memory bank approach inspired by PatchCore, in which each element is the concatenation of 2D and 3D features extracted by a pre-trained image backbone and a traditional point cloud descriptor, i.e., FPFH [33], respectively. M3DM [36] improves BTF using powerful transformer-based pre-trained backbones, such as DINO-v1[8] and Point-MAE [25], to extract features from RGB images and point clouds. It also learns to fuse 2D and 3D features into multimodal features. During inference, a trained One-Class SVM learns to aggregate modality-specific scores. AST [32] follows a Teacher-Student paradigm, in which the Teacher leverages Normalizing Flows to model the multimodal features distribution, while the Student employs a feed-forward network to model such distribution. This asymmetry in the architectures exacerbates the discrepancies between nominal and anomalous features at inference time. Recently, starting from the same transformer-based backbones as in [36], CFM [14] learns to map features from one modality to the other by nominal samples and then, at inference time, detects anomalies by pinpointing inconsistencies between observed and predicted features. In this paper, we employ the above-mentioned methods to investigate the challenges of SiM3D by adapting them to work in a multiview setup.

3. Dataset Creation

Material preparation. We scanned eight types of manufactured objects spanning different sizes, shapes, textures and materials, as shown in Fig. 3 (1.i). For each object type, between 20 and 100 instances were purchased from a store. We selected objects suitable for single-instance scenarios – instances are manufactured according to a unique

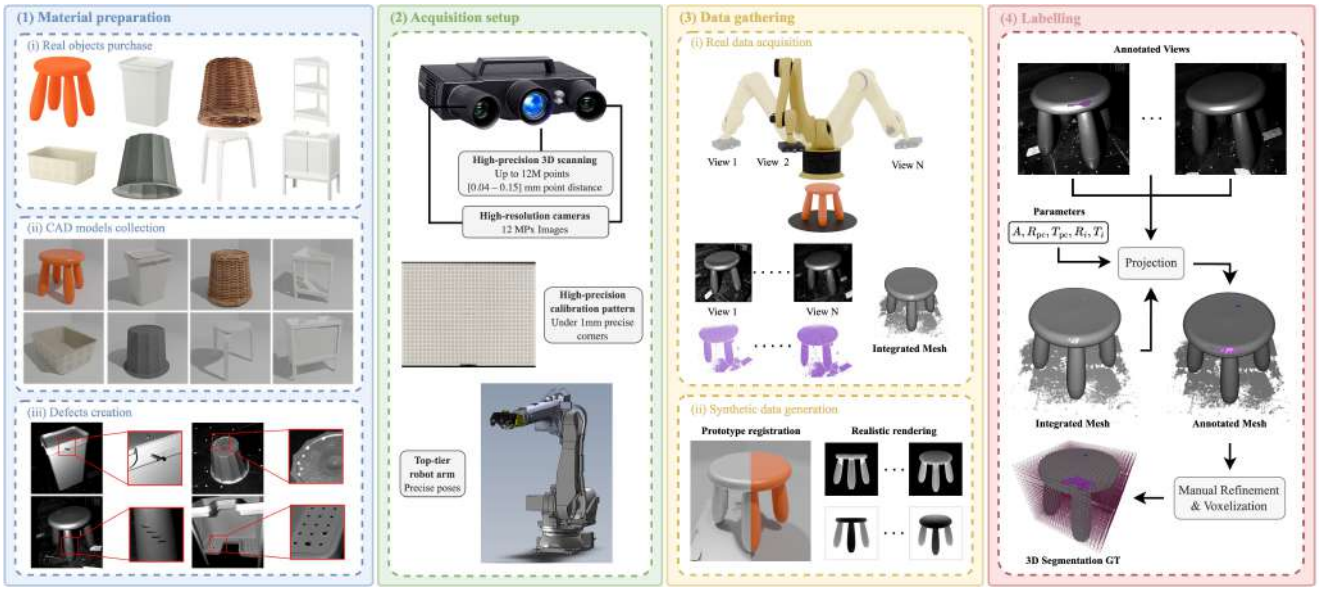

Figure 3. **SiM3D dataset creation pipeline.** (1) We purchased eight types of objects with different properties, each associated with a CAD model. We altered part of the samples to create realistic anomalies; (2) We designed and calibrated an acquisition setup consisting of a high-precision 3D sensor with high-resolution cameras on top an industrial robot arm; (3) We acquired 360-degree scans of each object instance to populate the real split of the training and test sets, and rendered images and point clouds from the CAD models to create the synthetic split of the training sets; (4) We created voxel-based segmentation GTs by a mixed 2D-3D labelling pipeline.

prototype model so that all nominal ones should be identical – and for which an official CAD prototype was available from the manufacturer. The prototype CAD models are depicted in Fig. 3 (1.ii). To simulate various anomalies, we manually introduced different types of defects that represent realistic manufacturing flaws, as illustrated in Fig. 3 (1.iii)). We altered the objects' appearance (e.g., paint modifications, Fig. 3 (1.iii) – bottom left), geometry (e.g., dents, Fig. 3 (1.iii) – bottom right), or both (e.g., contaminations, Fig. 3 (1.iii) – top left). It is important to point out that only half of the instances within each object type are modified, while the remaining half are kept in their original state to serve as nominal data. For each object type, one nominal instance will be used as the training set in the `real2real` experimental setup, while all other ones, both nominal and modified (i.e., anomalous), will make up the test set. This very same test set is also used in the `synth2real` setup, though the training data comes from CAD prototypes.

