Shape-Guided Dual-Memory Learning for 3D Anomaly Detection

Yu-Min Chu [*1] **Chieh Liu** [*1] **Ting-I Hsieh** [1] **Hwann-Tzong Chen** [1 2] **Tyng-Luh Liu** [3]

Abstract

We present a shape-guided expert-learning framework to tackle the problem of unsupervised 3D anomaly detection. Our method is established on the effectiveness of two specialized expert models and their synergy to localize anomalous regions from color and shape modalities. The first expert utilizes geometric information to probe 3D structural anomalies by modeling the implicit distance fields around local shapes. The second expert considers the 2D RGB features associated with the first expert to identify color appearance irregularities on the local shapes. We use the two experts to build the dual memory banks from the anomaly-free training samples and perform shape-guided inference to pinpoint the defects in the testing samples. Owing to the per-point 3D representation and the effective fusion scheme of complementary modalities, our method efficiently achieves state-of-the-art performance on the MVTec 3D-AD dataset with better recall and lower false positive rates, as preferred in real applications.

1. Introduction

Unsupervised anomaly detection and localization have many applications in manufacturing and health care. Previous methods mainly use color information to identify defects and abnormal regions in the input images. While the color information is generally sufficient for localizing anomalies in most cases, it has also been shown that the 3D geometric information, when adequately used, can be beneficial for achieving better performance (Horwitz & Hoshen, 2022). Our work aims to solve the problem of 3D anomaly detection and localization on the recently published MVTec 3D-AD dataset. We propose shape-guided dual-memory learning to combine the color and geometric information for higher anomaly localization accuracy with lower computation and memory costs. Figure 1 illustrates the complementary advantage of our method for precisely localizing the defects from different modalities.

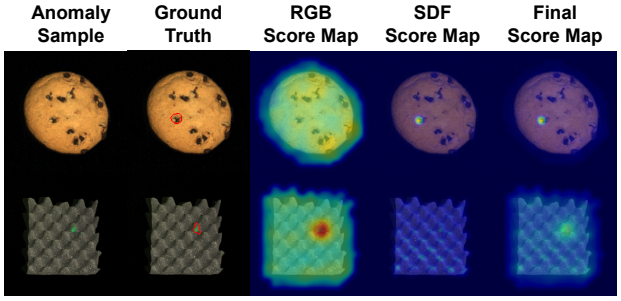

Figure 1. Results of our method on the MVTec 3D-AD dataset. Defects that are only perceptible in one but not the other modality can be successfully localized by our method.

The performance of anomaly detection is often evaluated by the per-region overlap (PRO) (Bergmann et al., 2021) and the corresponding false positive rate for consecutively increasing anomaly thresholds. The most common setting is to report the area under the PRO curve (AU-PRO) integrated up to a false positive rate of 30% (*i.e.*, integration limit at 0.3). However, in real applications, a false positive rate of 30% might be too large and thus imprecise to pinpoint the defect. To address this issue, we design our method to pursue higher AU-PRO at very small integration limits. Our method uses neural implicit functions (NIFs) to represent local shapes by signed distance fields, as done by current methods on 3D reconstruction (Jiang et al., 2020; Takikawa et al., 2021; Ma et al., 2021; 2022; Li et al., 2022). Partitioning a point-cloud sample into NIF-represented local patches allows us to model 3D objects of complex shapes under orientation changes. The local signed distance fields also enable fine-grained per-point anomaly prediction. As a result, our method achieves the state-of-the-art AU-PRO on the MVTec 3D-AD benchmark, even at very small integration limits, which is considered rather challenging for previous 2D and 3D anomaly detection methods.

We summarize the contributions of this work as follows:

1. The proposed shape-guided approach effectively in-

[*]Equal contribution [1]Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan [2]Aeolus Robotics, Taipei, Taiwan [3]Institute of Information Science, Academia Sinica, Taipei, Taiwan. Correspondence to: Tyng-Luh Liu <liutyng@iis.sinica.edu.tw>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

tegrates the complementary modalities of color and geometry. Our method requires less memory usage and facilitates faster inference.

2. We present the first work that uses neural implicit functions of signed distance fields to represent local shapes for 3D anomaly detection. Advantageously, we can model 3D point clouds of complex structures to the per-point fine-grained level.
3. Our method achieves state-of-the-art performance on the MVTec 3D-AD dataset, especially at small integration limits, which means better recall and lower false positive rates as preferred in real applications.

2. Related Work

2.1. 2D Anomaly Detection

Many methods have been presented to solve unsupervised 2D anomaly detection and localization. Most of the 2D-based methods are evaluated on the MVTec AD dataset (Bergmann et al., 2019; 2021), while a recent benchmark by (Zheng et al., 2022) also shows that many unsupervised 2D-based methods can also perform well on the MVTec 3D-AD dataset (Bergmann et al., 2022) even only using RGB information. In the following, we briefly review several 2D-based methods.

Feature embedding-based methods. Methods like (Defard et al., 2020; Lee et al., 2022; Roth et al., 2022) use a pretrained model to extract the normal features during the training phase. During the testing phase, the testing features are compared with the individual training features or their distributions using a distance metric. If a testing feature differs from the training features, the region of the testing feature is more likely to belong to an anomaly region. The feature embedding-based methods are straightforward, but there might be a higher computation cost in finding the corresponding normal features in the training data.

Normalizing flows. CFLOW-AD uses the conditional normalizing flow with a positional encoder to model a distribution of normal patches (Gudovskiy et al., 2022); it aims to separate the in-distribution testing patches and out-of-distribution ones based on the probability density function. In CSflow (Shi et al., 2022), feature maps of different scales are processed by a fully convolutional normalizing flow, transforming the original distribution of the input data to an interpretable latent space, which improves the accuracy of detecting anomalies.

