Self-supervised Feature Adaptation for 3D Industrial Anomaly Detection

Yuanpeng Tu[1*], Boshen Zhang[2*], and Liang Liu[2*], Yuxi Li[2], Jiangning Zhang[2], Yabiao Wang[3,2†], Chengjie Wang[2], Cairong Zhao[1†]

[1] Tongji University, Shanghai
{2030809, zhaocairong}@tongji.edu.cn
[2] YouTu Lab, Tencent, Shanghai
{boshenzhang,leoneliu,yukiyxli,vtzhang,caseywang,jasoncjwang}@tencent.com
[3] Zhejiang University

Abstract. Industrial anomaly detection is generally addressed as an unsupervised task that aims at locating defects with only normal training samples. Recently, numerous 2D anomaly detection methods have been proposed and have achieved promising results, however, using only the 2D RGB data as input is not sufficient to identify imperceptible geometric surface anomalies. Hence, in this work, we focus on multi-modal anomaly detection. Specifically, we investigate early multi-modal approaches that attempted to utilize models pre-trained on large-scale visual datasets, *i.e.*, ImageNet, to construct feature databases. And we empirically find that directly using these pre-trained models is not optimal, it can either fail to detect subtle defects or mistake abnormal features as normal ones. This may be attributed to the domain gap between target industrial data and source data. Towards this problem, we propose a Local-to-global Self-supervised Feature Adaptation (LSFA) method to finetune the adaptors and learn task-oriented representation toward anomaly detection. Both intra-modal adaptation and cross-modal alignment are optimized from a local-to-global perspective in LSFA to ensure the representation quality and consistency in the inference stage. Extensive experiments demonstrate that our method not only brings a significant performance boost to feature embedding based approaches, but also outperforms previous State-of-The-Art (SoTA) methods prominently on both MVTec-3D AD and Eyecandies datasets, e.g., LSFA achieves 97.1% I-AUROC on MVTec-3D, surpass previous SoTA by **+3.4%**. Code is available at `https://github.com/yuanpengtu/LSFA`.

Keywords: Self-supervision · Anomaly detection · Multi-modality

1 Introduction

Industrial anomaly detection is a widely-explored computer vision task, aiming at detecting unusual image-level/pixel-level patterns in industrial products [28].

* Equal contribution.
† Corresponding author.

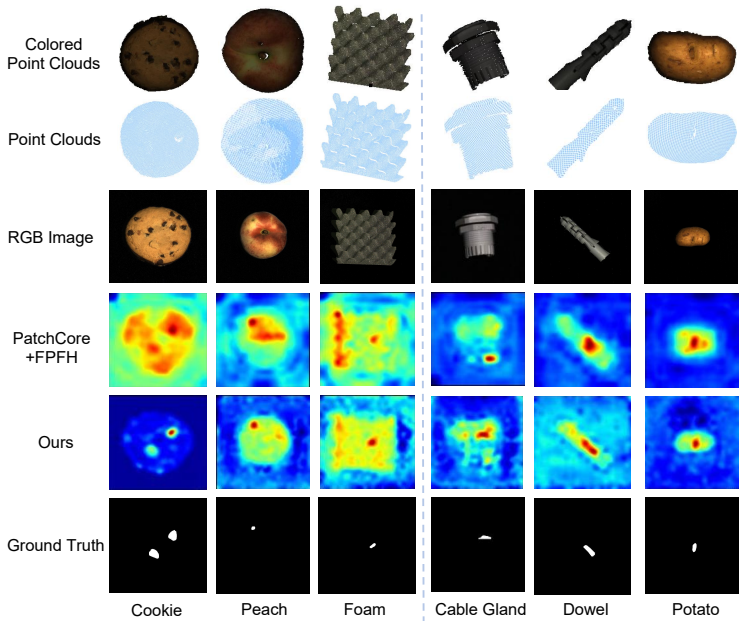

Fig. 1: Illustrations of MVTec-3D AD dataset [11]. The second and third rows are the input point cloud data and RGB data. The fourth and fifth rows are prediction results. Our method can avoid the overestimation issues (as shown left) and produce more accurate results for categories with complex textures (as shown right).

Since the lack of anomalous samples in real-world scenarios, current anomaly detection methods usually follow unsupervised paradigm [9, 18, 21, 24, 27, 31, 32, 36–39], *i.e.*, training with normal samples but testing on the mixed normal and abnormal samples. Most of the previous methods [18, 36, 40] are designed for 2D images and have achieved great success in 2D anomaly detection. However, in the scenarios of industrial inspection, due to lack of depth information, sometimes it is hard to differentiate between subtle surface defect and normal texture with only RGB information (e.g., cookie in Fig 1.). Therefore, recently there appears new benchmarks [11, 13] to encourage anomaly detection research in a multi-model view, where the objects are represented with both 2D images and 3D point clouds. To perform precise anomaly localization, existing 2D anomaly detection approaches can be roughly categorized into two families: reconstruction based and feature embedding based. The former utilizes the characteristic that a generator trained with only normal features cannot successfully reconstruct abnormal features. While the latter aims to model the distribution of normal samples through a well-trained feature extractor, and in inference stage, the out-of-distribution samples are treated as anomalies. The feature embedding based family [1, 4] is more flexible and show promising performance on 2D RGB anomaly detection task. However, simply transferring the 2D feature embedding paradigm into the 3D domain is not easy. Taking the state-of-the-

art embedding based method PatchCore [32] as an example, when combined with handcrafted 3D representations (FPFH [23]), it yields a strong multimodal anomaly detection baseline. However, as shown in Fig. 1, we experimentally find that the PatchCore+FPFH baseline shows two drawbacks, *First,* it is prone to mistake abnormal regions as normal ones due to the large discrepancy between pretrained knowledge and industrial scenes (see the left part in Fig. 1). *Second,* it sometimes fails to identify small anomaly patterns when it comes to categories with more complex textures, as shown in the right part in Fig. 1.