Sensor. We acquired our dataset with a top-tier 3D scanner, the Atos Q from ZEISS (Fig. 3 (2.i)). It consists of a stereo pair with two 12 `Mpx` greyscale cameras and a light projector. We consider the left camera to be our reference. The images have a resolution of 4096×3000 pixels. The point clouds can feature up to 12 million elements, with a point-to-point distance between 0.04 and 0.15 mm. The sensor is mounted on an industrial robot arm with high repeatability and precision (Fig. 3 (2.iii)).

Calibration. We calibrate the greyscale cameras with a standard chequerboard pattern, estimating the intrinsic parameters matrix, A, and the radial and tangential distortion coefficients, $\delta = [k_1, k_2, k_3, p_1, p_2]$. We also calibrate the rotation, R_{pc}, and translation, T_{pc}, from the point cloud to the left camera reference frame, with a precise dotted calibration pattern built by ZEISS (Fig. 3 (2.ii)) that provides centre coordinates with precision < 1 mm. More details are in the supplementary.

Real data acquisition. The acquisition process consists of 360-degree scanning operations around all object instances of our dataset (see Fig. 3 (3.i)). We define specific viewpoints to capture data from multiple angles around the object. These trajectories are sampled from concentric hemispheres that depend on the object's size. We provide poses relative to a reference view, i.e., v_1. Thus, the pose for the view i, v_i, consists of a rotation R_i and a translation T_i that aligns v_i to v_1, with R_1 the identity matrix and T_1 null. Since in real industrial setups object instances would be acquired from as similar as possible viewpoints, we place markers in the scene to facilitate consistent positioning of objects of the same type. Thus, thanks to the high repeatability of the robot arm, images, I_i, and point clouds, P_i, from the same viewpoint v_i turn out similar across acquisitions of different instances of the same object type. However, since the items were positioned manually with the help of the markers, slight displacements cannot be avoided. This introduces some variability across the scans of the instances of an object, as it also typically occurs in real indus-

trial pipelines. For each object, after collecting data from all viewpoints, the ATOS Q sensor also returns an integrated mesh, M, obtained by registering all point clouds from the different views using a proprietary algorithm. We collect and release the mesh to foster the development of methodologies that may also exploit this 3D representation. The 3D reference frame is aligned to the one of the first point cloud P_1 taken from the first viewpoint v_1.

Synthetic data generation. We employ the Blender [11] Python API to render synthetic data that simulates real acquisitions. To ensure the synthetic and real data are aligned, we select a reference mesh from the collected real instances for each object type, load it in Blender, and align it to the prototype CAD model, as shown in (Fig. 3 (3.ii) - left). Then, we set the Blender camera parameters to the intrinsics, A, of the left camera of our Atos Q sensor and render images from the same viewpoints v_i employed during the real acquisitions. The RGB renders are converted into greyscale images to match the real images provided by our acquisition setup (Fig. 3 (3.ii) - right). Renderings have been obtained by exploiting Cycles Renderer, a path-tracing renderer able to provide realistic results with reflections, emissive surfaces, and physically-based lighting. We also render depth maps (Fig. 3 (3.ii) - right), which we project into 3D point clouds (see Fig. 1, centre) using the intrinsic parameters, A.

Labelling. Since some anomalies, such as paintings and minor scratches, are visible only in greyscale images, while others, such as dents, are better visible in 3D, we must consider both sensing modalities to obtain precise 3D segmentation ground-truths for anomalous samples. Therefore, we develop a two-step labelling strategy, illustrated in Fig. 3 (4), that operates on both 2D images and 3D data. First, we manually annotate all the view images, I_i, where anomalies are clearly visible, obtaining dense 2D segmentation masks. We employ a different ID and colour for every defect to ease the voxel AUPRO computation (see Sec. 4). Subsequently, for each annotated image, I_i, we leverage the associated integrated mesh, M, the intrinsic parameters A, the point cloud to camera reference frame transformation, $(R_{\text{pc}}, T_{\text{pc}})$, and the transformation between v_i and v_1, (R_i, T_i), to project the 2D annotations onto the integrated mesh. Then, since the projection may not be precise due to occlusions and slanted surfaces, each ground-truth mesh is thoroughly inspected and manually refined using a 3D visualisation and editing software, i.e., CloudCompare [12]. The labelling process was conducted by a team of 4 experts, with a final cross-check of the results. Finally, the annotated mesh is converted into a voxel grid with a voxel size of 2 mm. The choice of the size is determined by the smallest defect size discernible in the meshes. In the conversion process, a voxel is labelled as anomalous if it intersects at least one triangle labelled as such in the annotated mesh.