Student-teacher networks and simulation-based approaches. In the Uninformed Students method (Bergmann et al., 2020), a teacher network is pretrained on a large dataset of nature data to learn the discriminative embeddings, and then the pretrained teacher network is used to train the student networks with non-defect data. The goal is to make the output from the student resemble the output from the teacher. For inference, the regression errors between two discriminative embeddings from both networks are treated as the anomaly scores of the input data. Not particular for 2D anomaly detection, AST (Rudolph et al., 2023) uses both color and depth information for anomaly detection and aims to increase the distances between the student and teacher outputs of the abnormal patches, which means the anomalies can be easier separated from the normal regions. To solve the issue of lacking abnormal samples in unsupervised anomaly detection, simulation-based methods like (Li et al., 2021; Schlüter et al., 2022; Yang et al., 2022) artificially add noise onto the normal data to simulate defects, and the models are trained on the simulated abnormal samples. Our method follows the standard setting of unsupervised anomaly detection. We do not rely on simulated abnormal samples while achieving state-of-the-art results on MVTec 3D-AD.

2.2. 3D Anomaly Detection

Due to the lack of more comprehensive 3D datasets, not much previous work has focused on unsupervised 3D anomaly detection, except for a few methods tackling the problem on 3D brain scans (Behrendt et al., 2022; Bengs et al., 2022; Viana et al., 2020). Bergmann et al. introduce the *MVTec 3D-AD* dataset (Bergmann et al., 2022) for benchmarking unsupervised 3D anomaly detection methods. The dataset contains high-resolution color point clouds of manufactured products. The training and validation sets consist of only *anomaly-free* samples as in real-world inspection scenarios. An unsupervised method trained from these anomaly-free samples must detect unknown types of defects shown in the test samples of the corresponding object categories. Unlike the previous color-image-based datasets for anomaly detection (Bergmann et al., 2019; 2021), the point-cloud representation of the MVTec 3D-AD dataset provides helpful geometric cues for detecting defects that are not easy to identify in color images. Bergmann and Sattlegger (Bergmann & Sattlegger, 2022) propose a student-teacher framework that learns adaptive geometric features for unsupervised 3D anomaly detection, where the teacher network is trained in a self-supervised manner to encode local geometric descriptions from local patches. They evaluate their method on the MVTec 3D-AD dataset and show that their proposed 3D Student-Teacher can reliably localize geometric anomalies in test point clouds. In this work, we also use the MVTec 3D-AD dataset to evaluate the proposed method and achieve state-of-the-art results.

Horwitz and Hoshen use the MVTec 3D-AD dataset to analyze the usefulness of 3D information for anomaly detection (Horwitz & Hoshen, 2022). They conclude that "*3D in-*

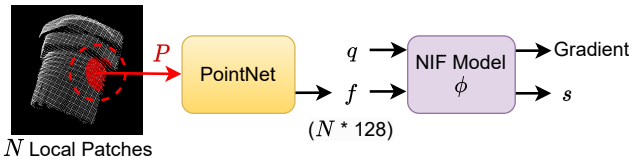

Figure 2. The shape expert model. Motivated by Neural-Pull (Ma et al., 2021), we consider PointNet and NIF to learn local representation of surface geometry. For each point $\mathbf{q}$ near the local surface of a 3D patch P, the shape model is trained to predict its signed distance $s = \phi(\mathbf{q}; \mathbf{f})$, where $\mathbf{f}$ is the feature vector by PointNet.

formation is often required to identify anomalies, even when color is available". Their study also shows that rotation-invariant 3D representations that model local fine-grained structures are critical for 3D anomaly detection. They further propose an approach called BTF (Back to the Feature), combining the complementary attributes from color and geometric modalities to achieve better results on the MVTec 3D-AD dataset.

We observe similar issues and properties of 3D representations for anomaly detection. Therefore, we extract rotation-invariant features from the point cloud and adopt an implicit representation that can model find-grained 3D local structures through signed distance functions. Furthermore, we present a shape-guided mechanism that effectively integrates the color and geometric modalities to achieve state-of-the-art performance in 3D anomaly detection.

3. Method

Different from the 2D setting, training data, *e.g.*, MVTec 3D-AD, for 3D anomaly detection are connectedly presented in two different modalities, including pixelwise RGB values and pointwise 3D coordinates. To fully exploit the complementary effect of the two representation forms, we design a shape-guided appearance reconstruction scheme that efficiently connects the two information streams to enhance the accuracy of predicting and localizing anomalies.

3.1. Shape-Guided Expert Learning

The proposed method is established based on the effectiveness of two specialized expert models and their synergy to better address the task of 3D anomaly detection. The first expert utilizes 3D information to probe possible anomalies in shape geometry, and the second expert considers the RGB information to single out any appearance irregularities (in the aspect of color). In what follows, we describe how these two expert models are developed and correlated.

Shape expert. With the availability of pointwise coordinates, we consider designing a 3D shape expert for anomaly detection by focusing on learning *local* geometry representation. Our motivation to aim for local representation is twofold. First, defects or anomalous parts often occur locally rather than globally. Second, the formulation for learning the local representation of point clouds tends to be more scalable and efficient.

As shown in Figure 2, we leverage two existing models, namely PointNet (Qi et al., 2017) and Neural Implicit Function (NIF) (Ma et al., 2022), for point-cloud applications to explore the 3D shape information. Specifically, we first divide a complete point cloud into 3D patches and carry out local representation learning. For each resulting patch, we sample, say, 500 points and apply PointNet to obtain its feature vector, denoted as $\mathbf{f}$, which encodes the corresponding local geometry. Now let the NIF model be ϕ. To train ϕ for anomaly detection, we follow the technique in (Ma et al., 2021) to sample a set of query points, $Q = \{\mathbf{q}\}$, near the surface of the underlying 3D patch, and pass these queries along with the PointNet feature $\mathbf{f}$ to the NIF model to predict their signed distances, $\{s\}$. We express the process of predicting the signed distance s of a query point $\mathbf{q} \in Q$ with respect to the local surface by

$$\mathbf{q} \in Q \xmapsto{\phi,\, \mathbf{f}} s = \phi(\mathbf{q}; \mathbf{f}), \tag{1}$$

where besides the input $\mathbf{q}$, the predicted outcome s is conditioned on the patchwise feature vector $\mathbf{f}$ by PointNet.

Each pair $\{\phi, \mathbf{f}\}$ in (1) constitutes a signed distance function (SDF) and can be used to measure the local surface geometry of a point cloud. Since the NIF ϕ is universal to all patches and category-agnostic, upon the completion of learning the shape expert, we only need to store all the patchwise feature vectors $\{\mathbf{f}\}$ into the SDF memory bank, denoted as M_S, to implicitly encode all "normal" local representations.

