Scientific Research Publishing

Open Journal of Applied Sciences, 2025, 15(3), 715-730
https://www.scirp.org/journal/ojapps
ISSN Online: 2165-3925
ISSN Print: 2165-3917

# Unsupervised Anomaly Detection Algorithm Based on Bidirectional Knowledge Distillation Network

**Hao Zhong, Shuai Kang, Ao Xiong**

College of Railway Transportation, Hunan University of Technology, Zhuzhou, China
Email: hownzcc@163.com

**How to cite this paper:** Zhong, H., Kang, S. and Xiong, A. (2025) Unsupervised Anomaly Detection Algorithm Based on Bidirectional Knowledge Distillation Network. *Open Journal of Applied Sciences*, **15**, 715-730.
https://doi.org/10.4236/ojapps.2025.153046

**Received:** March 4, 2025
**Accepted:** March 18, 2025
**Published:** March 21, 2025



## Abstract

Industrial appearance anomaly detection (AD) focuses on accurately identifying and locating abnormal regions in images. However, due to issues such as scarce abnormal samples, complex abnormal manifestations, and difficult abnormal annotation, the detection accuracy is limited. To solve these problems, based on the knowledge distillation framework, this paper proposes an unsupervised anomaly detection algorithm—Bidirectional knowledge distillation AD (BKD). This algorithm combines the advantages of forward and reverse distillation, enabling efficient anomaly detection. Experimental results have shown that the proposed method outperforms the state-of-the-art AD methods by 3% - 8% in AUC on the MVTec benchmarks.

## Keywords

Anomaly Detection, Knowledge Distillation, Unsupervised Learning

## 1. Introduction

Modern industrial products, such as aircraft wings and semiconductor wafers, are widely used in social infrastructure. Their quality directly impacts production stability and safety. Industrial anomaly detection technology is a core component of quality assurance. Its importance is increasingly recognized. Traditional manual screening methods are inefficient, costly, and difficult to scale. Recently, unsupervised anomaly detection techniques have emerged, requiring only normal samples for model training. This approach improves detection accuracy while significantly reducing the cost of collecting and annotating anomaly samples.

Unsupervised anomaly detection algorithms [1] identify and localize anomalies without prior knowledge. Among them, based on deep learning methods [2] have

achieved significant progress in anomaly detection tasks. However, their high computational resource demands are often hard to meet in practice. Methods based on One-Class Classification [3]-[5] have lower computational costs. Thus, they are widely used in image-level anomaly detection. OCSVM (One-Class Support Vector Machine) [3] and Deep SVDD (Deep Support Vector Data Description) [5] are two typical approaches. These methods train feature extraction networks to map data into feature spaces. They construct hyperplanes or spheres in the feature space. Data outside these boundaries are classified as anomalies, while data inside are considered normal. Methods based on Feature embedding [6]-[8] use networks to extract high-dimensional features from normal samples, analyze the feature space, and minimize intra-class distances. Patchcore [6] extracts feature through a pre-trained network to build a feature memory bank, which is compressed using a greedy coreset mechanism. Anomaly scores are obtained by measuring distances between test image patch features and the memory bank. Padim [7] leverages a pre-trained CNN to extract multi-scale features, constructs parametric Gaussian distributions, and detects anomalies using Mahalanobis distance to measure deviations from normal feature distributions. SPADE [8] integrates the KNN algorithm with a multi-scale feature pyramid, storing normal sample features in a feature library. During testing, it retrieves k-nearest neighbors from the library to compute anomaly scores. However, finding suitable feature spaces and decision boundaries remains challenging in complex datasets.

To address this challenge, researchers resort to knowledge distillation (KD) [9] to transfer the representational capacity of large pre-trained networks to lightweight student networks, while leveraging the discrepancies between teacher and student representations for anomaly detection. The Student-Teacher (S-T) paradigm [10] achieves exceptional speed-accuracy trade-offs, enabling efficient inference without compromising performance, thus becoming a cornerstone of unsupervised anomaly detection (AD). US [11] pioneered the integration of KD into unsupervised AD frameworks. Subsequent innovations include MKD [12] and STPM [13], which mitigate student network over-generalization through multi-scale feature alignment and structurally asymmetric architectures. Similarly, RD [14] introduced an inverse distillation framework, where the teacher and student roles are assigned to encoder and decoder modules, respectively.

These KD-based methods exclusively train student networks on normal data, premised on the hypothesis that student networks fail to replicate the teacher's representational capacity for anomalous regions. Consequently, they maintain high feature consistency on normal samples but exhibit pronounced discrepancies in anomalies. However, CNNs' inherent inductive biases and data consistency constraints may lead student networks to unintentionally learn anomalous feature patterns in practice, contradicting the foundational assumption. Current approaches thus tackle over-generalization from two perspectives: architectural design and knowledge disentanglement.

1) From the perspective of network architecture, forward distillation [11] di-

rectly mimics the output of the teacher network in Figure 1(a). Although simple, this approach often leads to over-generalization in the student network, making it difficult to exclude abnormal interference. Reverse distillation [14] recovers features from the embeddings of the teacher network in Figure 1(b). It leverages asymmetry to differentiate the information capacity between the teacher and student networks. The OCBE module provides compact feature representations, reducing inter-modal feature similarity and thereby preventing over-generalization. However, due to capacity differences and the direction of abnormal information transmission, reverse distillation may introduce pseudo abnormal features, causing the normal features of the student network to deviate from those of the teacher network.