Type	Dimensions [cm]	Dimensions [voxels]	Total Instances	Train Instances Nominal	Test Instances Nominal	Test Instances Anomalous	Views per Instance
Plastic Stool	$35 \times 35 \times 30$	$330 \times 326 \times 317$	22	1 `real` or 1 `synth`	10	10	12
Rubbish Bin	$26 \times 21 \times 33$	$333 \times 339 \times 326$	42	1 `real` or 1 `synth`	20	20	12
Wicker Vase	$17 \times 17 \times 15$	$329 \times 309 \times 290$	22	1 `real` or 1 `synth`	10	10	12
Bathroom Furniture	$33 \times 33 \times 50$	$371 \times 392 \times 363$	20	1 `real` or 1 `synth`	8	10	36
Container	$20 \times 25 \times 10$	$327 \times 308 \times 286$	94	1 `real` or 1 `synth`	46	46	12
Plastic Vase	$12 \times 12 \times 9$	$326 \times 308 \times 286$	99	1 `real` or 1 `synth`	48	49	12
Wooden Stool	$48 \times 42 \times 45$	$331 \times 367 \times 327$	15	1 `real` or 1 `synth`	6	7	12
Sink Cabinet	$44 \times 25 \times 50$	$342 \times 349 \times 357$	19	1 `real` or 1 `synth`	9	8	36

Table 2. **Dataset statistics.**

4. SiM3D Benchmark

4.1. Dataset, Tasks, and Metrics

Dataset and Setups. Using the procedure described in Sec. 3, we collected 333 object instances belonging to the eight object types, as shown in Tab. 2. We captured from 12 to 36 views for each instance, depending on the type. We split the data into training and test sets. For each type, the training set comprises a single instance, either real in the `real2real` setup or synthetic in the `synth2real` setup. The test set includes all other instances and is balanced between nominal and anomalous samples. The test set is the same for both setups.

Tasks. Our benchmark focuses on the tasks of detecting and segmenting objects' anomalies using multiview and multimodal data. Specifically, the inputs to the methods consist of a set of images $\{I_i\}_{i=1}^n$ captured from different viewpoints alongside 3D data, $\mathbf{X}$, that capture an object entirely. The 3D data that a method can use are a set point clouds $\{P_i\}_{i=1}^n$ or depth maps $\{D_i\}_{i=1}^n$ taken from the same vantage points as the images, an integrated mesh M, or also a merged point cloud P. A method capable of multiview multimodal 3D anomaly detection should take these inputs, integrate the 2D and 3D information from multiple views, and produce a global anomaly score for the test sample under inspection. On the other hand, a method pursuing multiview multimodal 3D anomaly segmentation should yield a 3D output, where each 3D position carries a score representing the likelihood of belonging to an anomaly. In our benchmark, a method for this task takes as input all images and 3D data, $(\{I_i\}_{i=1}^n, \mathbf{X})$ to output an Anomaly Volume, i.e. a voxel grid, $V \in \mathbb{R}^{X \times Y \times Z}$, where X, Y, and Z are the grid dimensions and each voxel carries an anomaly score. For several reasons, we chose voxel grids as the 3D output representation of our benchmark. Firstly, as their grid structure resembles 2D Anomaly Maps, existing 2D segmentation methods and metrics can be easily extended. Moreover, voxel grids are suitable for representing every type of 3D defect, including missing parts. One major challenge with voxel grids pertains to choosing the voxel size. Ideally, we would want a size that allows accurate segmentation of even the smallest defects. However, the memory occupancy of voxel grids scales cubically with the inverse of the voxel size. Thus, we create ground-truth volumes using a voxel resolution of 2 mm, which enables precise localisation of

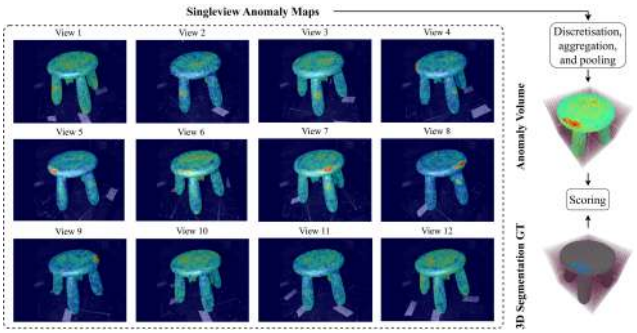

Figure 4. **Extending singleview methods.** Anomaly Maps obtained by processing singleview inputs with standard ADS methods are projected into the 3D and aggregated into a voxel grid, with each voxel containing a list of scores from multiple views. The Anomaly Volume is obtained by voxel-wise max pooling.

even the smallest defects present in our dataset while keeping memory occupancy affordable.