Appearance expert. The goal of constructing the appearance expert is to create a *shape-guided* memory bank M_A that can be used to reconstruct "normal" RGB features.

We consider the paired relationship of a point cloud and its 2D RGB image, as illustrated in Figure 3. Having learned the shape expert, we can examine the mapping between an SDF and its corresponding RGB features. For each SDF, we trace back its 500 sampled points (*i.e.*, the input to PointNet) in the 3D receptive field and then calculate their 2D coordinates to retrieve the corresponding RGB features. To enhance its representation capacity in color appearance, the 2D correspondences are uniformly expanded by two pixels on the feature map to include more RGB features. (See Figure 4.) In our implementation, each SDF would correspond to around 40 to 60 RGB feature vectors. As such, we can obtain the shape-guided memory bank M_A, which comprises SDF-specific RGB dictionaries of the same number as the SDFs in M_S.

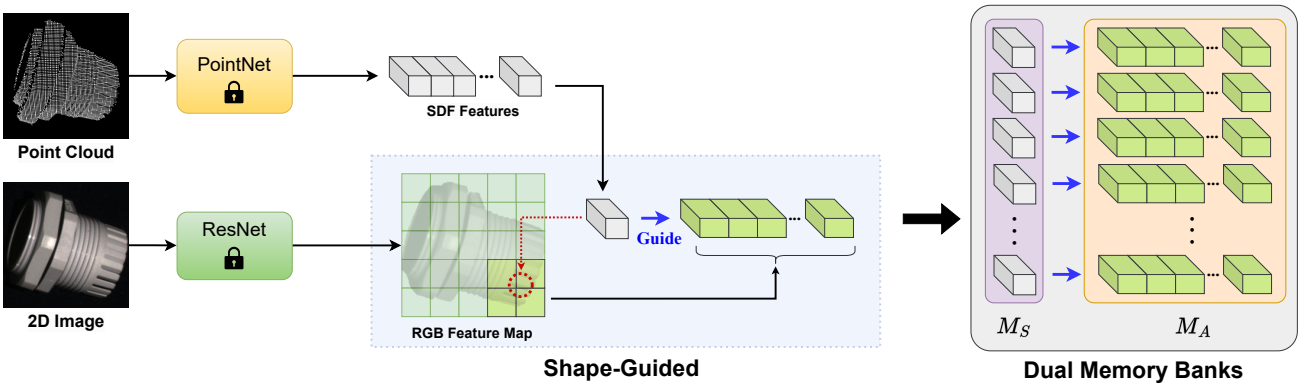

Figure 3. Dual memory banks. Since NIF is universal to all 3D patches, we only need to store their respective feature vector $\mathbf{f}$ into the SDF memory bank, denoted as M_S. On the other hand, each patch corresponds to a region in the RGB feature map and results in an SDF-specific dictionary. All such shape-guided RGB dictionaries are saved to form the RGB memory bank M_A.

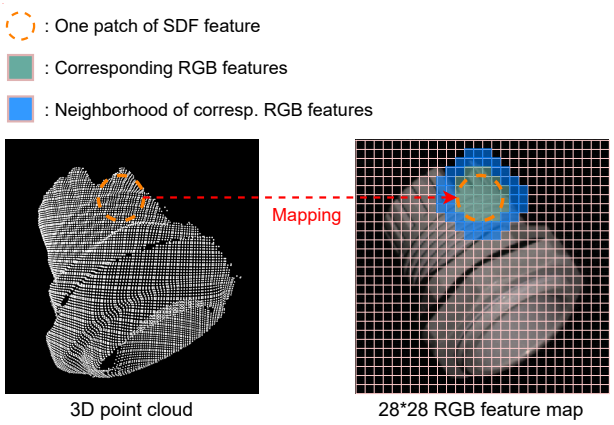

Figure 4. We project each 3D patch onto the image space and obtain its RGB features (in green). To account for the boundaries of possible defects of holes or cracks, the original mapping is extended by a 2-pixel neighborhood (in blue) to accommodate more RGB features. The tactic to expand RGB features can improve the detection of abnormalities in empty regions due to defects.

3.2. Shape-Guided Inference

With the dual memory banks M_S and M_A, we are ready to perform inference to detect whether a testing sample $\mathbf{x}$ includes anomalies/defects. (See Figure 5.) The steps are listed as follows.

1. Use PointNet to get all patch-level SDFs, $\{\tilde{\mathbf{f}}\}$ of $\mathbf{x}$.
2. Use ResNet to get the RGB feature map of $\mathbf{x}$. Those pixels that are associated with at least one SDF are considered foreground in the 2D RGB image.
3. For each SDF in $\{\tilde{\mathbf{f}}\}$, find its $k_1 = 10$ nearest neighbors in M_S to form the respective dictionary and obtain its approximation $\hat{\mathbf{f}}$ via sparse representation.
4. For each patch of $\mathbf{x}$, use the patchwise reconstructed $\hat{\mathbf{f}}$ to compute the signed distances, $s = \phi(\hat{\mathbf{q}}; \hat{\mathbf{f}})$, for all the 3D points, $\{\hat{\mathbf{q}}\}$, from its receptive field.
5. Adopt the absolute values of signed distances from all the patches of $\mathbf{x}$ to form the final SDF score map.
6. For all the relevant SDFs in M_S that are used in computing sparse representations of step 3, take the union of all their associated RGB dictionaries in M_A and form a shape-guided RGB dictionary, denoted as $\hat{D}$.
7. For each foreground RGB feature vector from step 2, find its $k_2 = 5$ nearest neighbors from $\hat{D}$ and obtain its sparse representation. The ℓ_2 distances yielded by the approximations form the final RGB score map.
8. Perform score-map alignment (to be described next) and pixelwise take the maximum of the SDF and RGB responses as the corresponding anomaly score.

Score-map alignment. Fusing the SDF and RGB score maps by max pooling requires values of the two to be in a comparable range. Since anomalous samples are not available in training for estimating proper statistics, we overcome this difficulty by simulating inference for 25 randomly selected training samples and adopting a "leave-oneself-out" strategy to mimic the testing outcomes. This would exclude the SDF and RGB features of a query itself from the nearest-neighbor searches in the testing steps. To align the two resulting score distributions, we consider the mapping $y \mapsto a \times y + b$ such that $\text{mean} \pm 3 \times \text{standard deviation}$ of the RGB score distribution would map to their SDF counterpart. The resulting scaling and shifting parameters a and b can be readily used in reference to rectify an RGB score y into $a \times y + b$.