To address the aforementioned problems, we resort to a feature adaptation strategy to further enhance the capacity of pre-trained models and learn task-oriented feature descriptors. *In terms of modality*, color is more effective to identify texture anomalies, while depth information can be helpful to detect geometric deformations in 3D space [23], thus it is more advisable to leverage both the intra-modal and cross-modal information for adaptation. On the other hand, *in terms of granularity*, the object-level correspondence between modalities helps to learn compact representation, while anomaly detection requires local sensitivity to identify subtle anomalies [32], hence a multi-grained learning objective is necessary. With these consideration above, we propose a novel **L**ocal-to-global **S**elf-supervised multi-modal **F**eature **A**daptation framework, named LSFA, to better transfer the pre-trained knowledge to downstream anomaly detection task. Specifically, LSFA performs adaptation from two views: intra-modality and cross-modality. The former adaptation introduces Intra-modal Feature Compactness (IFC) optimization, where multi-grained memory banks are applied to learn compact distribution of normal features. As for the latter one, Cross-modal Local-to-global Consistency (CLC) is designed to align features from different modality in both patch-level and object-level. With the help of multi-grained information from both modality, model adapted with LSFA yields target-oriented features toward anomaly detection in 3D space, thus it is capable of capturing small anomalies, while avoiding false positives (shown in Fig. 1). For the final inference of anomaly detection, we leverage the fine-tuned features by LSFA to construct memory bank and determine normal/anomaly by computing the feature difference as in [32]. The effectiveness of LSFA is verified on mainstream benchmarks, including MVTec-3D and Eyecandies. Where LSFA outperforms previous SoTA [34] by a large margin, *i.e.*, it obtains 97.1% (+**3.4%**) I-AUROC on MVTec-3D. To summarize, the key contributions of this work are as follows:

• We propose LSFA, a novel and effective framework towards 3D anomaly detection, it adapts the pre-trained features with local-to-global correspondence between modalities as supervision. It shows significant advantages on mainstream benchmarks and sets the new state-of-the-art record.

• In LSFA, Intra-modal Feature Compactness optimization (IFC) is proposed to improve feature compactness from both patch-wise and prototype-wise with dynamic-updated memory banks.

• In LSFA, Cross-modal Local-to-global Consistency alignment (CLC) is proposed to alleviate cross-modal misalignment and enhance local-sensitivity of representations with multi-granularity contrastive signals.

2 Related Work

2D industrial anomaly detection. As a binary classification task, unsupervised anomaly detection only trains models with normal samples to distinguish instances sampled from normal/anomaly distribution, which has drawn extensive attention [28]. Existing methods mainly consist of two classes: reconstruction based and feature embedding based. For the former, knowledge-distillation based ones [10, 18, 35] assume that there exists a difference between the pre-trained teacher model and the student model in the anomalous patch-level features. [8, 20] detect defects by comparing the reconstructed images and input ones. Besides these methods, feature embedding based methods recently have achieved superior performance with features extracted from models pre-trained on natural image datasets, *i.e.*, ImageNet. Normalizing flow [22, 40] based ones distinguish defects by transforming normal features into Normal distribution. PatchCore [32] stores normal patch-level features for localizing defects by comparing the target and normal features. CFA [26] proposes a coupled-hypersphere fine-tuning framework to adapt patch features to the target dataset.
3D industrial anomaly detection. Different from 2D anomaly detection, 3D industrial anomaly detection identifies anomaly patches by taking both RGB and point cloud samples into consideration. [11] introduces the first public 3D anomaly detection benchmark, MVTec-3D AD for evaluation of methods. [12] proposes a 3D teacher-student framework to extract local-geometry aware descriptors for point clouds. [23] firstly explores the appliance of memory bank on this task and utilizes local geometry features extracted from pre-trained models. M3DM [2] proposes a multimodal industrial anomaly detection method with hybrid feature fusion to promote interaction between multimodal features. 3DSR [41] proposes a depth-Aware discrete auto-encoder, that enables learning a joint discrete latent space. However, these methods generally perform cross-modal alignment while overlook the importance of intra-modal feature compactness. Therefore, their extracted single-modal features are likely to form a distribution where the anomalous/normal features are difficult to be separated from each other. Such feature distribution limits their ability to effectively integrate information from both modalities as well, leading to inaccurate anomaly detection. Additionally, these methods only consider local-level cross-modal alignment without incorporating global-level alignment of features, which is also crucial for enhancing information interaction between the two modalities. Motivated by this, we propose a local-to-global self-supervised multimodal adaption method to boost the voxel-level detection performance of feature embedding based approaches from both patch-level and object-level views.

3 Methodology

3.1 Overview

Framework overview and symbol definition. In this section, we first give out the overview of our LSFA framework. As shown in Fig. 2, LSFA takes both

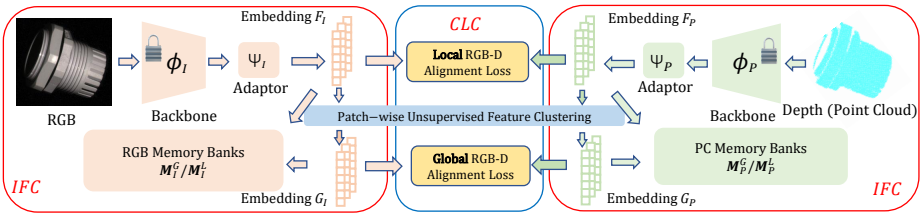

Fig. 2: The pipeline of LSFA. The features of two modalities are adapted from two views: Intra-modal Feature Compactness optimization (IFC) and Cross-modal Local-to-global Consistency alignment (CLC). The fine-tuned results of the adaptors are used for defect localization.

point clouds and RGB images in $\mathbb{D} = \{(P_i, I_i)\}_{i=1}^{|\mathbb{D}|}$ as input for joint defect detection, where $P_i \in \mathbb{R}^{N\times 3}$ and $I_i \in \mathbb{R}^{H\times W\times 3}$. For both modality representation P_i and I_i, a pretrained feature extractor ϕ_P/ϕ_I is applied to obtain modality-specific representation. Since there exists severe domain bias between pre-trained backbones and downstream detection task, a vanilla transformer encoder layer [19] is utilized as the adaptor for these features (note that several other adaptor structures are also investigated in our appendix). The adaptors for RGB/3D modalities are denoted as $\Psi_I(\cdot)/\Psi_P(\cdot)$, we propose to perform task-oriented feature adaptation for $\Psi_I(\cdot)/\Psi_P(\cdot)$ from two views: Intra-modal Feature Compactness optimization (IFC) and Cross-modal Local-to-global Consistency alignment (CLC). (1) IFC constructs both global-level and local-level dynamic-updated memory banks for both RGB/3D modality to minimize the distance between normal features from the multi-granularity view, leading to better distinction between normal and abnormal features. (2) CLC consists of local-to-global cross-modal alignment modules, which alleviates feature misalignment between two modalities and enhances the multi-modal information interaction of spatial structures with self-supervised signals.

Inference with adapted representation. After the adaptation process, since local-sensitive features are more useful for detecting anomaly patterns, the global features are discarded for inference. For either modality of RGB or Point Cloud, only the local features from adaptor is utilized to calculate the anomaly score of each pixel/voxel through off-the-shelf PatchCore [32] algorithm. Finally both anomaly scores from two modalities are averaged as the final anomaly estimation.