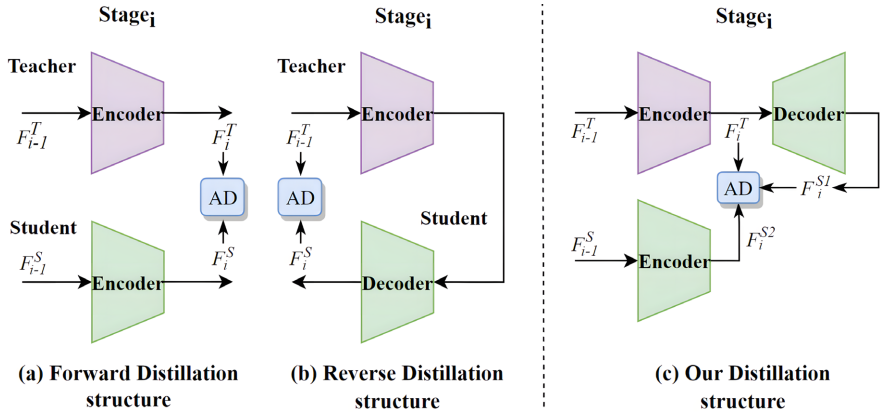



Figure 1. (a) Forward distillation structure. (b) Reverse distillation structure. (c) Our proposed bidirectional distillation method.

2) From the perspective of content information, some methods improve the basic S-T network by extracting content from both normal and abnormal data. Memory Knowledge Distillation (MemKD) [15] addresses the "normal forgetting" problem. It uses a memory bank to guide the student to generate normal features with teacher features, indirectly amplifying feature differences in abnormal regions. Decouple Distillation (DMDD) [16] introduces anomaly synthesis into the Knowledge Distillation (KD) paradigm, explicitly distinguishing between student and teacher features in abnormal regions. However, in this paradigm, the student network merely imitates the current teacher network without fully considering the diversity of samples. This defect leads to a single form of knowledge representation, and the learned knowledge lacks flexibility. As a result, when facing actual abnormal scenarios, especially those with rotation or complex backgrounds, its performance is unsatisfactory.

To solve these problems, we make the following improvements:

1) In structure, we combine the advantages of forward and reverse distillation to construct a bidirectional distillation network, as Figure 1(c). Forward distillation is used to detect region-level abnormalities, and reverse distillation is used to detect pixel-level abnormalities. The bidirectional design effectively avoids the

wrong injection of pseudo abnormal features.

2) In content, we propose an information alignment distillation method based on few sample registration. By aligning normal features for distillation, we can fully utilize the diversity of the teacher network and prevent the student network from overgeneralizing. Meanwhile, we combine the memory module to alleviate the "normal forgetting" problem. In summary, through comprehensive detection at the regional (coarse) and pixel (fine) levels, our approach overcomes the limitations of existing KD solutions.

## 2. Our Approach

We constructed a bidirectional knowledge distillation network (BKD) for unsupervised anomaly detection (AD), and its architecture is shown in Figure 2.

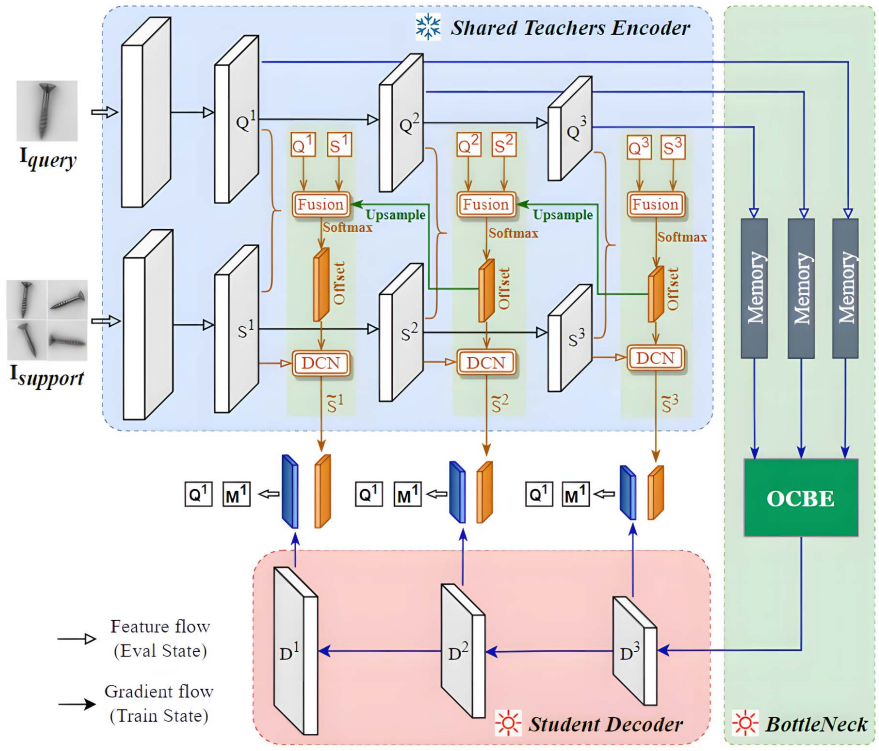



Figure 2. The structure of bidirectional distillation network.

During training, only normal images are fed into the model. In contrast, the test set contains both normal images and abnormal images unseen during training. The training objective of the unsupervised AD model is to enable it to detect and locate abnormal regions during inference.

### 2.1. Framework Overview

The network mainly consists of three modules: the shared teacher encoder, the feature bottleneck, and the student decoder.

The proposed framework operates through two core stages, as illustrated in Fig-

ure 2. The forward distillation stage (yellow arrows) is executed within the shared teacher encoder, where knowledge propagation and hierarchical feature learning are achieved by jointly optimizing the fusion module and deformable convolution network (DCN) [17] [18]. In contrast, the reverse distillation stage (blue arrows) facilitates inverse performance optimization via coordinated training of the feature bottleneck module and the student decoder. Notably, the encoder utilizes a pre-trained model with frozen parameters, while all other modules dynamically update their parameters throughout the training process. The subsequent sections elaborate on the implementation mechanisms of these dual-stage distillation paradigms, followed by a comprehensive description of the loss function design and the anomaly localization methodology.