Finally, we remark that so far our formulation is described to address 3D anomaly detection only for a single category.

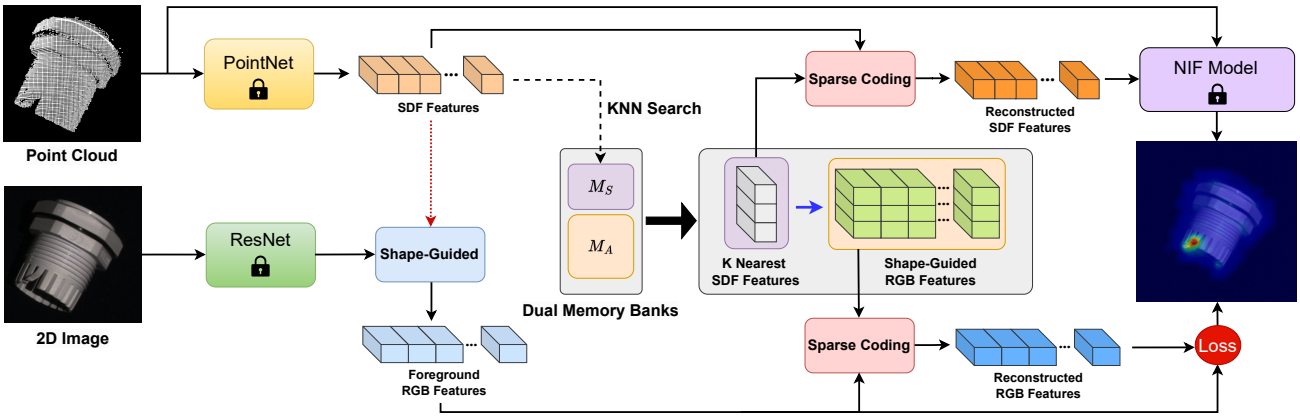

Figure 5. Shape-guided inference. The dual memory banks are connected by an SDF-to-RGB relationship that enables each SDF in M_S to locate its corresponding shape-guided RGB dictionary in M_A. The design facilitates the use of sparse representation in inference.

Nevertheless, considering that the ten object categories of MVTec 3D-AD are *unmistakably distinct* and our implementation of a classifier indeed achieves 100% classification accuracy, our method essentially provides a unified approach to tackling anomaly detection on MVTec 3D-AD.

4. Experiments

4.1. Experimental Setup

Dataset. We evaluate our method on MVTec 3D-AD (Bergmann & Sattlegger, 2022), which provides ten different categories of 3D objects for 2D+3D anomaly detection. MVTec 3D-AD contains 2,656 training, 294 validation, and 1,197 test samples. The training and validation data do not have any defects, while the test data are split into 249 normal samples and 948 anomalous samples. The anomalous test samples include about 4 to 5 different types of defects in each category. MVTec 3D-AD differs from the preceding 2D-AD dataset: Each sample is provided with a high-resolution point cloud and the corresponding RGB image. Our proposed method aims to fully utilize the important 2D and 3D modalities for better performance in defect detection. We divide each point-cloud sample of the training data into patches to enrich the shape diversity and use the patches to train the PointNet and the NIF model. The same training samples with the associated RGB modality are then used to build the dual memory banks.

Preprocessing. The preprocessing of point clouds comprises several steps. First, we follow the baseline method BTF (Horwitz & Hoshen, 2022) to remove the point cloud of the background in the whole dataset. Next, we prepare the npz files of the cropped patches from the training and test samples in advance according to the procedure of extracting local patches described in the next paragraph. The npz files for training and testing contain 3D points and their corresponding 2D indexes, but the npz files for training PointNet and NIF consist of both the spatially sub-sampled points and the original points. Furthermore, we resize the original point clouds and images from a resolution of 800×800 to 224×224 using nearest and bicubic interpolation, respectively, like the baseline (Horwitz & Hoshen, 2022).

Local patches. The previous methods PCP (Ma et al., 2022) and LIG (Jiang et al., 2020) perform well by dividing the point cloud into several local regions. Recent progress in learning implicit neural features also facilitates analysis and modeling of the local structure of the point cloud. Our method also divides the entire point cloud into local patches to model the local structures of the point cloud. Inspired by Point-MAE (Pang et al., 2022), we use Farthest Point Sampling (FPS) to sample a set of points from the original point could, and then we find the K-nearest neighbors within the receptive field centered at each FPS point to form a local patch. Note that each point in the original point cloud may be considered a K-nearest neighbor of multiple FPS points, *i.e.*, the local patches may overlap with each other to share some neighborhoods. Due to the overlapping of patches, we dynamically adjust the size of the sampled set (*i.e.*, the number of FPS points) to ensure that the union of the local patches covers as much as possible the original point cloud.

Parameter settings. We divide each point-cloud sample into overlapped 3D local patches. Each local patch contains 500 points ($K = 500$). We ensure that the number of local patches is large enough to cover all points in a point-cloud sample jointly. For example, we choose an overlapping ratio of 10 such that we may obtain roughly 150 local patches for a point-cloud sample consisting of about 7,500 points (derived from 'overlapping ratio' times 'total number of points' divided by 'size of a local patch', *i.e.*, $10 \times 7500/500 = 150$). In the preprocessing step for

Table 1. Anomaly detection performance evaluated by the metric of Img-AUROC on the MVTec 3D-AD dataset (Bergmann et al., 2022). The best results are marked in red, and the second-best results are in blue.