Metrics. Akin to standard practice in 2D ADS [4], we evaluate detection performance by the Area Under the Receiver Operator Curve (AUROC) of the global anomaly scores predicted by the methods. As we compute a global score for each object *instance*, we name it instance-level AUROC (I-AUROC). Regarding segmentation performance, we evaluate methods with an adaptation of the popular Area Under the Per-Region Overlap (AUPRO) curve. First of all, we define the voxel-based Per-Region Overlap (PRO) metric as $\text{PRO}_t = \frac{1}{N}\sum_{n=1}^{N}\frac{|\overline{\Psi}_t \cap V_n|}{|V_n|}$, where $\overline{\Psi}_t$ is an Anomaly Volume binarised using a threshold, t, N is the number of blobs, i.e., defects, in the ground-truth, and V_n are the voxels belonging to that blob. Since during labelling, we annotated each defect in our dataset with a different ID (see Sec. 3), the N defect blobs in a ground-truth are easily identifiable. This metric is computed across multiple thresholds, t, estimated at different False Positive Rates (FPRs). We calculate the FPR using only non-empty voxels. We sample FPRs from 0 to the maximum value uniformly, we estimate the corresponding threshold t, and we binarise the score grid accordingly. In this way, we construct a curve that plots the PRO values against the corresponding FPRs. The segmentation performance is obtained by integrating such a curve up to a specific False Positive Rate. As proposed in [14], we use 1% as integration bound for the PRO curve to set a challenging bar representative of the requirements of real industrial applications. Finally, we normalise the resulting area to the interval $[0, 1]$. We dub this metric voxel-level AUPRO (V-AUPRO).

4.2. Baselines

Extending singleview methods to Multiview 3D ADS. Current ADS methods predict a 2D anomaly map from a singleview input, either unimodal, e.g., images, or multimodal, e.g., images and point clouds. To obtain a set of baselines for our novel multiview 3D ADS task, we extend popular and effective singleview methods by means of the 3D aggregation strategy illustrated in Fig. 4. Given the considered method, we process the data corresponding to an input view, v_i, so as to obtain a 2D Anomaly Map. Similarly to Sec. 3, given the mesh, M, intrinsic parameters A, point cloud to camera reference frame transformation, $(R_{\text{pc}}, T_{\text{pc}})$ and view pose v_0, (R_i, T_i), we can project each pixel of the 2D Anomaly Map to its corresponding 3D position. We discretise the 3D position, finding the corresponding voxel index. We associate that voxel with the corresponding anomaly score. We repeat the process for each view, keeping track of the scores that project onto the same voxels. Eventually, we compute the maximum among all scores projected into each voxel, obtaining the final Anomaly Volume. To extract the global score required by the I-AUROC, we compute the maximum across all the scores within the Anomaly Volume.

Selected Methods. We extend some of the most popular and effective unimodal and multimodal singleview ADS methods, starting from their official code, and evaluate them on SiM3D. Regarding unimodal methods, we select PatchCore [30], with either WideResNet101 [42] or DINO-v2 [24] as backbone, to extract features from images of all views, $\{I_i\}_{i=1}^{N}$ of the single-instance training set to create memory banks. Moreover, we use EfficientAD [1] using all views as their training set. As for multimodal ADS methods, we use BTF [18] employing WideResNet101 [42] as the image encoder and FPFH [33] as the algorithm to extract point cloud features. We create memory banks using features extracted from images, $\{I_i\}_{i=1}^{N}$, and point clouds, $\{P_i\}_{i=1}^{N}$, of all views of the single-instance training set. We use the same strategy with M3DM [36]. As 2D feature extractor, we exploit DINO-v2. As point cloud feature extractor, instead of using a transformer backbone, i.e., Point-MAE [17], we utilise FPFH to extract 3D features, similarly to BTF. Indeed, we would need a severe downsample (from $\sim$5 – 7M to $\sim$8k points) to work with Point-MAE, and we verified that at such low resolution, most of the defects no longer show up in the point clouds. Finally, we select CFM [14], using all views and clouds as a single training set. Based on the same consideration as for M3DM, we use DINO-v2 and FPFH to produce 2D and 3D features. Due to computational constraints, all methods process images padded and downsampled to 1540×1540. Moreover, for compatibility with the employed 2D feature extractors, which accept RGB images, we create 3-channel inputs by stacking our greyscale images thrice.

3D Data pre-processing. To obtain less noisy data and improve the results of the above-mentioned baselines, we remove the background from 3D data. We estimate a 3D plane using a RANSAC-based plane segmentation algo-