3.2 CLC: Cross-modal Local-to-global Consistency Alignment

Feature projection. To extract local-sensitive features for anomaly detection, the ViT [19] and PointMAE [29] are utilized as ϕ_I/ϕ_P. ViT splits 2D image I_i into N_m patches and extract deep feature for each patch, correspondingly, PointMAE group 3D points from P_i into N_d groups and extract group-wise feature. To build dense local correspondence between two modality, we remap 3D points into 2D patches via geometric interpolation and projection. Specifically, we denote the deep feature of i-th point group as A_i and the group center is

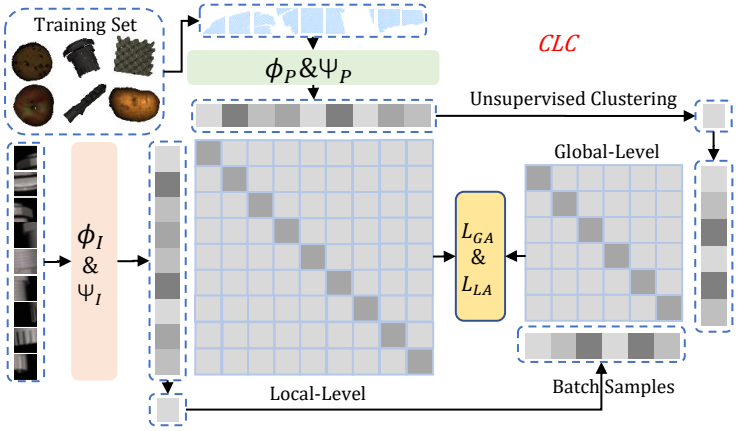

Fig. 3: The proposed inter-modal local-to-global consistency alignment. For the local view, similarity of path-wise features in the same/different location of the RGB image and its corresponding 3D point cloud is maximized/minimized to guarantee local-geometry consistency of two modalities. For global view, instance-wise features clustered from patch-wise features are optimized similarly.

denoted as $c_i \in \mathbb{R}^3$, then for each point $p \in \mathbb{R}^3$ in P_i, a point-wise deep feature f_p can be obtained via distance-based interpolation:

$$f_p = \sum_{i=1}^{N_d} \alpha_i A_i, \quad \alpha_i = \frac{\frac{1}{\|c_i - p\|_2}}{\sum_{k=1}^{N_m} \frac{1}{\|c_k - p\|_2}}. \tag{1}$$

Meanwhile, we can verify whether a 3D point p is projected into a 2D patch with camera parameters, thus for each image patch from ViT, we average the feature f_p of all points projected into the same patch as 2D projection of original point-cloud features. By this means, we obtain a 2D patch-wise representation of 3D point features, which shares the same patch number N_m as image features, and the local correspondence is naturally obtained by associating RGB features and projected point features of the same patch. Finally, both patch-wise representation of RGB and point cloud are fed to adaptor $\Psi_I(\cdot)/\Psi_P(\cdot)$ respectively. The adapted features are denoted as $\mathbb{D}_F = \{(F_{P_i}, F_{I_i})\}_{i=1}^{|\mathbb{D}|}$.

Cross-modal local-to-global consistency alignment. The features of two modalities are aligned in spatial location after the previous step. However, without cross-modal interaction in the adaption process, cross-modal feature misalignment may lead to inferior results when fusing scores of two modalities during the inference stage. To address this issue, as shown in Fig. 3, we perform local-to-global consistency alignment, which can utilize the cross-modal self-supervised signals to enhance feature quality.

Specifically, the adapted patch-wise features for both RGB/3D point clouds $\{F_{I_i}, F_{P_i}\}_{i=1}^{N_b}$ are first mapped into the same dimension with two fully-connected layers, denoted as H_I/H_P, where N_b is the batch size. The projected features

are denoted as $\{F'_{I_i}, F'_{P_i}\}_{i=1}^{N_b}$, then a patch-wise contrastive loss is calculated to maximize the feature similarity between patches from different modal but the same location, and minimize similarity between patches from different location:

$$\mathcal{L}_{LA} = -\log\left(\frac{\exp\left(\left\langle F'^{j}_{I_i}, F'^{j}_{P_i}\right\rangle\right)}{\sum_{t=1}^{N_m}\sum_{k=1}^{N_m}\exp\left(\left\langle F'^{t}_{I_i}, F'^{k}_{P_i}\right\rangle\right)}\right), \tag{2}$$

where $\langle\cdot,\cdot\rangle$ denotes the innder production between vectors. Since Eq. 2 only involves local geometry clues while lacking the interaction of global structural information, we further clustering the local feature F_{I_i}/F_{P_i} to obtain an instance-wise feature G_{I_i}/G_{P_i} with the k-means clustering algorithm. And then performing a similar operation on this global features, the corresponding **G**lobal **A**lignment loss is denoted as L_{GA}.

$$\mathcal{L}_{GA} = -\log\left(\frac{\exp\left(\left\langle G'_{I_i}, G'_{P_i}\right\rangle\right)}{\sum_{t=1}^{N_b}\sum_{x=1}^{N_b}\exp\left(\left\langle G'_{I_t}, G'_{P_x}\right\rangle\right)}\right). \tag{3}$$

Thus the overall loss function for CLC is formulated as:

$$\mathcal{L}_{\mathrm{CLC}} = \mathcal{L}_{\mathrm{LA}} + \mathcal{L}_{\mathrm{GA}}. \tag{4}$$

3.3 IFC: Intra-modal Feature Compactness Optimization

The proposed intra-modal feature compactness optimization strategy aims at helping models generate more compact representation for normal samples, thus making models more sensitive to anomaly patterns.

Local-to-global compactness optimization. Since there exists severe domain bias for the pre-trained models without adaptation, the extracted features are likely to form a distribution where the anomalous/normal features are difficult to be separated from each other. Consequently, previous feature embedding based methods [23] are inevitably prone to mistake anomalies as normal areas. Motivated by this, as shown in Fig. 4, we design a dynamic-updated memory-bank in both local and global level to guide compactness optimization.

Since the optimization is conducted within each modality, here we take RGB feature as an example and the point-cloud feature is processed in a similar manner. Concretely, we denote the memory bank consisting of patch-level RGB features as M_I^L with length $|M_I^L| = n_I^L$. The j-th patch-level feature $F_{I_i}^j$ of I_i in batch $\{F_{I_i}\}_{i=1}^{N_b}$ is utilized for nearest neighbor searching in M_I^L, where N_b is the batch size. A mean squared error loss is utilized to minimize the discrepancy between $F_{I_i}^j$ and its corresponding nearest item in M_I^L. Thus the **L**ocal patch-level **C**ompactness $\mathcal{L}_{\mathrm{LC}}$ loss can be derived as follows:

$$\mathcal{L}_{\mathrm{LC}} = \sum_{i=1}^{N_b}\sum_{j=1}^{N_m}\min_{\mathrm{Q}\in\mathrm{M_I^L}}\left\|F_{I_i}^j - Q\right\|_2. \tag{5}$$

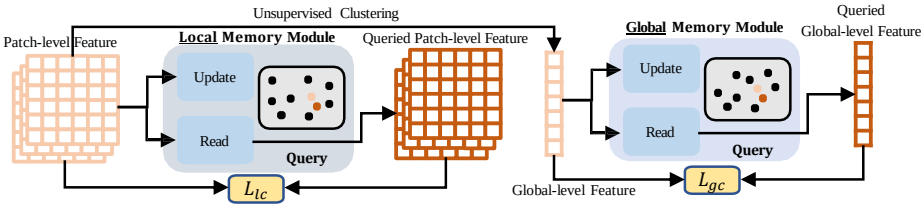

Fig. 4: The local-to-global compactness optimization strategy, where both prototype-wise global-level and patch-wise local-level memory banks are involved.