	Method	Bagel	Cable Gland	Carrot	Cookie	Dowel	Foam	Peach	Potato	Rope	Tire	**Mean**
RGB only	PaDim	0.975	0.775	0.698	0.582	0.959	0.663	0.858	0.535	0.832	0.760	0.764
	CSflow	0.894	0.917	0.749	0.668	0.938	0.897	0.603	0.419	0.971	0.726	0.778
	BTF (RGB)	0.854	0.840	0.824	0.687	0.974	0.716	0.713	0.593	0.920	0.724	0.785
	CFlow	0.880	0.858	0.828	0.563	0.986	0.738	0.757	0.628	0.970	0.720	0.793
	PatchCore	0.912	0.902	0.885	0.709	0.952	0.733	0.727	0.562	0.962	0.768	0.811
	AST (RGB)	0.947	0.928	0.851	0.825	0.981	0.951	0.895	0.613	0.992	0.821	0.880
	Ours (RGB)	0.911	0.936	0.883	0.662	0.974	0.772	0.785	0.641	0.884	0.706	0.815
3D only	BTF (SIFT)	0.696	0.553	0.824	0.696	0.795	0.773	0.573	0.746	0.936	0.553	0.714
	BTF (FPFH)	0.820	0.533	0.877	0.769	0.718	0.574	0.774	0.895	0.990	0.582	0.753
	AST (Depth)	0.881	0.576	0.965	0.957	0.679	0.797	0.990	0.915	0.956	0.611	0.833
	Ours (SDF)	0.983	0.682	0.978	0.998	0.960	0.737	0.993	0.979	0.966	0.871	0.916
3D+RGB	BTF (RGB+FPFH)	0.938	0.765	0.972	0.888	0.960	0.664	0.904	0.929	0.982	0.726	0.873
	AST (RGB + Depth)	0.983	0.873	0.976	0.971	0.932	0.885	0.974	0.981	1.000	0.797	**0.937**
	Ours (Shape-guided)	0.986	0.894	0.983	0.991	0.976	0.857	0.990	0.965	0.960	0.869	**0.947**

training PointNet and NIF, we sample just 20 query points around each real point as done by PCP (Ma et al., 2022). We set the learning rate and batch size to 0.0001 and 32, respectively, which empirically achieves efficient convergence.

4.2. Implementation Detail

Training the experts. We use the training samples to train a simplified PointNet for extracting 3D features from local patches and fine-tune an ImageNet pretrained ResNet for extracting the 28×28 RGB feature maps. We also train a Neural Implicit Function (NIF) to derive the sign distance function (SDF) from the 3D features extracted by PointNet, as shown in Figure 2. The simplified PointNet consists of three convolution layers and two fully connected layers, each including batch normalization. The NIF model is a multilayer perceptron to characterize the latent shape of the local geometry. Our 2D model for the RGB cue comprises a Wide ResNet-50-2 (Zagoruyko & Komodakis, 2016) as in PatchCore (Roth et al., 2022), and we extract and combine the features from the first and second layers.

Score alignment. The scale of the RGB features is very different from that of the SDF features, which yields different score distributions. We need to calibrate the two distributions before fusing the scores. We randomly choose 25 training samples to simulate the score distribution before pixel-level testing. To calibrate the distributions of scores, we align the mean $\pm 3\times$ standard deviation of RGB scores with the mean $\pm 3\times$ standard deviation of SDF scores by applying an affine transformation to the RGB scores. When testing, we use the previously calculated weight and bias of the affine transformation to align the RGB scores to the SDF scores. Finally, we can fuse the two score maps directly by taking the per-pixel maximum.

4.3. Evaluation Metrics

We adopt the Area Under the Receiver Operator Curve (AUROC) to evaluate the performance of the proposed method on both image-level (Img-AUROC) and pixel-level (Pix-AUROC). To evaluate the prediction more precisely for each pixel in the MVTec 3D-AD data, we use per-region overlap (PRO) (Bergmann et al., 2021) and compute the Area Under PRO curve (AUPRO) as an evaluation metric for anomaly localization using the produced anomaly scores and the ground-truth connected components.

4.4. Experimental Results

Table 1 compares our method and existing methods on the MVTec 3D-AD dataset, evaluated with the Img-AUROC metric. We compare our method with PaDim (Defard et al., 2020), CSflow(Shi et al., 2022), BTF (Horwitz & Hoshen, 2022), CFlow (Gudovskiy et al., 2022), PatchCore (Roth et al., 2022), AST (Rudolph et al., 2023), and 3D-ST (Bergmann & Sattlegger, 2022). Table 2 shows anomaly localization performance with the AUPRO metric, where we compute the integration of the PRO values over the false-positive rates (FPRs). Like most previous methods, we use 0.3 as the upper limit of the FPR integration limit. A smaller FPR integration limit means we assume a lower toleration of false positives. Since it is more critical for the anomaly localization performance at low integration limits in real-world scenarios, we compare our method with existing methods at seven different integration limits $\{0.3, 0.2, 0.1, 0.07, 0.05, 0.03, 0.01\}$ in Figure 6. The results show that our method achieves state-of-the-art performance at the standard integration limit of 0.3 and very low integration limits.

Table 2. Anomaly localization performance measured by the metric of AUPRO on the MVTec 3D-AD dataset. The best results are marked in red, and the second-best results are in blue.

	Method	Bagel	Cable Gland	Carrot	Cookie	Dowel	Foam	Peach	Potato	Rope	Tire	**Mean**
RGB only	CFlow	0.855	0.919	0.958	0.867	0.969	0.500	0.889	0.935	0.904	0.919	0.871
	BTF (RGB)	0.898	0.948	0.927	0.872	0.927	0.555	0.902	0.931	0.903	0.899	0.876
	PatchCore	0.899	0.953	0.957	0.918	0.930	0.719	0.920	0.937	0.938	0.929	0.910
	PaDim	0.980	0.944	0.945	0.925	0.961	0.792	0.966	0.940	0.937	0.912	0.930
	Ours (RGB)	0.946	0.972	0.96	0.914	0.958	0.776	0.937	0.949	0.956	0.957	0.933
3D only	3D-ST	0.950	0.483	0.986	0.921	0.905	0.632	0.945	0.988	0.976	0.542	0.833
	BTF (SIFT)	0.894	0.722	0.963	0.871	0.926	0.613	0.870	0.973	0.958	0.873	0.866
	BTF (FPFH)	0.972	0.849	0.981	0.939	0.963	0.693	0.975	0.981	0.980	0.949	0.928
	Ours (SDF)	0.974	0.871	0.981	0.924	0.898	0.773	0.978	0.983	0.955	0.969	0.931
3D+RGB	BTF (RGB+FPFH)	0.976	0.967	0.979	0.974	0.971	0.884	0.976	0.981	0.959	0.971	**0.964**
	Ours (Shape-guided)	0.981	0.973	0.982	0.971	0.962	0.978	0.981	0.983	0.974	0.975	**0.976**

Table 3. Comparison of inference time (second) per sample, frame per second (FPS), the number of features (NoF), and the percentage of RGB memory usage on an Nvidia GTX 1080 GPU.