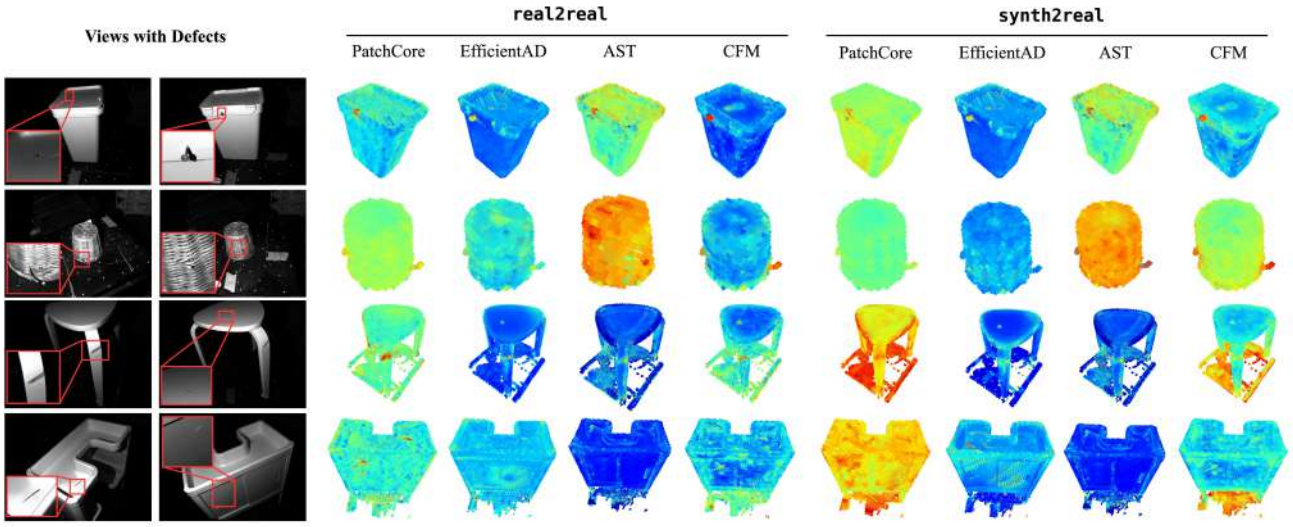

Figure 5. **Qualitatives.** Selected views with defects (left) and Anomaly Volumes for several methods (right) downsampled for visualisation.

Method	Modality	real2real									synth2real								
		Pl. Stool	Rub. Bin	W. Vase	B. Furn.	Cont.	Pl. Vase	W. Stool	Sink Cab.	**Mean**	Pl. Stool	Rub. Bin	W. Vase	B. Furn.	Cont.	Pl. Vase	W. Stool	Sink Cab.	**Mean**
PatchCore w/ WRN-101	RGB	0.740	0.987	0.636	0.777	**0.774**	0.551	**1.000**	0.566	**0.754**	0.462	0.235	0.537	0.353	**0.678**	0.576	0.517	0.250	0.451
PatchCore w/ DINO-v2	RGB	0.500	0.958	0.636	0.622	0.578	0.563	**1.000**	0.563	0.678	0.958	**0.897**	0.413	0.464	0.207	**0.684**	0.607	0.087	**0.540**
EfficientAD	RGB	0.280	0.732	0.000	**0.878**	0.424	**0.730**	0.928	0.712	0.585	0.123	0.673	0.000	**0.848**	0.008	0.487	**0.750**	0.662	0.443
AST	RGB	0.537	0.566	0.611	0.515	0.573	0.496	0.839	0.537	0.584	**1.000**	0.344	**1.000**	0.525	0.482	0.388	0.428	0.187	0.544
BTF	RGB + Point Cloud	0.421	0.217	0.504	0.565	0.545	0.471	0.678	0.424	0.478	0.462	0.294	0.520	0.444	0.464	0.424	0.482	0.287	0.422
CFM w/ DINO-v2 + FPFH	RGB + Point Cloud	0.198	0.301	0.074	0.515	0.483	0.456	0.732	0.825	0.448	0.008	0.000	0.190	0.424	0.313	0.459	0.428	0.362	0.273
M3DM w/ DINO-v2 + FPFH	RGB + Point Cloud	0.702	**0.988**	0.661	0.545	0.556	0.649	0.392	0.475	0.621	0.107	0.117	0.735	0.565	0.381	0.586	0.214	0.512	0.402
AST	RGB + Depth	**0.950**	0.927	**0.785**	0.474	0.542	0.470	0.428	**0.925**	0.687	0.636	0.002	0.504	0.474	0.462	0.569	0.428	**0.887**	0.495

Table 3. **Anomaly detection results (I-AUROC).** Best results in **bold**, runner-ups underlined.

rithm, which, iteratively, randomly selects a small subset of points (e.g., 10) to fit a candidate plane model. For the candidate planes, we compute the consensus set with threshold τ on the point-to-plane distance. After 1000 iterations, we return the best-fitting plane. To account for spurious background artefacts, we shift the plane by an offset α and finally remove all points below the shifted plane. Given the different sizes of the objects, for each object type, we select different optimal thresholds τ and offsets α, as detailed in the supplementary material. We apply this background removal technique to the integrated mesh, M, due to the larger number of background points, which facilitates the RANSAC algorithm. Then, from the background-filtered integrated mesh, we leverage ray-tracing techniques and known camera parameters to render, for each view, depth maps, $\{D_i\}_{i=1}^{N}$, pixel-aligned to the corresponding images, $\{I_i\}_{i=1}^{N}$ at the sensor resolution (4096×3000). Finally, these depth maps are padded and downsampled to 1540×1540, and the depth is lifted in 3D via the camera parameters to obtain dense and clean point clouds. The depth maps and point clouds obtained by rendering the pre-processed meshes serve as the input data for all the experiments presented in the subsequent section and will be available in the SiM3D dataset.