## 2.2. Forward Distillation

Forward distillation is carried out in the multi-teacher encoding part. Inspired by the way humans detect anomalies in images, a simple and effective strategy is to compare the samples to be detected with normal samples to find differences. Therefore, we randomly select a pair of images $\left(I_{query}, I_{support}\right)$ from the given training set of normal samples in the same category, serving as the query set and the support set respectively, with the number of the support set denoted by k-shot.

During training, the teacher encoder uses the first four stages of WideResNet50 pretrained on ImageNet as the teacher network. First, a query—support image pair is input. The support set is randomly drawn from normal samples and undergoes data augmentation through translation, rotation, scale transformation, etc. Subsequently, these two groups of data are input into the teacher network sharing the same network weights.

We extract the features of the query set and the support set through the shared teacher network. Since the weights of the teacher network are frozen during training, its semantic information is consistent, only differing in spatial distribution. Based on this, we propose an information alignment distillation method under few shot registration to achieve semantic consistency between the support set and the query set. During testing, the semantics of normal regions are easy to align, while those of abnormal regions have large alignment errors. We can use this characteristic to locate abnormal regions. This design not only provides constraints on abnormal regions for subsequent distillation but also effectively suppresses the over generalization of the student network. The feature alignment network involves two key steps: feature fusion and feature alignment, which occur as indicated by the yellow lines in **Figure 2**.

In the feature fusion part, the feature maps from the query set and the support set at the same level are collected, denoted as $Q^L$ and $S_k^L$ respectively. Here, $L$ represents the feature maps generated by the encoder at different stages, and $k$ is the number of samples in the support set. The fusion operation of the feature maps is as follows:

$$F_{offset}^{L} = f\left(\left[Q^L, S_k^L\right]\right) = \mathrm{Conv}\left(\mathrm{Concat}\left(Q^L, S_k^L\right), W\right), L \in [1:3] \tag{1}$$

In the formula, $[\cdot,\cdot]$ represents the concatenation operation of feature maps, $f$ represents the convolution operation, and $W$ is the weight of the convolution kernel. The objective of the fusion stage is to obtain a deviation feature map containing offset information. This feature map is helpful for identifying and locating the deviations between features in subsequent processing.

In the design of the feature alignment network, STN [19] is often the core module. Yet, it has limitations in handling image edges and local details, and CNN also struggles with geometric transformations. In this study, DCN [17] [18] is used to replace STN. DCN introduces learnable offset parameters, which can dynamically adjust the positions of convolutional kernels to achieve adaptive feature alignment. This overcomes the limitations of traditional convolution, improves the ability to model geometric transformations, and enhances the accuracy and robustness of feature alignment.

More precisely, In this study, the support set $S_k^L$ is regarded as the dataset to be registered, and the query set $Q^L$ is taken as the reference dataset. The aim is to precisely align the features between the support set and the query set.

To achieve this goal, a modulated deformable convolution module is adopted and applied to each shot in the support set. Assume that the deformable convolutional kernel has $m$ sampling positions, and its weights and offsets are obtained through pre-learning. The $m$-th position is represented as $\omega_m$, $p_m = \{(-1,-1),(-1,0),\cdots,(1,1)\}$. For a $3\times3$ convolutional kernel, the total number of sampling points is defined as 9.

Based on this, the aligned feature $\tilde{S}_k^L$ at position $p_0$ in the feature map of layer $L$ can be calculated by the following formula:

$$\tilde{S}_k^L = \mathrm{DConv}\left(S_k^L, \Delta p_m^L\right) = \sum_{m=1}^{\text{kernel}} \omega_m \cdot \left[S_k^L \otimes \left(p_0 + p_m + \Delta p_m^L\right)\right] \cdot \Delta m_m^L \tag{2}$$

Among them, The symbol $\otimes$ represents the convolution operation between the convolutional kernel and the feature map. $\Delta m_m^L$ is the modulation factor, which is predicted together with $\Delta p_m^L$ by DCNv1 [17]. $\Delta p_m^L$ represents the learnable offset of the $m$-th sampling point in layer $L$, which is obtained by activating the feature map $F_{offset}^L$, that is,

$$\Delta p_m^L = \mathrm{softmax}\left(F_{offset}^L\right),\ L \in [1:3] \tag{3}$$

For simplicity, only the learnable offset $\Delta p_m^L$ is considered, and the modulation factor $\Delta m_m^L$ is ignored. Since the position $\left(p_0 + p_m + \Delta p_m^L\right)$ in the offset calculation may be a decimal, bilinear interpolation is used to obtain the actual offset.

In the alignment task, to tackle complex motion and large parallax, models with a large effective receptive field are found to perform better. Two improvements are proposed:

First, construct an $L$-level pyramid using three downsampling layers of the backbone network to capture long range dependencies and address large parallax.

Second, adopt a cascaded refinement strategy. In layer $L$, combine the ×2 up-

sampled offsets and features from the previous layer to predict the current layer offsets and features. The specific formulas are as follows:

$$\Delta p_m^L = f\left(\left[Q^L, S_k^L\right]\right) \odot \left(\Delta p_m^{L+1}\right)^{\uparrow 2} \tag{4}$$

$$\tilde{S}_k^L = \mathrm{DConv}\left(S_k^L, \Delta p_m^L\right) \tag{5}$$

Here, $(\cdot)^{\uparrow s}$ represents the upsampling scale factor $s$, and ×2 upsampling is achieved through bilinear interpolation in this case. DConv is the deformable convolution operation defined in Equation (2). subsequent offsets are cascaded to further refine the aligned features (as shown in the part of the green lines in the **Figure 1**).

## 2.3. Reverse Distillation

As shown in the figure, reverse distillation consists of the feature bottleneck module and the student decoder module. The bottleneck module modulates the feature space using the memory-guided mechanism and fuses features with the OCBE (Orthogonal Channel Bottleneck Embedding) method. The student decoder recovers the features embedded in the bottleneck during training.

To increase the feature differences in abnormal modalities between the student and teacher networks, this study integrates a memory guided module into reverse distillation. The structure of the module is shown in **Figure 3**.