Method	Inference Time ↓	FPS ↑	NoF ↓	Memory Usage ↓	Img-ROC ↑	Pix-ROC ↑	AUPRO (0.3) ↑	AUPRO (0.01) ↑
BTF	2.19	0.46	**20,823**	**10%**	0.873	0.993	0.964	0.394
w/o Shape-guided	3.60	0.29	208,230	100%	**0.947**	**0.996**	**0.976**	0.453
Shape-guided	**2.05**	**0.69**	26,452	13.5%	**0.947**	**0.996**	**0.976**	**0.456**

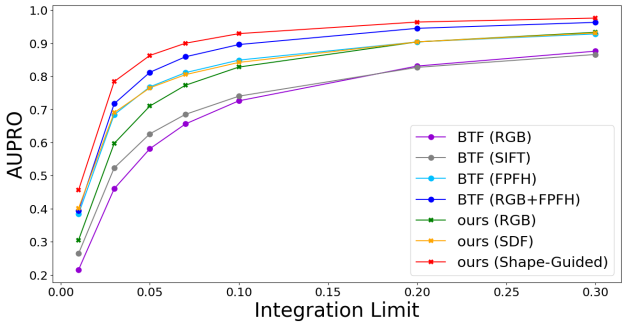

Figure 6. Anomaly localization performance (AUPRO) of ours and previous methods for varying integration limits.

4.5. Computational Complexity

Inference time and memory usage. Inference performance and memory usage are important in industrial applications. Our method with the shape-guided mechanism requires only low memory usage to achieve state-of-the-art results of both the pixel-level and image-level predictions, as shown in Table 3. We show the average inference time per sample and the average RGB memory usage of our method and BTF. BTF uses the PatchCore method to subsample 10% features for the coreset, where 10% is the most common setting in PatchCore. Note that the memory occupied by the RGB features is much larger than that occupied by SDF features, and the inference time for RGB query is also much greater than SDF query, so we only compare the RGB-related computations. We also include the inference speed (measured in frames per second) and the number of features involved in the NN search when detecting anomalies on a single test sample. Our shape-guided mechanism contributes to achieving the best AUPRO (0.3) and AUPRO (0.01) at the fastest speed of 0.69 fps.

Detailed analysis. The computational complexity of anomaly detectors can be assessed by examining the computations executed on both GPU and CPU. GPUs primarily handle feature extraction using models like ResNet or PointNet. For instance, the ResNet model we employ for RGB features necessitates 5.0 GMACs with 4.13M parameters, while our SDF model (PointNet + NIF) for point cloud demands 1.29 GMACs with 3.17M parameters. Conversely, BTF, following PatchCore, only employs the ResNet model for RGB features, which requires 5.0 GMACs with 4.13M parameters. Overall, our method requires about 25% more GMACs than BTF. Nonetheless, we found that subsequent operations, such as kNN feature search carried out by CPUs, predominantly influence the total computational costs of anomaly detection algorithms. The FPFH representation in BTF also needs additional CPU computation. Hence, a more comprehensive comparison of AD algorithms' computational costs would involve evaluating their overall inference time and memory usage in conjunction with GPU GMACs/FLOPs.

Table 4. Comparison of different patch size K.

K	Img-ROC ↑	AUPRO (0.3) ↑	AUPRO (0.01) ↑	Inference Time ↓
1000	0.894	0.965	0.413	**1.72**
750	0.929	0.973	0.439	1.88
500	0.947	**0.976**	**0.456**	2.05
250	**0.966**	0.975	0.451	3.61

4.6. Patch Size Analysis

We evaluate our method for different patch sizes K. The results are listed in the Table 4. Increasing the value of K corresponds to dividing the point cloud into overlapped 3D patches of a larger size. Observe that a large patch size has the advantage of encompassing a complete region of anomalous occurrence, but meanwhile may confuse the SDF to encode the local 3D surface geometry properly. We choose to set $K = 500$ for the sake of inference efficiency and accuracy.

4.7. Qualitative Results

Comparison with other methods. Figure 7 demonstrates that our method outperforms others in precisely localizing anomalous regions. Our method performs well even with a low integration limit.

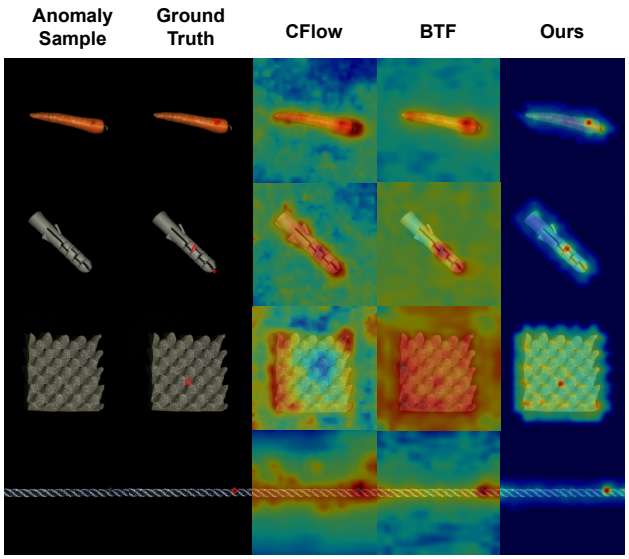

Figure 7. Visualization of ours and other methods.

Benefits of cross-modality. We provide more qualitative results in Figure 8. We highlight the complementarity of the appearance and shape experts in our approach. A green checkmark (✓) on an expert means the expert is responsive to the anomalous region, while a red cross (X) means the expert is inactive. Our fusion scheme benefits from such complementarity and can successfully identify the anomalous region on the final score map.

Failure cases. Our method combines the advantages of RGB and 3D point cloud to make them complementary in anomaly detection. Therefore, if neither can effectively detect anomalies, the performance will fall below expectations. In Figure 9, most of our failure cases occur on elusive anomalies that are difficult to detect, and existing methods also underperform on these types of data.