5. Benchmark Results

Main Results. In Tab. 3 and Tab. 4, we report the results obtained on the SiM3D benchmark by extending the considered methods as Multiview 3D ADS baselines. By looking at the mean detection performance (Tab. 3), we notice how methods that natively process only RGB images outperform natively multimodal ones in both the `real2real` and `synth2real` setups. With the notable exception of AST, this also stands for mean segmentation performance (Tab. 4). We posit that this is due to the main multimodal methods having been designed and tuned to process point clouds of the resolution featured by existing multimodal benchmarks, which is much lower than that of SiM3D, i.e. $\sim$ 8K vs $\sim$ 2.3M points (after downsampling). Indeed, AST is the only one designed to work natively with depth maps. Hence, our experiments suggest that current multimodal ADS methods do not scale to significantly higher resolutions. In turn, we could not even deploy the native 3D feature extractors of M3DM and CFM, the methods currently excelling on [6, 7]. Furthermore, we notice how methods relying on memory banks, such as PatchCore and M3DM, tend to attain slightly better mean detection and segmentation performance than their counterparts

Method	Modality	real2real									synth2real								
		Pl. Stool	Rub. Bin	W. Vase	B. Furn.	Cont.	Pl. Vase	W. Stool	Sink Cab.	**Mean**	Pl. Stool	Rub. Bin	W. Vase	B. Furn.	Cont.	Pl. Vase	W. Stool	Sink Cab.	**Mean**
PatchCore w/ WRN-101	RGB	0.710	0.461	0.761	0.675	0.694	0.747	0.380	0.609	0.630	**0.734**	0.437	0.751	0.454	0.678	0.741	0.386	0.618	0.600
PatchCore w/ DINO-v2	RGB	0.745	0.469	0.775	0.792	0.709	0.753	0.435	0.690	0.671	0.701	0.457	0.772	0.563	0.648	0.734	0.321	0.575	0.596
EfficientAD	RGB	0.682	0.462	0.763	0.534	0.680	0.743	0.407	0.488	0.594	0.680	0.449	0.729	0.523	0.669	0.747	0.431	0.361	0.573
AST	RGB	0.720	0.502	0.789	0.803	**0.716**	**0.764**	0.446	0.758	0.687	0.726	0.488	0.791	**0.806**	0.705	**0.764**	**0.461**	0.695	0.679
BTF	RGB + Point Cloud	0.551	0.402	0.750	0.377	0.614	0.741	0.092	0.030	0.444	0.547	0.402	0.756	0.369	0.610	0.741	0.103	0.040	0.446
CFM w/ DINO-v2 + FPFH	RGB + Point Cloud	0.597	0.415	0.768	0.505	0.640	0.743	0.315	0.321	0.538	0.523	0.413	0.753	0.542	0.621	0.741	0.222	0.314	0.516
M3DM w/ DINO-v2 + FPFH	RGB + Point Cloud	0.733	0.452	0.767	0.702	0.702	0.752	0.288	0.126	0.565	0.690	0.454	0.770	0.461	0.645	0.734	0.318	0.132	0.526
AST	RGB + Depth	**0.750**	**0.503**	**0.792**	**0.807**	**0.716**	**0.764**	**0.467**	**0.798**	**0.699**	**0.751**	**0.502**	**0.793**	0.805	**0.717**	**0.764**	0.450	**0.785**	**0.695**

Table 4. **Anomaly segmentation results (V-AUPRO@1%).** Best results in **bold**, runner-ups underlined.

Method	Modality	Features		real2real		synth2real	
		2D	3D	Detection	Segmentation	Detection	Segmentation
PatchCore	RGB	WRN-101	–	**0.754**	0.630	0.451	**0.600**
	RGB	DINO-v2	–	0.678	**0.671**	**0.540**	0.596
	Depth	–	WRN-101	0.582	0.486	0.507	0.484
	Depth	–	DINO-v2	0.600	0.496	0.246	0.471
	Point Cloud	–	FPFH	0.415	0.464	0.313	0.460
BTF	RGB + Point Cloud	WRN-101	FPFH	0.478	0.444	0.422	0.446
	RGB + Depth	WRN-101	WRN-101	**0.707**	**0.609**	**0.448**	**0.543**
CFM	RGB + Point Cloud	DINO-v2	FPFH	0.448	0.538	0.273	0.516
	RGB + Depth	DINO-v2	DINO-v2	**0.548**	**0.663**	**0.311**	**0.620**
M3DM	RGB + Point Cloud	DINO-v2	FPFH	0.621	**0.565**	**0.402**	**0.526**
	RGB + Depth	DINO-v2	DINO-v2	**0.659**	0.503	0.254	0.476
AST	RGB	EffNet-B5	–	0.584	0.687	**0.544**	0.679
	RGB + Depth	EffNet-B5	EffNet-B5	**0.687**	**0.699**	0.495	**0.695**

Table 5. **Anomaly detection and segmentation mean results with different input modalities.** Best results per method in **bold**.