Where N_m is the patch number. Furthermore, to enhance the compactness of features for each category, a global compactness loss is designed to simultaneously optimize the global feature G_{I_i}. Denote the memory bank consisting of global RGB features with length n_I^G as M_I^G. A similar nearest neighbor search operation is performed for G_{I_i} and M_I^G to enhance sensitivity against anomalies from the global view. Therefore, the **G**lobal **C**ompactness loss $\mathcal{L}_{\mathrm{GC}}$ is:

$$\mathcal{L}_{\mathrm{GC}} = \sum_{i=1}^{N_b} \min_{\mathrm{Q} \in \mathrm{M_I^G}} \left\| G_{I_i}^j - Q \right\|_2 . \tag{6}$$

After each iteration, the local-level/global-level features of current batch samples are enqueued into M_I^L/M_I^G respectively, which can be derived as:

$$\begin{cases} M_I^L = M_I^L \cup \{F_{I_i}^j | j \in [1, N_m], i \in [1, N_b]\} \\ M_I^G = M_I^G \cup \{G_{I_i} | i \in [1, N_b]\}. \end{cases} \tag{7}$$

Meanwhile, the least recently appended features with the same length as the enqueued features will be popped out from M_I^L/M_I^G to keep the features in banks up-to-date when the length of M_I^L/M_I^G is larger than n_I^G/n_I^L. Similar global and local compactness optimization operations are performed for the point cloud features $\{F_{P_i}\}_{i=1}^{N_b}$ as well, where the global and local memory bank sizes of point cloud features are the same as RGB modality. Consequently, the loss function of the proposed IFC can be summarized as:

$$\mathcal{L}_{\mathrm{IFC}} = \mathcal{L}_{\mathrm{LC}} + \mathcal{L}_{\mathrm{GC}}. \tag{8}$$

Therefore to summarize, the overall training loss for our LSFA is derived as:

$$\mathcal{L}_{\mathrm{LSFA}} = \mathcal{L}_{\mathrm{IFC}} + \lambda \mathcal{L}_{\mathrm{CLC}}. \tag{9}$$

Where λ is a balancing hyper-parameter.

3.4 Defect Localization

Since LSFA is designed for adapting pre-trained features to estimate anomaly patterns better. We utilize the pre-trained backbones and the adaptors for final feature extraction. The adapted features of two modalities are respectively

Algorithm 1: Training for the proposed LSFA.

Input: Memory banks $\{M_I^G, M_I^L, M_P^G, M_P^L\}$, adaptors $\{\Psi_I, \Psi_P\}$, linear projection layer $\{H_I, H_P\}$, training set features $\{F_I, F_P\}$.
Output: Parameters of adaptors $\{\Theta_I, \Theta_P\}$.

Initialize $M_I^G, M_I^L, M_P^G, M_P^L$.
for $F_{I_i}, F_{P_i} \in \mathbb{D}_F$ **do**
  $F'_{I_i} \longleftarrow H_I(F_{I_i})$; $F'_{P_i} \longleftarrow H_P(F_{P_i})$ `/* Inter-modal Local-to-global Consistency Alignment */`
  $\Theta_I, \Theta_P \overset{optim}{\longleftarrow} \mathcal{L}_{\mathrm{CLC}}(F'_{I_i}; F'_{P_i}; \Theta_P; \Theta_I)$ `/* Cross-modal Feature Compactness Optimization */`
  $\Theta_I \overset{optim}{\longleftarrow} \mathcal{L}_{\mathrm{IFC}}(F_{I_i}; M_I^G; M_I^L; \Theta_I)$ $\Theta_P \overset{optim}{\longleftarrow} \mathcal{L}_{\mathrm{IFC}}(F_{P_i}; M_P^G; M_P^L; \Theta_P)$ `/* Update Memory Banks */`
  $M_I^G, M_I^L \overset{update}{\longleftarrow} F_{I_i}$; $M_P^G, M_P^L \overset{update}{\longleftarrow} F_{P_i}$
end

fed into the off-the-shelf feature embedding based method PatchCore [32]. The anomaly scores of two modalities are averaged as the final anomaly score for each pixel/voxel to evaluate the effectiveness on anomaly detection. The overall pseudo-code of LSFA can be found in Algorithm 1.

Discussion. Since the framework of LSFA is similar to M3DM [2], here we discuss their difference in detail. First, rather than introducing extra modules for feature fusion in [2], we only perform feature adaptation for each modality and needs no extra memory bank, thus introducing no extra time and memory cost for inference. Moreover, M3DM overlooks the importance of object-level feature alignment to accurate anomaly detection. And our LSFA performs cross-modal feature alignment from both object-level and patch-level views to fully enhance the consistency and interaction of cross-modal discriminative information, thus demonstrating much superior performance to it. Finally, LSFA takes the intra-modal feature compactness into consideration, which is ignored in M3DM as well. Specifically, similar to cross-modal alignment, the intra-modal feature compactness optimization is also conducted from both patch-level and object-level perspectives to alleviate the influence of domain bias of pre-trained features and obtain high-quality single-modal features.

4 Experiments

4.1 Experimental Details

Dataset. Specifically, we conduct experiments on three 3D industrial anomaly detection datasets: MVTec-3D AD [11], Eyecandies [13] and Real3D-AD [3]. And we following the standard evaluation protocol for fair comparisons. Details of the datasets and evaluation protocols are discussed in the Appendix.

Implementation details. For the feature extractors of the RGB modality, a ViT-B/8 [19] with DINO [14] is adopted. The 768-dim output of the final layer is

Table 1: I-AUROC for anomaly detection of all categories of MVTec-3D AD. '*' denotes replacing its features with the same pre-trained features as LSFA for PatchCore. Results with confidence intervals of LSFA are shown in the Appendix.