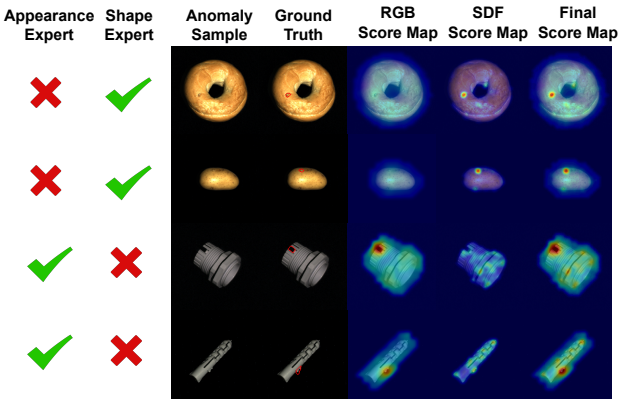

Figure 8. Complementary characteristics of cross-modality.

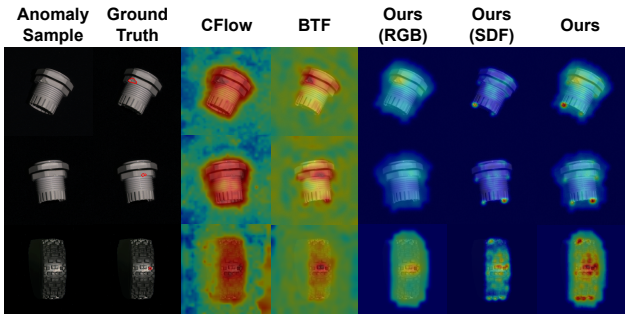

Figure 9. Failure cases of our method.

5. Additional Ablation

5.1. Benefit of Combining RGB and 3D Information

In the MVtec 3D-AD dataset, most anomalies appear simultaneously in geometric structure and color. However, some anomalies only appear in one of the two modalities. As previously shown in Figure 1, using the 2D color-based method cannot detect geometric anomalies, such as holes in cookies that resemble chocolate chips, which cannot be unambiguously detected from color cues. On the other hand, the discoloration defect on foam also cannot be detected using a pure 3D method. Thus we use the proposed dual-expert learning and score alignment on the RGB and SDF scores to combine the advantage of both modalities.

Table 5. Ablation study of our method on the MVTec 3D-AD dataset with the evaluation metric of AUPRO at the integration limit of 0.3.

Method	Bagel	Cable Gland	Carrot	Cookie	Dowel	Foam	Peach	Potato	Rope	Tire	**Mean**
RGB only	0.946	0.972	0.96	0.914	0.958	0.776	0.937	0.949	0.956	0.957	0.933
SDF only	0.974	0.871	0.981	0.924	0.898	0.773	0.978	**0.983**	0.955	0.969	0.931
Shape-guided RGB only	0.948	0.972	0.959	0.91	0.961	0.774	0.936	0.95	0.954	0.956	0.932
Shape-guided RGB + SDF	**0.981**	**0.973**	**0.982**	**0.971**	**0.962**	**0.978**	**0.981**	**0.983**	**0.974**	**0.975**	**0.976**

Table 6. Ablation study of our method adopting sparse coding (SC) or nearest neighbor (NN) in the RGB and SDF features, respectively, for computing the distance from the query feature to the normal feature set. Using sparse-coding reconstructed features is better than using the nearest neighbor for RGB and SDF features.

RGB NN	RGB SC	SDF NN	SDF SC	Img-ROC	Pix-ROC	AUPRO (0.3)	AUPRO (0.01)
✓		✓		0.937	0.994	0.972	0.434
✓			✓	0.944	0.994	0.972	0.443
	✓	✓		0.942	0.996	0.975	0.451
	✓		✓	**0.947**	**0.996**	**0.976**	**0.456**

In Table 5, we compare the AUPRO scores of our method with those of the RGB-only, SDF-only, and shape-guided RGB-only methods. Figure 10 also shows that the combined Img-AUROC scores exhibit significant margins between the anomalous and the normal data distributions. From these experimental results, we can see that our dual-expert models' synergy improves the performance by effectively combining the 2D color-based and the 3D geometry-based information.

5.2. The Effectiveness of Sparse Coding

Instead of directly computing the distance between the target feature and its nearest feature in the memory bank as the anomaly score, we use sparse coding to reconstruct the target feature for both RGB and SDF features. Since the sparse coding uses a dictionary derived from the normal features in the memory bank, the sparse representation can accurately describe an anomaly-free feature. The sparse representation helps the feature extracted from the non-defective local region generalize better to its reconstructive counterpart, allowing us to distinguish the normal and anomaly more reliably. Table 6 shows the results of adopting sparse coding or nearest neighbor in the RGB and SDF features, respectively, for computing the distance from the query feature to the normal feature set. Consequently, using reconstructed features by sparse coding is better than using its nearest neighbor for RGB and SDF features.

6. Conclusion

We have presented a new method to achieve the state-of-the-art performance of unsupervised 3D anomaly detection on the MVTec 3D-AD dataset. Our method has better recall rates and lower false-positive rates, which is preferable in real applications requiring precise localization of defects. Furthermore, the proposed framework is efficient, as our implementations of the dual memory banks and the shape-guide inference significantly reduce the computation and memory costs. We have shown that using neural implicit functions to model 3D local shapes is a great advantage in detecting detailed irregularities in point clouds. This work also provides hands-on techniques for fusing predictions from different modalities, which, with the new shape-guided expert learning framework, may benefit future development in solving the unsupervised 3D anomaly detection task.