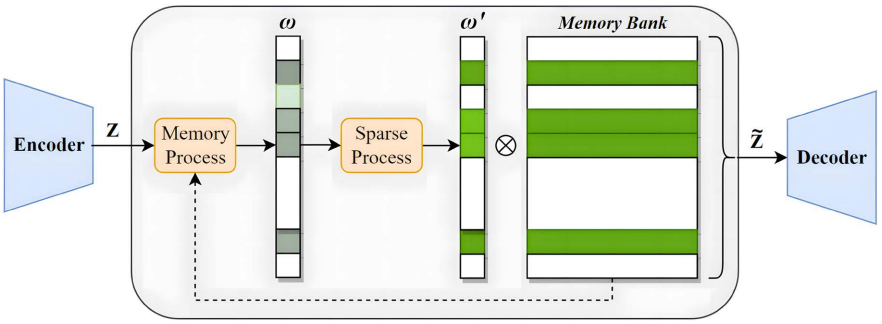



**Figure 3.** The structure of memory module.

Define the vector space as $\mathbb{R}$ and introduce a memory matrix $M \in \mathbb{R}^{I \times N}$. Here, $I$ is the number of row vectors in $M$. Each row vector is a memory unit, and its dimension is the same as that of the latent feature vector $z$ output by the encoder.

The memory processing module generates the attention coefficient $\omega$ by calculating the similarity between the query vector $z$ and each memory unit. The calculation formula is as follows:

$$\omega_i = \mathrm{Softmax}\left(d\left(z, m_i\right)\right) \tag{6}$$

$$d\left(z, m_i\right) = \frac{z m_i^{\mathrm{T}}}{\|z\| \cdot \|m_i\|} \tag{7}$$

where $z \in \mathbb{R}^{1\times N}$ is the query vector, $m_i \in \mathbb{R}^{1\times N}$ is the $i$-th row vector of the memory matrix $M$, $d(z, m_i)$ represents the cosine similarity between them, and $\omega_i$ is the $i$-th element of the attention coefficient $\omega$.During the training process, only normal samples are used to update the memory matrix $M$.

To avoid the small and dense attention coefficient $\omega$ interfering with the reconstruction of the features of abnormal samples, a sparsification strategy is adopted to eliminate insignificant attention coefficients, which are finally obtained after activation. The specific calculation formulas are as follows:

$$\omega_i' = \begin{cases} \omega_i & \omega_i > \xi \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

$$\omega_i' = \frac{\text{Relu}(\omega_i - \xi)\cdot\omega_i}{|\omega_i - \xi| + \delta} \tag{9}$$

where $\xi = 0.002$, and $\delta = 10^{-12}$. Finally, multiply the sparsely processed attention coefficient $\omega$ by the memory matrix to obtain the new latent feature $z'$. The formula is as follows:

$$z' = \omega' M = \sum_i^N \omega_i' m_i \tag{10}$$

Subsequently, a One-Class Bottleneck Embedding (OCBE) module is employed to collect feature information, and its structure is shown in Figure 4. Before single-stage embedding, OCBE enhances feature diversity with the aid of a Multi-Scale Feature Fusion (MFF) block. First, the module uses a convolutional layer to downsample the shallow layer features. Then, batch normalization and ReLU activation are performed on the downsampled features to align the connected feature representations. Subsequently, a convolutional layer is used for feature dimensionality reduction. Finally, after batch normalization and ReLU activation, a compact feature representation is generated and transmitted to the decoder.

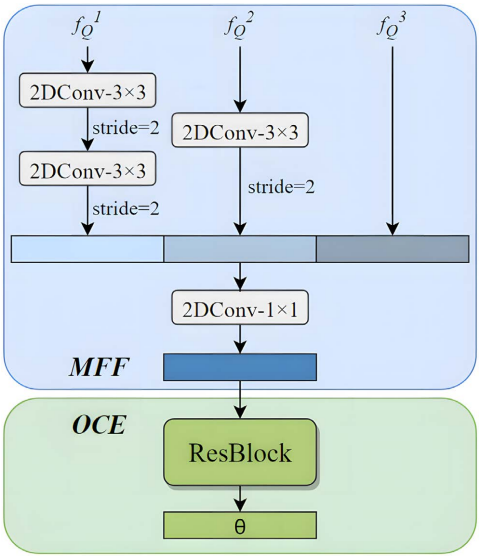



Figure 4. The structure of OCBE.

During training, the OCBE module and the decoder are optimized collaboratively to improve the accuracy and robustness of feature representations.

### 2.4. Loss Function and Anomaly Graph Calculation

During the training stage, the loss value is calculated based on the loss constraints of features from multiple intermediate layers to simulate the generalization representation of the teacher network for normal samples. The formula for the total loss is:

$$L_{total} = Loss^1 + Loss^2 + Loss^3 \tag{11}$$

Here, $Loss^1$, $Loss^2$ and $Loss^3$ correspond to the output features $Q^1$, $Q^2$, $Q^3$ of the query set and the distillation output features $M^1$, $M^2$, $M^3$ respectively. The distillation output feature $M^i\left(i \in 1,2,3\right)$ consists of the output feature $\tilde{S}^i\left(i \in 1,2,3\right)$ of the support set after modulation and alignment and the feature $D^i\left(i \in 1,2,3\right)$ output by the student decoder. The loss of the $i$-th layer is:

$$Loss^i = Loss\left(Q^i, D^i\right) + Loss\left(Q^i, \tilde{S}^i\right) \tag{12}$$

Each loss term $Loss^i$ is composed of a value loss ( $L_{val}$ ) and a direction loss ( $L_{dir}$ ).