	Method	Bagel	Cable Gland	Carrot	Cookie	Dowel	Foam	Peach	Potato	Rope	Tire	Mean
3D	Depth GAN [11]	0.530	0.376	0.607	0.603	0.497	0.484	0.595	0.489	0.536	0.521	0.523
	Depth AE [11]	0.468	0.731	0.497	0.673	0.534	0.417	0.485	0.549	0.564	0.546	0.546
	Depth VM [11]	0.510	0.542	0.469	0.576	0.609	0.699	0.450	0.419	0.668	0.520	0.546
	Voxel GAN [11]	0.383	0.623	0.474	0.639	0.564	0.409	0.617	0.427	0.663	0.577	0.537
	Voxel AE [11]	0.693	0.425	0.515	0.790	0.494	0.558	0.537	0.484	0.639	0.583	0.571
	Voxel VM [11]	0.750	**0.747**	0.613	0.738	0.823	0.693	0.679	0.652	0.609	0.690	0.699
	3D-ST [12]	0.862	0.484	0.832	0.894	0.848	0.663	0.763	0.687	0.958	0.486	0.748
	FPFH [23]	0.825	0.551	0.952	0.797	0.883	0.582	0.758	0.889	0.929	0.653	0.782
	AST [34]	0.881	0.576	0.965	0.957	0.679	0.797	**0.990**	0.915	0.956	0.611	0.833
	FPFH*/M3DM [2]	0.941	0.651	0.965	0.969	0.905	0.760	0.880	**0.974**	0.926	0.765	0.874
	LSFA(Ours)	**0.986**	0.669	**0.973**	**0.990**	**0.950**	**0.802**	0.961	0.964	**0.967**	**0.944**	**0.921**
RGB	DifferNet [33]	0.859	0.703	0.643	0.435	0.797	0.790	0.787	0.643	0.715	0.590	0.696
	PADiM [17]	**0.975**	0.775	0.698	0.582	0.959	0.663	0.858	0.535	0.832	0.760	0.764
	PatchCore [32]	0.876	0.880	0.791	0.682	0.912	0.701	0.695	0.618	0.841	0.702	0.770
	STFPM [36]	0.930	0.847	0.890	0.575	0.947	0.766	0.710	0.598	0.965	0.701	0.793
	CS-Flow [22]	0.941	**0.930**	0.827	0.795	**0.990**	0.886	0.731	0.471	0.986	0.745	0.830
	AST [34]	0.947	0.928	0.851	**0.825**	0.981	**0.951**	0.895	0.613	**0.992**	**0.821**	**0.880**
	PatchCore*/M3DM [2]	0.944	0.918	0.896	0.749	0.959	0.767	0.919	0.648	0.938	0.767	0.850
	LSFA(Ours)	0.951	0.920	**0.911**	0.762	0.961	0.770	**0.930**	**0.675**	0.938	0.787	0.861
3D + RGB	Depth GAN [11]	0.538	0.372	0.580	0.603	0.430	0.534	0.642	0.601	0.443	0.577	0.532
	Depth AE [11]	0.648	0.502	0.650	0.488	0.805	0.522	0.712	0.529	0.540	0.552	0.595
	Depth VM [11]	0.513	0.551	0.477	0.581	0.617	0.716	0.450	0.421	0.598	0.623	0.555
	Voxel GAN [11]	0.680	0.324	0.565	0.399	0.497	0.482	0.566	0.579	0.601	0.482	0.517
	Voxel AE [11]	0.510	0.540	0.384	0.693	0.446	0.632	0.550	0.494	0.721	0.413	0.538
	Voxel VM [11]	0.553	0.772	0.484	0.701	0.751	0.578	0.480	0.466	0.689	0.611	0.609
	3D-ST [12]	0.950	0.483	**0.986**	0.921	0.905	0.632	0.945	**0.988**	0.976	0.542	0.833
	PatchCore + FPFH [23]	0.918	0.748	0.967	0.883	0.932	0.582	0.896	0.912	0.921	0.886	0.865
	AST [34]	0.983	0.873	0.976	0.971	0.932	0.885	0.974	0.981	**1.000**	0.797	0.937
	PatchCore*+FPFH* [23]	0.981	0.831	0.980	0.985	0.960	0.905	0.936	0.964	0.967	0.780	0.929
	M3DM [2]	0.994	0.909	0.972	0.976	0.960	0.942	0.973	0.899	0.972	0.850	0.945
	LSFA(Ours)	**1.000**	**0.939**	0.982	**0.989**	**0.961**	**0.951**	**0.983**	0.962	0.989	**0.951**	**0.971**

used and then pooled into 56×56 for subsequent training. For the 3D modality, a point transformer [30] pre-trained on ShapeNet [15] dataset is utilized and the outputs from 3/7/11 layer are concatenated to fuse multi-scale information. Details of the implementation are discussed in the Appendix.

4.2 Comparison on 3D AD Benchmark

To evaluate the effectiveness of our method, we first conduct experiments on both 3D/RGB/3D+RGB modality on MVTec-3D AD. Tab. 1 and Tab. 2 present the comparison results of I-AUROC and AUPRO, the methods are grouped by modality (we also report P-AUROC in the Appendix). 1) For the I-AUROC metric, our method can not only bring a significant boost to the baseline method on both single-modality benchmarks but also multi-modality combined ones, especially for the challenging categories, e.g., cable gland and tire. The single-modality results demonstrate that our intra-modal feature compactness optimization effectively improves the feature quality, thus benefiting the anomaly localization in the inference process. Moreover, our method significantly outperforms all previous methods regarding the average of all classes by a large margin of 4.7% for 3D, and 4.2% for the combination. A new state-of-the-art performance is achieved in 17 of all 30 cases for all the individual classes and data modalities. 2) For the AUPRO metric, LSFA can also achieve consistently higher

Table 2: AUPRO for anomaly segmentation of all categories of MVTec-3D. '*' denotes replacing its features with the same pre-trained features as LSFA for PatchCore. Results with confidence intervals of LSFA are shown in the Appendix.