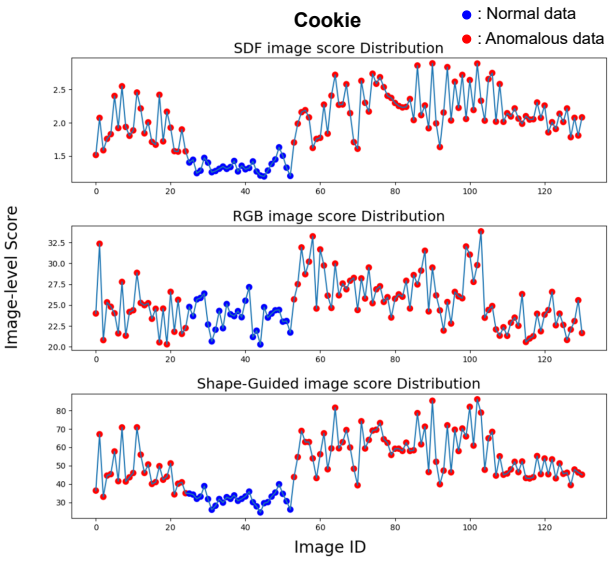

Figure 10. Distributions of anomaly detection scores (Img-AUROC). The fused Img-AUROC score distributions of shape-guided RGB+SDF models (bottom row) have a larger margin separating anomalous and normal data, meaning better anomaly detection performance.

Acknowledgements

This work was supported in part by NSTC grants 111-2221-E-001-011-MY2 and 111-2634-F-007-010 of Taiwan. We are grateful to National Center for High-performance Computing for providing computational resources and facilities.

References

Behrendt, F., Bengs, M., Rogge, F., Krüger, J., Opfer, R., and Schlaefer, A. Unsupervised anomaly detection in 3d brain MRI using deep learning with impured training data. In *ISBI*, 2022.

Bengs, M., Behrendt, F., Laves, M., Krüger, J., Opfer, R., and Schlaefer, A. Unsupervised anomaly detection in 3d brain MRI using deep learning with multi-task brain age prediction. *CoRR*, 2022.

Bergmann, P. and Sattlegger, D. Anomaly detection in 3d point clouds using deep geometric descriptors. *CoRR*, 2022.

Bergmann, P., Fauser, M., Sattlegger, D., and Steger, C. MVTec AD - A comprehensive real-world dataset for unsupervised anomaly detection. In *CVPR*, 2019.

Bergmann, P., Fauser, M., Sattlegger, D., and Steger, C. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *CVPR*, 2020.

Bergmann, P., Batzner, K., Fauser, M., Sattlegger, D., and Steger, C. The MVTec anomaly detection dataset: A comprehensive real-world dataset for unsupervised anomaly detection. *IJCV*, 2021.

Bergmann, P., Jin, X., Sattlegger, D., and Steger, C. The MVTec 3D-AD dataset for unsupervised 3d anomaly detection and localization. In *17th Int. Joint Conf. Comput. Vis., Imaging and Comput. Graph. Theory and Appl., VISIGRAPP 2022, Volume 5: VISAPP*, 2022.

Defard, T., Setkov, A., Loesch, A., and Audigier, R. Padim: A patch distribution modeling framework for anomaly detection and localization. In *ICPR*, 2020.

Gudovskiy, D. A., Ishizaka, S., and Kozuka, K. CFLOW-AD: real-time unsupervised anomaly detection with localization via conditional normalizing flows. In *WACV*, 2022.

Horwitz, E. and Hoshen, Y. Back to the feature: Classical 3d features are (almost) all you need for 3d anomaly detection. *CoRR*, abs/2203.05550v3, 2022.

Jiang, C. M., Sud, A., Makadia, A., Huang, J., Nießner, M., and Funkhouser, T. A. Local implicit grid representations for 3d scenes. In *CVPR*, 2020.

Lee, S., Lee, S., and Song, B. C. CFA: coupled-hypersphere-based feature adaptation for target-oriented anomaly localization. *IEEE Access*, 2022.

Li, C., Sohn, K., Yoon, J., and Pfister, T. Cutpaste: Self-supervised learning for anomaly detection and localization. In *CVPR*, 2021.

Li, K., Tang, Y., Prisacariu, V. A., and Torr, P. H. S. Bnv-fusion: Dense 3d reconstruction using bi-level neural volume fusion. In *CVPR*, 2022.

Ma, B., Han, Z., Liu, Y., and Zwicker, M. Neural-pull: Learning signed distance function from point clouds by learning to pull space onto surface. In *ICML*, 2021.

Ma, B., Liu, Y., Zwicker, M., and Han, Z. Surface reconstruction from point clouds by learning predictive context priors. In *CVPR*, 2022.

Pang, Y., Wang, W., Tay, F. E. H., Liu, W., Tian, Y., and Yuan, L. Masked autoencoders for point cloud self-supervised learning. In *ECCV*, 2022.

Qi, C. R., Su, H., Mo, K., and Guibas, L. J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017.

Roth, K., Pemula, L., Zepeda, J., Schölkopf, B., Brox, T., and Gehler, P. V. Towards total recall in industrial anomaly detection. In *CVPR*, 2022.

Rudolph, M., Wehrbein, T., Rosenhahn, B., and Wandt, B. Asymmetric student-teacher networks for industrial anomaly detection. In *WACV*, 2023.

Schlüter, H. M., Tan, J., Hou, B., and Kainz, B. Natural synthetic anomalies for self-supervised anomaly detection and localization. In *ECCV*, 2022.

Shi, H., Zhou, Y., Yang, K., Yin, X., and Wang, K. Csflow: Learning optical flow via cross strip correlation for autonomous driving. In *IEEE Intell. Vehicles Symposium*, 2022.

Takikawa, T., Litalien, J., Yin, K., Kreis, K., Loop, C. T., Nowrouzezahrai, D., Jacobson, A., McGuire, M., and Fidler, S. Neural geometric level of detail: Real-time rendering with implicit 3d shapes. In *CVPR*, 2021.

Viana, J. S., de la Rosa, E., Vyvere, T. V., Robben, D., and Sima, D. M. Unsupervised 3d brain anomaly detection. In Crimi, A. and Bakas, S. (eds.), *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries - 6th International Workshop with MICCAI*, 2020.

Yang, M., Wu, P., Liu, J., and Feng, H. Memseg: A semi-supervised method for image surface defect detection using differences and commonalities. *CoRR*, 2022.

Zagoruyko, S. and Komodakis, N. Wide residual networks. In Richard C. Wilson, E. R. H. and Smith, W. A. P. (eds.), *Proceedings of the British Machine Vision Conference (BMVC)*, 2016.

Zheng, Y., Wang, X., Qi, Y., Li, W., and Wu, L. Benchmarking unsupervised anomaly detection and localization. *CoRR*, 2022.