During the testing stage, the anomaly map ( $\Omega$ ) is obtained through calculation, with a size of $w \times h$. The pixel point $x_{i,j}$ in the input feature map represents the feature value at the position $\left(i, j\right)$. The calculation formula for the anomaly map is:

$$\Omega\left(x\right) = \prod_{n=1}^{3} \text{Upsameple}\left(\Omega_{i,j}^n\right) \tag{13}$$

where $\Omega_{i,j}^n$ represents the degree to which the pixel point at the position $\left(i, j\right)$ in the output feature map of the $n$-th convolutional layer deviates from the normal data flow. Finally, the anomaly maps from three different layers are upsampled to a unified size and then multiplied for fusion to generate a comprehensive anomaly detection result.

## 3. Experimentation

### 3.1. Experimental Environment and Dataset

Our evaluation employs two widely-recognized anomaly detection (AD) benchmarks derived from real industrial scenarios, both designed for defect identification tasks.

- **MVTec AD** [20] dataset, a publicly available industrial quality inspection dataset, is specifically designed for training and evaluating unsupervised anomaly detection algorithms. To verify the effectiveness of the model proposed in this paper, this dataset is used for experimental validation.
- **MPDD** [21] dataset, a recently introduced benchmark, is tailored for detecting surface defects in painted metal components during manufacturing processes, covering six distinct categories. It comprises images captured under diverse

spatial configurations (e.g., orientations, positions, distances) of objects, alongside varying illumination conditions and non-uniform background environments.

### 3.2. Experimental Setting and Evaluation Criteria

In this study, the resolutions of all images in the MVTec AD [20] dataset were adjusted to ensure consistent input. The experiments followed the common procedures in the field of anomaly detection, performing anomaly detection and localization tasks on the data category by category. To optimize memory usage, the size of the memory cache item was set to $M = 200$. In terms of the optimization strategy, the Adam optimizer was selected, with hyperparameters $\beta = (0.5, 0.999)$ and an initial learning rate of 0.005. The model was trained for 200 epochs with a batch size of 16.

For industrial image anomaly detection, it's necessary not only to identify anomalies but also to localize and segment abnormal regions. To comprehensively evaluate the performance of the algorithm, we adopt the following evaluation metrics:

- **ImageAUC** is an image-level classification metric used to evaluate the algorithm's ability to detect anomalies in the whole image. Output is the anomaly value matrix output by the algorithm for the whole image, Label is the ground-truth defect label. as shown in Equation (14).
- **PixelAUC** is a pixel-level segmentation metric, it measures the precision of the algorithm in segmenting abnormal regions. TP represents the number of pixel values in the defect area correctly detected by the algorithm, and FP represents the number of pixel values in the defect area that the algorithm fails to detect. as shown in Equation (15).
- **F1-Score** combines the segmentation accuracy and false detection effect of the algorithm. Precision in the formula is related to the pixel-level index, as Equation (15), and recall is related to the false-detection rate, as Equation (16). as shown in Equation (17).

$$\text{ImageAUC} = \frac{\text{rank}\left(\text{Output} \bigcap \text{Label}\right)}{D_n} \tag{14}$$

$$\text{PixelAUC} = \text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{15}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{16}$$

$$\text{F1-Score} = \frac{2\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{17}$$

### 3.3. Experimental Results and Analysis

To evaluate the effectiveness of the algorithm in this paper, the US [11] and MKD [12] method based on distillation, and the DiffNet [22] method based on few shot

registration are selected as comparative experiments.

According to the AUC index in Table 1, the BKD (Bidirectional Distillation) algorithm has significant advantages over the MKD and US methods.

Table 1. Comparison of training results of different models in terms of AUC at image-level/pixel-level on the MVTec dataset.

| Image Size | | 128 | | | 256 | | |
|---|---|---|---|---|---|---|---|
| Category/Method | | US [11] | MKD [12] | BKD (ours) | US [11] | MKD [12] | BKD (ours) |
| Textures | Carpet | -/87.4 | 95.7/- | **95.7/94.8** | -/89.9 | **97.9**/- | 97.6/**95.9** |
| | Grid | -/**95.0** | 92.8/- | **96.7**/92.7 | -/96.2 | 93.7/- | **98.3/96.8** |
| | Leather | -/94.5 | **96.9**/- | 96.8/**98.4** | -/95.5 | **97.6**/- | 97.3/**99.0** |
| | Carpet tile | -/**93.2** | 90.0/- | **93.2**/90.8 | -/**94.6** | 92.4/- | **96.4/95.1** |
| | Grid wood | -/**89.3** | 85.8/- | **92.6**/89.0 | -/91.1 | 90.5/- | **94.9**/95.1 |
| | Average | -/**91.9** | 92.2/- | **95.0/93.1** | -/93.5 | 94.4/- | **96.9/94.2** |
| Objectes | Bottle | -/92.3 | 96.3/- | **97.8/95.1** | -/93.1 | **98.4**/- | **98.0**/97.0 |
| | Cable | -/81.6 | 92.5/- | **96.8/91.3** | -/86.8 | **97.2/-** | 96.8/**90.5** |
| | Capsule | -/**95.5** | 95.9/- | **96.4**/87.6 | -/**96.8** | **99.0**/- | 98.8/94.7 |
| | Hazelnut | -/**93.3** | 96.6/- | **98.3**/90.9 | -/**96.5** | **99.0**/- | 98.8/94.5 |
| | Metal nut | -/**92.0** | 95.4/- | **95.9**/89.0 | -/**94.2** | **98.1/-** | 96.9/92.0 |
| | Pill | -/**95.3** | 92.6/- | **96.9**/92.0 | -/96.1 | 96.5/- | **98.0/96.3** |
| | Screw | -/92.1 | 96.0/- | **97.8/94.6** | -/94.2 | **98.9**/- | 98.6/**98.0** |
| | Toothbrush | -/**92.3** | 96.1/- | **98.0**/86.6 | -/93.0 | 97.8/- | **99.0/93.3** |
| | Transistor | -/60.6 | 90.6/- | **95.5/86.0** | -/66.6 | **94.0/-** | 92.1/**68.0** |
| | Zipper | **-/94.4** | **93.9**/- | 93.0/91.6 | **-/95.1** | 96.5/- | **98.1/95.1** |
| | Average | **-/88.9** | 94.6/- | **96.6**/90.4 | **-**/91.2 | **97.5/-** | **97.5/91.8** |
| Total average | | **-/90.4** | 93.5/- | **95.8/91.8** | **-**/92.4 | 96.0/- | **97.2/93.0** |