	Method	Bagel	Cable Gland	Carrot	Cookie	Dowel	Foam	Peach	Potato	Rope	Tire	Mean
3D	Depth GAN [11]	0.111	0.072	0.212	0.174	0.160	0.128	0.003	0.042	0.446	0.075	0.143
	Depth AE [11]	0.147	0.069	0.293	0.217	0.207	0.181	0.164	0.066	0.545	0.142	0.203
	Depth VM [11]	0.280	0.374	0.243	0.526	0.485	0.314	0.199	0.388	0.543	0.385	0.374
	Voxel GAN [11]	0.440	0.453	0.875	0.755	0.782	0.378	0.392	0.639	0.775	0.389	0.583
	Voxel AE [11]	0.260	0.341	0.581	0.351	0.502	0.234	0.351	0.658	0.015	0.185	0.348
	Voxel VM [11]	0.453	0.343	0.521	0.697	0.680	0.284	0.349	0.634	0.616	0.346	0.492
	FPFH [23]	0.973	0.879	**0.982**	0.906	0.892	0.735	0.977	0.982	0.956	0.961	0.924
	FPFH*/M3DM [2]	0.943	0.818	0.977	0.882	0.881	0.743	0.958	0.974	0.950	0.929	0.906
	LSFA(Ours)	**0.974**	**0.887**	0.981	**0.921**	**0.901**	**0.773**	**0.982**	**0.983**	**0.959**	**0.981**	**0.934**
RGB	PatchCore [32]	0.901	0.949	0.928	0.877	0.892	0.563	0.904	0.932	0.908	0.906	0.876
	PatchCore*/M3DM [2]	0.952	0.972	**0.973**	0.891	0.932	0.843	**0.970**	0.956	0.968	**0.966**	0.942
	LSFA(Ours)	**0.957**	**0.976**	0.970	**0.912**	**0.934**	**0.851**	0.960	**0.957**	**0.970**	0.961	**0.945**
RGB + 3D	Depth GAN [11]	0.421	0.422	0.778	0.696	0.494	0.252	0.285	0.362	0.402	0.631	0.474
	Depth AE [11]	0.432	0.158	0.808	0.491	0.841	0.406	0.262	0.216	0.716	0.478	0.481
	Depth VM [11]	0.388	0.321	0.194	0.570	0.408	0.282	0.244	0.349	0.268	0.331	0.335
	Voxel GAN [11]	0.664	0.620	0.766	0.740	0.783	0.332	0.582	0.790	0.633	0.483	0.639
	Voxel AE [11]	0.467	0.750	0.808	0.550	0.765	0.473	0.721	0.918	0.019	0.170	0.564
	Voxel VM [11]	0.510	0.331	0.413	0.715	0.680	0.279	0.300	0.507	0.611	0.366	0.471
	3D-ST [12]	0.950	0.483	**0.986**	0.921	0.905	0.632	0.945	**0.988**	**0.976**	0.542	0.833
	PatchCore + FPFH [23]	0.976	0.969	0.979	**0.973**	0.933	0.888	0.975	0.981	0.950	0.971	0.959
	PacthCore*+FPFH* [23]	0.968	0.925	0.979	0.914	0.909	**0.948**	0.975	0.976	0.967	0.965	0.953
	M3DM [2]	0.970	0.971	0.979	0.950	**0.941**	0.932	0.977	0.971	0.971	0.975	0.964
	LSFA(Ours)	**0.986**	**0.974**	0.981	0.946	0.925	0.941	**0.983**	0.983	0.974	**0.983**	**0.968**

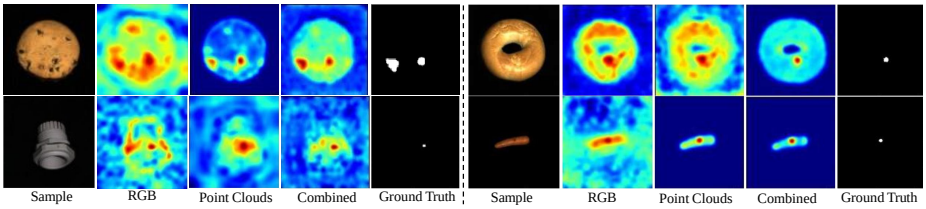

Fig. 5: Qualitative results of RGB/D modality.

scores than all previous methods for anomaly segmentation, demonstrating that our method is better at mining localized and detailed clues to discover crucial unexpected patterns. Besides MVTec-3D AD, we further perform a detailed evaluation on the latest large-scale 3D AD dataset Eyecandies. The corresponding results are shown in the Table 3, where our method obtains the best results and significantly outperforms all the previous approaches, achieving the average I-AUROC/AUPRO of 87.5% and 97.8% respectively for RGB modality. For the comparison of AUPRO results, it can be referred in the supplement, where the results on the Real3D-AD [3] dataset are available as well.

4.3 Ablation Study

To study the influence of each component within the proposed LSFA, we conduct ablation analysis on MVTec-3D.

Investigation on IFC. We first conduct studies to analyze the influence of the proposed IFC. The method that utilizes pre-trained features without adaptation

Table 3: I-AUROC score for anomaly segmentation of all categories of Eyecandies [13] dataset on RGB modality. '*' denotes using the same pre-trained backbone as LSFA.

	Method	Candy Cane	Choco late C.	Choco late P.	Confetto	Gummy Bear	Hazel nut T.	Licor ice S.	Lollipop	Marsh mallow	Peppe rmint C.	Mean
RGB	G [7]	0.485	0.512	0.532	0.504	0.558	0.486	0.467	0.511	0.481	0.528	0.507
	DFKDE [6]	0.539	0.577	0.482	0.548	0.541	0.492	0.524	0.602	0.658	0.591	0.555
	DFM [5]	0.532	0.776	0.624	0.675	0.681	0.596	0.685	0.618	0.964	0.770	0.692
	STFPM [36]	0.551	0.654	0.576	0.784	0.737	**0.790**	0.778	0.620	0.840	0.749	0.708
	PaDiM [17]	0.531	0.816	0.821	0.856	0.826	0.727	0.784	0.665	0.987	0.924	0.794
	AE [13]	0.527	0.848	0.772	0.734	0.590	0.508	0.693	0.760	0.851	0.730	0.701
	PatchCore*/M3DM [2]	0.648	0.949	0.941	**1.000**	0.878	0.632	0.933	0.811	**0.998**	**1.000**	0.879
	LSFA(Ours)	**0.681**	**0.958**	**0.945**	**1.000**	**0.883**	0.671	**0.939**	**0.824**	**0.998**	**1.000**	**0.890**
3D	M3DM [2]	0.482	0.589	0.805	0.845	**0.780**	0.538	0.766	0.827	0.800	0.822	0.725
	LSFA(Ours)	**0.517**	**0.602**	**0.847**	**0.850**	**0.780**	**0.589**	**0.773**	**0.830**	**0.811**	**0.843**	**0.744**
RGB+3D	AE [13]	0.529	0.861	0.739	0.752	0.594	0.498	0.679	0.651	0.838	0.750	0.690
	PatchCore*/M3DM [2]	0.624	**0.958**	0.958	**1.000**	0.886	0.758	**0.949**	0.836	**1.000**	**1.000**	0.897
	EasyNet [16]	**0.737**	0.934	0.866	0.966	0.717	**0.822**	0.847	**0.863**	0.977	0.960	0.869
	LSFA(Ours)	0.670	0.954	**0.961**	**1.000**	**0.913**	0.767	0.943	0.854	**1.000**	**1.000**	**0.906**

Table 4: Investigation on the loss functions within CLC.

Component		I-AUROC	AUPRO	P-AUROC
L_{GA}	L_{LA}			
✗	✗	0.929	0.953	0.987
✗	✓	0.949	0.961	0.989
✓	✗	0.952	0.961	0.990
✓	✓	**0.959**	**0.964**	**0.992**

Table 5: Ablation results for two components in LSFA, *i.e.*, IFC and CLC.