within the same modality that require substantial training, i.e., EfficientAD and CFM. We argue that this may be ascribed to the single-instance setup, which renders the latter category of approaches harder to optimise due to the limited number of training images, which are as many as the views. The methods most effectively extended to our multiview 3D task are Patchcore, which, equipped with either WRN-101 or DINO-v2, can achieve the best detection in both setups, and AST, achieving the best segmentation performance in both setups. Indeed, Patchcore yields I-AUROCs of 0.754 (WRN-101) and 0.540 (DINO-v2), while AST yields a V-AUPRO@1% of 0.699 and 0.695, in the `real2real` and `synth2real` setups, respectively. It is also worth pointing out how, due to the domain shift, we can notice a gap between the `real2real` and `synth2real` setups, especially in detection (0.754 vs. 0.540, I-AUROC) compared to segmentation performance (0.699 vs. 0.695, V-AUPRO@1%). The qualitative results, shown in Fig. 5, visually support the above observations. Overall, the relatively low, definitely far from saturated, performance figures provided by baselines designed by straightforward extension of state-of-the-art methods delivering impressive results in singleview 2D benchmarks hint at the novel and challenging nature of the task set forth by SiM3D. Thus, our experimental findings by baseline approaches reveal the need for more advanced and specific methods and paradigms, both in the `real2real` and, even more so, the `synth2real` setups.

Alternative 3D Representations. Given the lack of feature extractors amenable to processing high-resolution point clouds, we explore the use of depth maps to capture 3D cues. The 2D grid structure of depth maps enables the deployment of popular backbones pre-trained on RGB data that support high-resolution inputs, such as DINO-v2. Thus, we create and evaluate additional baselines using depth maps and report the results in Tab. 5. Similarly to greyscale images, we tile depth maps along the channel axis to ensure compatibility with the employed backbones. The results show that the use of depth maps rather than point clouds in multimodal methods tends to improve detection and segmentation performance by sizable margins, e.g., in the `real2real` setup, BTF yields 0.707 I-AUROC and 0.609 V-AUPRO when processing depths compared to 0.478 I-AUROC and 0.444 V-AUPRO when fed with point clouds (row 6 vs. row 7) while relying on the same WRN-101 image backbone. Indeed, all multimodal methods but M3DM achieve their best performance in both tasks and setups by processing depth maps instead of point clouds. Yet, as vouched by the results achieved by PatchCore, RGB images compare favourably to depth maps in terms of detection performance. We ascribe this to the feature extractors used to compute depth features being pre-trained on RGB images, which may be conducive to yielding weaker features when fed with depth inputs. This points out an intriguing research question: would it be possible to develop a foundation model to process depth maps effectively?

6. Final Discussion

We introduced SiM3D, the first benchmark and dataset focused on integrating multiview and multimodal information for comprehensive 3D ADS. Our experiments highlight several open challenges. First, a need for methods that can process multiview inputs more effectively, e.g., by taking all views as input rather than integrating singleview outputs, and that can be trained with a very limited number of nominal samples, e.g., a single instance. Second, training with synthetic data requires more robust techniques to address domain-shift challenges. Finally, the current literature lacks 3D backbones capable of processing high-resolution point clouds. Employing depth maps seems a promising strategy to facilitate this processing, yet a foundational model for depth maps is still missing. We believe that SiM3D may serve as a strong foundation for tackling these challenges and fostering future research toward 3D ADS.

Acknowledgment

We acknowledge financial support from the National Recovery and Resilience Plan (NRRP), Mission 4, Component 2, Investment 1.1, Call No. 1409 published on 14 September 2022 by the Italian Ministry of University and Research (MUR), funded by the European Union.

This work was also partially supported by the FSE+ 2021–2027 program under Article 24, paragraph 3, letter a) of Law 240/2010 and its subsequent amendments, as well as by Regional Resolution No. 693/2023 (Project Code: 2023-20090/RER), Grant CUP: J19J23000730002.

We sincerely thank Lucia Gasperini for her contribution to dataset acquisition and labelling. We also extend our gratitude to Manuel Turrini for his support in setting up the acquisition process.

References

[1] Kilian Batzner, Lars Heckler, and Rebecca König. Efficientad: Accurate visual anomaly detection at millisecond-level latencies. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 128–138, 2024. 3, 6

[2] Liron Bergman, Niv Cohen, and Yedid Hoshen. Deep nearest neighbor anomaly detection. *arXiv preprint arXiv:2002.10445*, 2020. 3

[3] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. *arXiv preprint arXiv:1807.02011*, 2018. 3

[4] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad – a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 3, 6

[5] Paul Bergmann, Kilian Batzner, Michael Fauser, David Sattlegger, and Carsten Steger. Beyond dents and scratches: Logical constraints in unsupervised anomaly detection and localization. *International Journal of Computer Vision*, 130, 2022. 3

[6] Paul Bergmann, Jin Xin, 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*, pages 202–213, 2022. 1, 2, 3, 7

[7] 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 16th Asian Conference on Computer Vision (ACCV2022)*, 2022. ACCV. 1, 2, 3, 7

[8] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021. 3

[9] Li-Ling Chiu and Shang-Hong Lai. Self-supervised normalizing flows for image anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2926–2935, 2023. 3

[10] Niv Cohen and Yedid Hoshen. Sub-image anomaly detection with deep pyramid correspondences. *ArXiv*, 2020. 3

[11] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2024. 5

[12] CloudCompare Online Community. *CloudCompare - 3D point cloud and mesh processing software Open Source Project*. EDF, 2024. 5