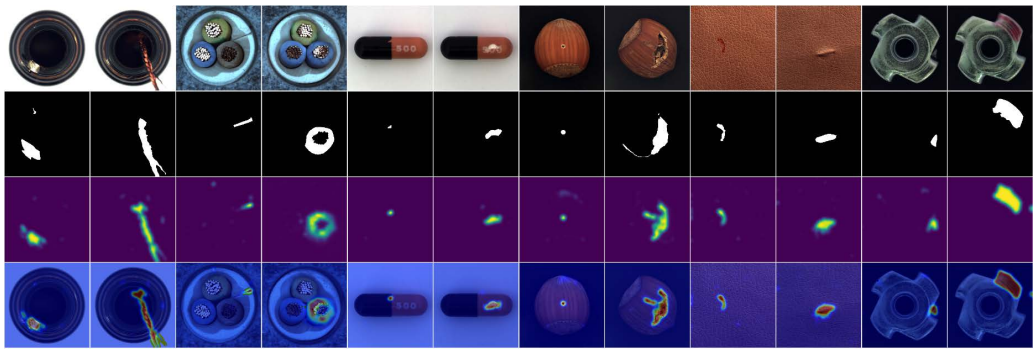


Figure 5. The result of our method on the MVTec dataset.

Compared with other methods, our method improves the pixel-level and image-level AUROC indices at resolutions of 128 and 256. At high resolutions, the average indices perform well. Notably, at low resolutions, the image-level scores

are more prominent. This is due to the fact that our method utilizes bidirectional knowledge distillation, giving full play to the advantages of anomaly detection. The experimental results are presented in **Figure 5**.

Meanwhile, to verify the effectiveness of the method using few shot support set registration, the DiffNet [22] method based on few shot registration is selected for comparative experiments. The MVTec dataset is used for the F1-Score index test. One target category is selected, and support sets of different sizes are used for testing. The experimental results are shown in the following **Table 2**. Compared with DiffNe [22], the F1-Score index of BKD on the MVTec dataset has increased by 3.2%, 4.1%, and 3.8% respectively.

**Table 2.** Comparison of results under different supported sets in terms of F1-Score on the MVTec dataset.

| Category | $k = 2$ | | $k = 4$ | | $k = 8$ | |
|---|---|---|---|---|---|---|
| | DiffNet+ [22] | BKD (ours) | DiffNet+ [22] | BKD (ours) | DiffNet+ [22] | BKD (ours) |
| Carpet | 56.3 | **61.4** | 59.0 | **68.9** | **69.5** | 69.0 |
| Carpet tile | **59.8** | 58.5 | **63.9** | 60.6 | 66.3 | **67.8** |
| Grid wood | 40.9 | 55.7 | 50.9 | **60.2** | **69.0** | 66.3 |
| Toothbrush | 54.0 | **74.8** | 54.8 | **79.3** | 54.5 | **83.8** |
| Transistor | **96.6** | 80.3 | **98.2** | 68.0 | **98.0** | 81.2 |
| Zipper | 48.8 | **68.9** | 49.9 | **70.4** | 52.5 | **74.5** |
| Average | 59.4 | **66.6** | 62.8 | **67.9** | 68.3 | **72.1** |

In cross-dataset benchmarking across MVTec AD and MPDD, we compared the proposed algorithm with state-of-the-art methods, using average F1-Score as the evaluation metric. **Table 3** demonstrate that the synergistic optimization strategy combining few-shot registration and bidirectional distillation achieves remarkable robustness in complex industrial defect detection scenarios, particularly maintaining stable classification boundaries under high-noise and multi-scale deformation conditions.

**Table 3.** Results of MVTec and MPDD dataset under the average F1-score index ($k = 2$).

| Dataset | US [11] | MKD [12] | DiffNet [22] | BKD (ours) |
|---|---|---|---|---|
| MVTec [20] | 66.4 | 73.6 | 80.6 | **87.4** |
| MPDD [21] | 50.6 | 57.7 | 59.4 | **62.8** |

## 3.4. Ablation Experiment

In this section, an ablation study of anomaly detection and localization is carried out on the MVTEC dataset.

As shown in **Table 4**, the "Augment" module refers to support-set augmentation, the "Memory" module is the memory module, and the "DCN" is the feature

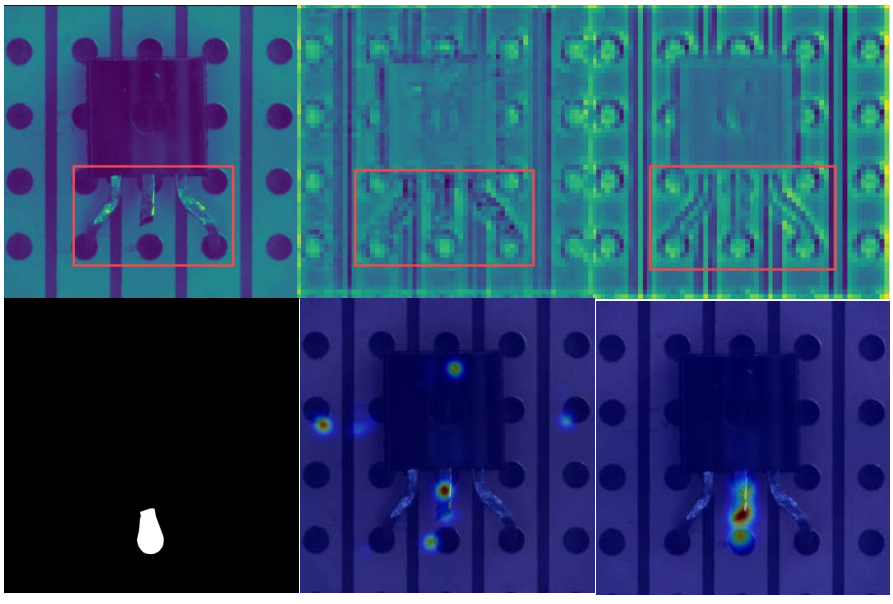
alignment network. For all categories within the dataset, the macro average AUC values are presented. The methods highlighted in bold demonstrate the best performance.