Component		I-AUROC	AUPRO	P-AUROC
IFC	CLC			
✗	✗	0.929	0.953	0.987
✗	✓	0.957	0.963	0.990
✓	✗	0.959	0.964	0.992
✓	✓	**0.971**	**0.968**	**0.993**

for PatchCore is used as the baseline for all the evaluations. As shown in Tab. 5, the baseline method achieves inferior accuracy for all the metrics with the fixed pre-trained features. By contrast, IFC brings a significant performance boost (about 2.8%/1.0%↑ for I-AUROC/AUPRO) by explicitly optimizing the feature compactness and keeping consistent with the inference process, which enhances the feature sensitivity to abnormal patterns. Tab. 6 shows a detailed analysis of each loss term within IFC, where both global and local compactness losses contribute to the final performance as well.

Investigation on CLC. We then investigate the influence of CLC. As shown in Tab. 5, CLC also achieves similar accuracy to IFC by performing multi-granularity cross-modal contrastive representation learning. This mainly accounts for that the proposed CLC can alleviate the impact of inter-modal misalignment from multiple views and meanwhile utilize the self-supervised signals for feature extraction. Similarly, Tab. 4 shows the results of each sub-component in CLC, where both global and local cross-modal contrastive losses boost performance over the baseline method. Moreover, further improvement in accuracy can be observed by combining IFC and CLC. Therefore, the above results verified the effectiveness of the proposed IFC, CLC, as well as their own key components.

Qualitative results. We conduct qualitative experiments to investigate the impact of RGB/3D modality. Fig. 5 shows the prediction results of single/combined-

Table 6: Investigation on the loss functions within IFC.

Component $\mathcal{L}_{GC}$	Component $\mathcal{L}_{LC}$	I-AUROC	AUPRO	P-AUROC
✗	✗	0.929	0.953	0.987
✓	✗	0.950	0.960	0.988
✗	✓	0.952	0.960	0.989
✓	✓	**0.957**	**0.963**	**0.990**

Table 7: Investigation on the structure of Ψ_I/Ψ_P.

Structure Ψ_I/Ψ_P	I-AUROC	AUPRO	P-AUROC
Linear projection	0.953	0.959	0.989
Single encoder layer	**0.974**	**0.968**	**0.993**
Two encoder layers	0.954	0.963	0.984
1×1 Convolution	0.951	0.962	0.986

Table 8: Training LSFA with LoRA/AdaLoRA on MVTec-3D.

	I-AUROC			AUPRO		
Method	3D	RGB	RGB+3D	3D	RGB	RGB+3D
LSFA-LoRA	91.06	85.43	93.91	92.17	93.97	95.16
LSFA-AdaLoRA	91.11	85.72	93.98	92.24	94.15	95.33

modality. It can be observed that the results of RGB modality are more dispersed and impose large scores in the edge regions. By contrast, the distribution of scores for 3D modality is more focused around the defects. Finally, the combined results demonstrate that both two modality helps precise defect localization.

Parameter sensitivity. Next, we evaluate the parameter sensitivity of important hyper-parameters in LSFA, including the size of the memory bank n_I^L and the balancing factor λ. As shown in Fig. 6 (left), LSFA achieves similar performance across all the sizes, thus not sensitive to n_I^L. To balance the performance and memory cost, we set $n_I^L = 5 \times 10^4$. For the λ, the results in Fig. 6 (right) demonstrate that LSFA is not sensitive to the value of λ as well. Since larger λ leads to a slight performance drop, we set $\lambda = 0.6$ to get the best results.

Investigation on adaptor structure. As shown in Tab. 7, besides the above experiments, we finally investigate the influence of different adaptor structures, including linear projection layer, single vanilla transformer encoder layer, multiple vanilla transformer encoder layers, and 1×1 convolution layer, where the single vanilla transformer encoder layer performs best among these structures.

4.4 Few-shot Anomaly Detection

To evaluate the effectiveness of LSFA in extreme cases, we conduct experiments on few-shot settings. Specifically, we randomly sample 5/10/50 images from each class as the training set and perform the evaluation on the whole test set. The results show that LSFA can also achieve superior performance, even compared with some of the methods trained with the whole training set in Tab. 9.

4.5 Comparison with Fine-tuning Methods

Here we remove the adaptors ϕ_I/ϕ_P and combine LSFA with off-the-shelf fine-tuning methods LoRA [25] and AdaLoRA [42] in PEFT. The results are shown

Table 9: Performance of LSFA under few-shot settings.

Method	I-AUROC	AUPRO	P-AUROC
5-shot	0.834	0.936	0.984
10-shot	0.871	0.943	0.987
50-shot	0.926	0.962	0.989
Full dataset	0.971	0.968	0.993

Table 10: Different fine-tuning schemes for RGB+3D modality on MVTec-3D. 'S-N'/'All' denotes training last N blocks/the whole network.

Metric	S-1	S-2	S-3	All
I-AUROC	95.42	94.26	92.14	84.57
AUPRO	96.01	95.45	95.21	90.15

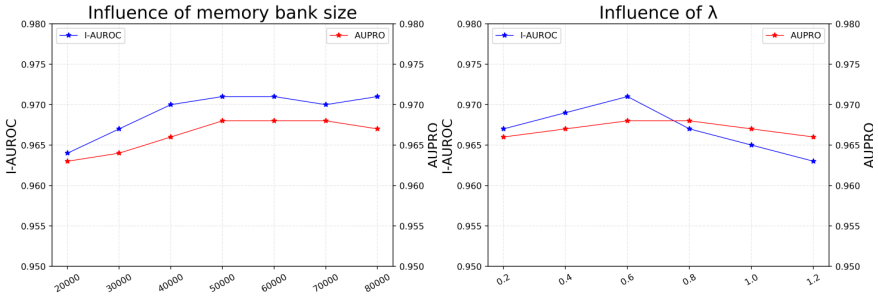

Fig. 6: Investigation on the influence of memory bank size n_I^L (left) and balancing hyber-parameter λ (right).

in Table. 8, which are slightly inferior to results of our LSFA. We remove the adaptors and evaluate the results of training the whole network and training the last few stages of the backbone network in our LSFA respectively. As shown in Table. 10, with more modules used for training, a more severe performance drop is observed, especially for training all the blocks. Such phenomenon indicates that training with only part/none of the modules fixed will result in severe catastrophic forgetting and over-fitting to specific data domains, thus failing to distinguish anomalies from normal patterns. Moreover, we provide comparison with fusion based methods in the Appendix, where our LSFA consistently outperforms the compared methods as well.

5 Conclusion

In this paper, we propose LSFA, a simple yet effective self-supervised multi-modal feature adaptation framework for multi-modal anomaly detection. Specifically, LSFA performs feature adaptation in both intra-modal and inter-modal aspects. For the former, a dynamic-updated memory-bank based feature compactness optimization scheme is proposed to enhance the feature sensitivity to unusual patterns. For the latter, a local-to-global consistency alignment strategy is proposed for multi-scale inter-modality information interaction. Extensive experiments show that LSFA achieves much superior performance than previous methods and prominently boosts existing feature embedding based baselines.