[13] Alex Costanzino, Pierluigi Zama Ramirez, Mirko Del Moro, Agostino Aiezzo, Giuseppe Lisanti, Samuele Salti, and Luigi Di Stefano. Test time training for industrial anomaly segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 3910–3920, 2024. 3

[14] Alex Costanzino, Pierluigi Zama Ramirez, Giuseppe Lisanti, and Luigi Di Stefano. Multimodal industrial anomaly detection by crossmodal feature mapping. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2024. CVPR. 1, 3, 6

[15] 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. 3

[16] 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*, pages 98–107, 2022. 3

[17] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 3, 6

[18] 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, 6

[19] Jinlei Hou, Yingying Zhang, Qiaoyong Zhong, Di Xie, Shiliang Pu, and Hong Zhou. Divide-and-assemble: Learning block-wise memory for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8791–8800, 2021. 3

[20] 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. 3

[21] Jiaqi Liu, Guoyang Xie, Xinpeng Li, Jinbao Wang, Yong Liu, Chengjie Wang, Feng Zheng, et al. Real3d-ad: A dataset of point cloud anomaly detection. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. 3

[22] Jiaqi Liu, Guoyang Xie, Jingbao Wang, Shangnian Li, Chengjie Wang, Feng Zheng, and Yaochu Jin. Deep industrial image anomaly detection: A survey. *arXiv preprint arXiv:2301.11514*, 2, 2023. 3

[23] Fabio Valerio Massoli, Fabrizio Falchi, Alperen Kantarci, Şeymanur Akti, Hazim Kemal Ekenel, and Giuseppe Amato. Mocca: Multilayer one-class classification for anomaly detection. *IEEE Transactions on Neural Networks and Learning Systems*, 33(6):2313–2323, 2021. 3

[24] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023. 3, 6

[25] 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. 3

[26] Jonathan Pirnay and Keng Chai. Inpainting transformer for anomaly detection. In *International Conference on Image Analysis and Processing*, pages 394–406. Springer, 2022. 3

[27] Tal Reiss, Niv Cohen, Liron Bergman, and Yedid Hoshen. Panda: Adapting pretrained features for anomaly detection and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2806–2814, 2021. 3

[28] Oliver Rippel, Patrick Mertens, and Dorit Merhof. Modeling the distribution of normal data in pre-trained deep features for anomaly detection. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 6726–6733. IEEE, 2021.

[29] Nicolae-Cătălin 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*, pages 13576–13586, 2022. 3

[30] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of 2022 IEEE Conference on Computer Vision and Pattern Recognition*, pages 14298–14308, 2022. 1, 3, 6

[31] Marco Rudolph, Bastian Wandt, and Bodo Rosenhahn. Same same but differnet: Semi-supervised defect detection with normalizing flows. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1907–1916, 2021. 3

[32] Marco Rudolph, Tom Wehrbein, Bodo Rosenhahn, and Bastian Wandt. Asymmetric student-teacher networks for industrial anomaly detection. In *Winter Conference on Applications of Computer Vision (WACV)*, 2023. 3

[33] 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, 2009. 3, 6

[34] Kihyuk Sohn, Chun-Liang Li, Jinsung Yoon, Minho Jin, and Tomas Pfister. Learning and evaluating representations for deep one-class classification. In *International Conference on Learning Representations*, 2021. 3

[35] Chengjie Wang, Wenbing Zhu, Bin-Bin Gao, Zhenye Gan, Jiangning Zhang, Zhihao Gu, Shuguang Qian, Mingang Chen, and Lizhuang Ma. Real-iad: A real-world multi-view dataset for benchmarking versatile industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22883–22892, 2024. 1, 3

[36] 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. 1, 3, 6

[37] Julian Wyatt, Adam Leach, Sebastian M Schmon, and Chris G Willcocks. Anoddpm: Anomaly detection with denoising diffusion probabilistic models using simplex noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 650–656, 2022. 3

[38] Minghui Yang, Peng Wu, and Hui Feng. Memseg: A semi-supervised method for image surface defect detection using differences and commonalities. *Engineering Applications of Artificial Intelligence*, 119:105835, 2023.

[39] Jihun Yi and Sungroh Yoon. Patch svdd: Patch-level svdd for anomaly detection and segmentation. In *Proceedings of the Asian conference on computer vision*, 2020.

[40] Seungdong Yoa, Seungjun Lee, Chiyoon Kim, and Hyunwoo J Kim. Self-supervised learning for anomaly detection with dynamic local augmentation. *IEEE Access*, 9:147201–147211, 2021.

[41] Jiawei Yu, Ye Zheng, Xiang Wang, Wei Li, Yushuang Wu, Rui Zhao, and Liwei Wu. Fastflow: Unsupervised anomaly detection and localization via 2d normalizing flows. *arXiv preprint arXiv:2111.07677*, 2021. 3

[42] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *BMVC*, 2016. 6

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

[44] Zheng Zhang and Xiaogang Deng. Anomaly detection using improved deep svdd model with data structure preservation. *Pattern Recognition Letters*, 148:1–6, 2021. 3

[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. *arXiv preprint arXiv:2207.14315*, 2022. 3