The results in the table clearly indicate that these modules significantly enhance the detection metrics.

**Table 4.** Ablation studies of anomaly detection.

| Category | | | MVTec [20] | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | image | | | pixel | | |
| Augment | Memory | DCN | $k=2$ | $k=4$ | $k=8$ | $k=2$ | $k=4$ | $k=8$ |
| | | | 75.0 | 77.8 | 79.1 | 89.0 | 91.2 | 94.5 |
| √ | | | 79.2 | 83.4 | 88.2 | 92.4 | 95.6 | 97.6 |
| | √ | | 80.3 | 82.4 | 86.0 | 91.2 | 94.5 | 97.0 |
| | √ | √ | 79.1 | 83.3 | 83.7 | 90.8 | 93.8 | 96.3 |
| √ | √ | | 82.4 | 86.7 | 87.8 | **94.7** | **95.6** | 97.1 |
| √ | √ | √ | **84.5** | **88.3** | **90.0** | 94.6 | 95.4 | **97.9** |

A qualitative analysis from the perspective of the feature space was conducted for the memory module. As shown in Figure 6, by visualizing the feature maps of the student network, it can be observed that the introduction of the memory module helps to increase the feature distance between anomalies, enabling better anomaly responses.



(a) (b) (c)

**Figure 6.** Test results comparison. (a) Original image and mask. (b) Memory module is not configured. (c) Configure the memory module.

Additionally, an ablation study was performed on the data augmentation aspect

for different transformation versions (when $k = 2$, T stands for shift, and R stands for rotation). Table 5 shows the macro average AUC values under various data augmentation methods. The best performing methods are shown in bold.

Table 5. Ablation studies of different transformation.

| Data | No DCN | T | R | Scale | Shear | R + scale | T + scale | T + R | T + R + scale | Affine |
|---|---|---|---|---|---|---|---|---|---|---|
| MVTec | 82.8 | 83.2 | 85.5 | 84.2 | 83.8 | **85.9** | 84.8 | 84.0 | 84.3 | 84.6 |

Additionally, under unified conditions (MVTec dataset, 256 × 256 resolution, RTX 4070 Ti GPU, WideResNet50 backbone), As is shown in Figure 7, BKD achieves the highest F1-Score (87.5%) with 21.3 M parameters and 42.8 ms latency, DiffNet maintains efficiency with 15.4 M parameters and 32.1 ms, while US performs the worst (66.4%), and MKD sits in between with 13.5 M parameters and 40.5 ms. In industrial inspection scenarios, BKD significantly improves detection performance at an acceptable inference time cost, making it suitable for applications sensitive to missed detection rates.

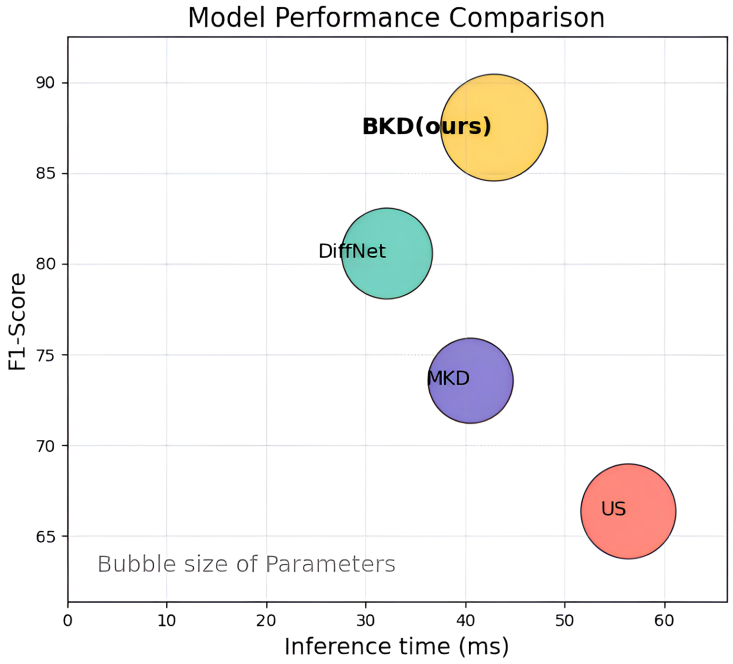



Figure 7. The structure of OCBE.

## 4. Conclusion

This study focuses on industrial visual anomaly detection, using anomaly image datasets as the research object. Based on the theory of unsupervised knowledge distillation, it makes improvements from two aspects: network structure and memory information, aiming to enhance the performance and efficiency of anomaly detection. By optimizing the network architecture design and establishing relevant theoretical models, a bidirectional distillation network architecture is pro-

posed. This algorithm reduces misjudgments caused by pseudo anomalies and alleviates the "normalcy forgetting" problem. Extensive experiments on public datasets have verified that it has significantly improved in terms of both accuracy and speed, providing new ideas and references for industrial anomaly detection. However, the study has limitations. For example, the adaptability of the algorithm to multiple scenarios in actual industrial settings has not been fully verified. Future research will focus on exploring more challenging multi-scenario anomaly detection algorithms to validate and optimize the method proposed in this paper further and unlock its potential in a wider range of industrial applications.

## Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

## References

[1] Tao, X., Gong, X., Zhang, X., Yan, S. and Adak, C. (2022) Deep Learning for Unsupervised Anomaly Localization in Industrial Images: A Survey. *IEEE Transactions on Instrumentation and Measurement*, **71**, 1-21. https://doi.org/10.1109/tim.2022.3196436

[2] Chalapathy, R., Khoa, N.L.D. and Chawla, S. (2020). Robust Deep Learning Methods for Anomaly Detection. *Proceedings of the* 26*th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, California, 6-10 July 2020, 3507-3508. https://doi.org/10.1145/3394486.3406704

[3] Erfani, S.M., Rajasegarar, S., Karunasekera, S. and Leckie, C. (2016) High-Dimensional and Large-Scale Anomaly Detection Using a Linear One-Class SVM with Deep Learning. *Pattern Recognition*, **58**, 121-134. https://doi.org/10.1016/j.patcog.2016.03.028

[4] Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S.A., Binder, A. and Kloft, M. (2018) Deep One-Class Classification. 2018 *International Conference on Machine Learning*, Stockholm, 10-15 July 2018, 4393-4402.

[5] Zhou, Y., Liang, X., Zhang, W., Zhang, L. and Song, X. (2021) Vae-Based Deep SVDD for Anomaly Detection. *Neurocomputing*, **453**, 131-140. https://doi.org/10.1016/j.neucom.2021.04.089

[6] Yi, J. and Yoon, S. (2021) Patch SVDD: Patch-Level SVDD for Anomaly Detection and Segmentation. In: *Lecture Notes in Computer Science*, Springer, 375-390. https://doi.org/10.1007/978-3-030-69544-6_23

[7] Defard, T., Setkov, A., Loesch, A. and Audigier, R. (2021) Padim: A Patch Distribution Modeling Framework for Anomaly Detection and Localization. In: *Lecture Notes in Computer Science*, Springer, 475-489. https://doi.org/10.1007/978-3-030-68799-1_35

[8] Cohen, N. and Hoshen, Y. (2020) Sub-Image Anomaly Detection with Deep Pyramid Correspondences. https://arxiv.org/pdf/2005.02357

[9] Chebotar, Y. and Waters, A. (2016) Distilling Knowledge from Ensembles of Neural Networks for Speech Recognition. *Inter Speech* 2016, San Francisco, 8-12 September 2016, 3439-3443. https://doi.org/10.21437/interspeech.2016-1190

[10] Chen, P., Liu, S., Zhao, H. and Jia, J. (2021) Distilling Knowledge via Knowledge Review. 2021 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*

(*CVPR*), Nashville, 20-25 June 2021, 5006-5015. https://doi.org/10.1109/cvpr46437.2021.00497

[11] Bergmann, P., Fauser, M., Sattlegger, D. and Steger, C. (2020) Uninformed Students: Student-Teacher Anomaly Detection with Discriminative Latent Embeddings. 2020 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Seattle, 13-19 June 2020, 4182-4191. https://doi.org/10.1109/cvpr42600.2020.00424

[12] Salehi, M., Sadjadi, N., Baselizadeh, S., Rohban, M.H. and Rabiee, H.R. (2021) Multi-resolution Knowledge Distillation for Anomaly Detection. 2021 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Nashville, 20-25 June 2021, 14897-14907. https://doi.org/10.1109/cvpr46437.2021.01466

[13] Yamada, S. and Hotta, K. (2021) Reconstruction Student with Attention for Student-Teacher Pyramid Matching. https://arxiv.org/pdf/2111.15376

[14] Deng, H. and Li, X. (2022) Anomaly Detection via Reverse Distillation from One-Class Embedding. 2022 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), New Orleans, 18-24 June 2022, 9727-9736. https://doi.org/10.1109/cvpr52688.2022.00951

[15] Gu, Z., Liu, L., Chen, X., Yi, R., Zhang, J., Wang, Y., *et al.* (2023) Remembering Normality: Memory-Guided Knowledge Distillation for Unsupervised Anomaly Detection. 2023 *IEEE/CVF International Conference on Computer Vision* (*ICCV*), Paris, 1-6 October 2023, 16355-16363. https://doi.org/10.1109/iccv51070.2023.01503

[16] Liu, X., Wang, J., Leng, B. and Zhang, S. (2024) Dual-Modeling Decouple Distillation for Unsupervised Anomaly Detection. *Proceedings of the* 32*nd ACM International Conference on Multimedia*, Melbourne, 28 October-1 November 2024, 5035-5044. https://doi.org/10.1145/3664647.3681669

[17] Cao, W. and Chen, X. (2019) Deformable Convolutional Networks Tracker. *DEStech Transactions on Computer Science and Engineering*, 2475-8841. https://doi.org/10.12783/dtcse/iteee2019/28747

[18] Zhu, X., Hu, H., Lin, S. and Dai, J. (2019) Deformable ConvNets V2: More Deformable, Better Results. 2019 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Long Beach, 15-20 June 2019, 9300-9308. https://doi.org/10.1109/cvpr.2019.00953

[19] Batagelj, V., Doreian, P., *et al.* (2014) Understanding Large Temporal Networks and Spatial Networks. Wiley.

[20] Bergmann, P., Fauser, M., Sattlegger, D. and Steger, C. (2019) Mvtec AD—A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. 2019 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Long Beach, 15-20 June 2019, 9584-9592. https://doi.org/10.1109/cvpr.2019.00982

[21] Jezek, S., Jonak, M., Burget, R., Dvorak, P. and Skotak, M. (2021) Deep Learning-Based Defect Detection of Metal Parts: Evaluating Current Methods in Complex Conditions. 2021 13*th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops* (*ICUMT*), Brno, 25-27 October 2021, 66-71. https://doi.org/10.1109/icumt54235.2021.9631567

[22] Rudolph, M., Wandt, B. and Rosenhahn, B. (2021) Same but Different: Semi-Supervised Defect Detection with Normalizing Flows. 2021 *IEEE Winter Conference on Applications of Computer Vision* (*WACV*), Waikoloa, 3-8 January 2021, 1906-1915. https://doi.org/10.1109/wacv48630.2021.00195