Acknowledgements

This work was supported by National Natural Science Fund of China (62076184, 61976158, 61976160, 62076182, 62276190), in part by Fundamental Research Funds for the Central Universities and State Key Laboratory of Integrated Services Networks (Xidian University), in part by Shanghai Innovation Action Project of Science and Technology (20511100700) and Shanghai Natural Science Foundation (22ZR1466700).

References

1. Deep learning for unsupervised anomaly localization in industrial images: A survey. TIM (2022)
2. Multimodal industrial anomaly detection via hybrid fusion. In: CVPR (2023)
3. Real3d-ad: A dataset of point cloud anomaly detection. arXiv (2023)
4. Deep industrial image anomaly detection: A survey. MIR (2024)
5. Ahuja, N.A., Ndiour, I., Kalyanpur, T., Tickoo, O.: Probabilistic modeling of deep features for out-of-distribution and adversarial detection. arXiv preprint arXiv:1909.11786 (2019)
6. Akcay, S., Ameln, D., Vaidya, A., Lakshmanan, B., Ahuja, N., Genc, U.: Anomalib: A deep learning library for anomaly detection. In: 2022 IEEE International Conference on Image Processing (ICIP). pp. 1706–1710. IEEE (2022)
7. Akcay, S., Atapour-Abarghouei, A., Breckon, T.P.: Ganomaly: Semi-supervised anomaly detection via adversarial training. In: ACCV (2019)
8. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In: CVPR (2019)
9. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In: CVPR (June 2020)
10. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In: CVPR (2020)
11. Bergmann, P., Jin, X., Sattlegger, D., Steger, C.: The mvtec 3d-ad dataset for unsupervised 3d anomaly detection and localization. In: VISIGRAPP (2022)
12. Bergmann, P., Sattlegger, D.: Anomaly detection in 3d point clouds using deep geometric descriptors. arXiv preprint arXiv:2202.11660 (2022)
13. Bonfiglioli, L., Toschi, M., Silvestri, D., Fioraio, N., De Gregorio, D.: The eyecandies dataset for unsupervised multimodal anomaly detection and localization. In: ACCV (2022)
14. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: ICCV. pp. 9650–9660 (2021)
15. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015)
16. Chen, R., Xie, G., Liu, J., Wang, J., Luo, Z., Wang, J., Zheng, F.: Easynet: An easy network for 3d industrial anomaly detection. In: Proceedings of the 31st ACM International Conference on Multimedia (2023)

17. Defard, T., Setkov, A., Loesch, A., Audigier, R.: Padim: a patch distribution modeling framework for anomaly detection and localization. In: ICPR (2021)
18. Deng, H., Li, X.: Anomaly detection via reverse distillation from one-class embedding. In: CVPR (2022)
19. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
20. Gong, D., Liu, L., Le, V., Saha, B., Mansour, M.R., Venkatesh, S., Hengel, A.v.d.: Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In: ICCV (2019)
21. Gudovskiy, D., Ishizaka, S., Kozuka, K.: Cflow-ad: Real-time unsupervised anomaly detection with localization via conditional normalizing flows. In: WACV (2022)
22. Gudovskiy, D., Ishizaka, S., Kozuka, K.: Cflow-ad: Real-time unsupervised anomaly detection with localization via conditional normalizing flows. In: WACV (2022)
23. Horwitz, E., Hoshen, Y.: An empirical investigation of 3d anomaly detection and segmentation. arXiv preprint arXiv:2203.05550 (2022)
24. Hou, J., Zhang, Y., Zhong, Q., Xie, D., Pu, S., Zhou, H.: Divide-and-assemble: Learning block-wise memory for unsupervised anomaly detection. In: ICCV (2021)
25. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. In: ICLR (2022)
26. Lee, S., Lee, S., Song, B.C.: Cfa: Coupled-hypersphere-based feature adaptation for target-oriented anomaly localization. IEEE Access **10**, 78446–78454 (2022)
27. Li, C.L., Sohn, K., Yoon, J., Pfister, T.: Cutpaste: Self-supervised learning for anomaly detection and localization. In: CVPR (2021)
28. Liu, J., Xie, G., Wang, J., Li, S., Wang, C., Zheng, F., Jin, Y.: Deep industrial image anomaly detection: A survey. arXiv e-prints pp. arXiv–2301 (2023)
29. Pang, Y., Wang, W., Tay, F.E.H., Liu, W., Tian, Y., Yuan, L.: Masked autoencoders for point cloud self-supervised learning (2022)
30. Pang, Y., Wang, W., Tay, F.E., Liu, W., Tian, Y., Yuan, L.: Masked autoencoders for point cloud self-supervised learning. In: ECCV (2022)
31. Ristea, N.C., Madan, N., Ionescu, R.T., Nasrollahi, K., Khan, F.S., Moeslund, T.B., Shah, M.: Self-supervised predictive convolutional attentive block for anomaly detection. In: CVPR (2022)
32. Roth, K., Pemula, L., Zepeda, J., Schölkopf, B., Brox, T., Gehler, P.: Towards total recall in industrial anomaly detection. In: CVPR (2022)
33. Rudolph, M., Wandt, B., Rosenhahn, B.: Same same but differnet: Semi-supervised defect detection with normalizing flows. In: WACV (2021)
34. Rudolph, M., Wehrbein, T., Rosenhahn, B., Wandt, B.: Asymmetric student-teacher networks for industrial anomaly detection. arXiv preprint arXiv:2210.07829 (2022)
35. Salehi, M., Sadjadi, N., Baselizadeh, S., Rohban, M.H., Rabiee, H.R.: Multiresolution knowledge distillation for anomaly detection. In: CVPR (2021)
36. Wang, G., Han, S., Ding, E., Huang, D.: Student-teacher feature pyramid matching for anomaly detection. In: BMVC (2021)
37. Wu, J.C., Chen, D.J., Fuh, C.S., Liu, T.L.: Learning unsupervised metaformer for anomaly detection. In: ICCV (2021)
38. Wu, K., Zhu, L., Shi, W., Wang, W., Wu, J.: Self-attention memory-augmented wavelet-cnn for anomaly detection. IEEE Transactions on Circuits and Systems for Video Technology (2022)

39. Yan, X., Zhang, H., Xu, X., Hu, X., Heng, P.A.: Learning semantic context from normal samples for unsupervised anomaly detection. In: AAAI (2021)
40. Yu, J., Zheng, Y., Wang, X., Li, W., Wu, Y., Zhao, R., Wu, L.: Fastflow: Unsupervised anomaly detection and localization via 2d normalizing flows. arXiv preprint arXiv:2111.07677 (2021)
41. Zavrtanik, V., Kristan, M., Skočaj, D.: Cheating depth: Enhancing 3d surface anomaly detection via depth simulation. In: CVPR (2024)
42. Zhang, Q., Chen, M., Bukharin, A., He, P., Cheng, Y., Chen, W., Zhao, T.: Adaptive budget allocation for parameter-efficient fine-tuning. CoRR **abs/2303.10512** (2023)