Neural Computing and Applications (2024) 36:14543–14557
https://doi.org/10.1007/s00521-024-09838-0

ORIGINAL ARTICLE

MCAD: Multi-classification anomaly detection with relational knowledge distillation

Zhuo Li[1] · Yifei Ge[1] · Xuebin Yue[2] · Lin Meng[3]

Received: 10 October 2023 / Accepted: 12 April 2024 / Published online: 8 May 2024

Abstract

With the wide application of deep learning in anomaly detection (AD), industrial vision AD has achieved remarkable success. However, current AD usually focuses on anomaly localization and rarely investigates anomaly classification. Furthermore, anomaly classification is currently requested for quality management and anomaly reason analysis. Therefore, it is essential to classify anomalies while improving the accuracy of AD. This paper designs a novel multi-classification AD (MCAD) framework to achieve high-accuracy AD with an anomaly classification function. In detail, the proposal model based on relational knowledge distillation consists of two components. The first one employs a teacher–student AD model, utilizing a relational knowledge distillation approach to transfer the interrelationships of images. The teacher–student critical layer feature activation values are used in the knowledge transfer process to achieve anomaly detection. The second component realizes anomaly multi-classification using the lightweight convolutional neural network. Our proposal has achieved 98.95, 96.04, and 92.94% AUROC AD results on MNIST, FashionMNIST, and CIFAR10 datasets. Meanwhile, we earn 97.58 and 98.10% AUROC for AD and localization in the MVTecAD dataset. The average classification accuracy of anomaly classification has reached 76.37% in fifteen categories of the MVTec-AD dataset. In particular, the classification accuracy of the leather category has gained 95.24%. The results on the MVTec-AD dataset show that MCAD achieves excellent detection, localization, and classification results.

Keywords Deep learning · Computer vision · Anomaly detection · Anomaly classification · Knowledge distillation

1 Introduction

With the advancement of deep learning, the application of deep learning for anomaly detection (AD) has been widely used in various industries. Industrial anomaly detection applying deep learning is crucial for improving product quality and analyzing products. However, deep learning-based industrial anomaly detection requires a large number of images. Normal images are easy to obtain, and anomalous images are both rare and diverse. This is a challenge for applying deep learning in industrial anomaly detection. To address this issue, unsupervised deep learning is applied to industrial anomaly detection [1, 2]. Unsupervised learning efficiently recognizes unlabeled anomaly images. Hence, unsupervised learning based on deep learning has been widely used in anomaly detection, including industrial anomaly detection [3–6], network anomaly detection [7–10] and hyperspectral anomaly detection [11, 12].

✉ Lin Meng
menglin@fc.ritsumei.ac.jp

Zhuo Li
gr0607si@ed.ritsumei.ac.jp

Yifei Ge
gr0607pe@ed.ritsumei.ac.jp

Xuebin Yue
yue-xb@fc.ritsumei.ac.jp

[1] Graduate School of Science and Engineering, Ritsumeikan University, Shiga, Japan

[2] Research Organization of Science and Technology, Ritsumeikan University, Shiga, Japan

[3] College of Science and Engineering, Ritsumeikan University, Shiga, Japan

Unsupervised learning-based anomaly detection includes embedding-based and reconstruction-based [13]. The reconstruction-based method relies on analyzing variations between the initial and reconstructed data for anomaly identification. First, a model is constructed using normal data for training to learn the characteristics of normal data. Then the anomaly data is fed into this model for reconstruction, and the degree of abnormality is determined by comparing the difference between the original and reconstructed data. The data is flagged as an anomaly if the reconstruction error exceeds the threshold value. Common methods include autoencoder [14], generative adversarial networks (GANs) [15–18], transformer [19–23] and diffusion [24–27]. Unlike the reconstruction-based approach, the embedding-based method is to generate a low-dimensional space and maps the data into the space. The significant idea of the embedding-based anomaly detection method is to determine anomalies based on the position or density in the embedding space. Normal data are clustered in the space, and anomalous data are separated from normal data. The common embedding-based approaches are one-class classification (OCC) [28–31], distribution map [32–37], memory bank [38, 39] and teacher–student model [40–46].

The teacher–student model is well-interpretable and generalizable that it has become a representative approach for industrial anomaly detection. Furthermore, the teacher–student model is an effective model compression method for classification tasks [47]. The common teacher–student model uses individual knowledge distillation for knowledge transfer. This approach of knowledge transfer only transfers the last layer of knowledge from the teacher's model to the student model, which may not adequately facilitate the acquisition of the teacher model's structural knowledge by the student model. Park et al. [48] propose relational knowledge distillation (RKD), which transfers structural knowledge from the teacher model to the student model. Through RKD, student models are able to gain more comprehensive and enriched knowledge. Consequently, student and teacher models with different structures performed better in the RKD. Meanwhile, the anomaly detection performance of teacher–student models with different structures has been shown to outperform teacher–student models with the same structure in multiresolution knowledge distillation (MKD) [42] and asymmetric student–teacher (AST) [6]. In addition, AST suggests that teacher–student models of the same structure extract significantly similar anomalous image features. Similar anomaly features are a challenge for anomaly detection and anomaly classification. Therefore, it is better to design different teacher–student structures than the same for anomaly detection and classification tasks.

The difficulty in anomaly classification lies in the rarity and diversity of anomaly images. First, the rarity of anomaly detection refers to the low percentage of anomalous images in the overall dataset. The data imbalance leads to image bias during training, which makes the model more likely to classify images as normal. The model cannot learn the anomaly images, which affects the accuracy of anomaly classification during the training process. Second, the diversity of anomalous images refers to the variability of anomalous images in terms of characteristics. Anomalous images are represented as various shapes, colors, et al. There is no apparent common feature among them. Most of the current work on anomaly detection is based on OCC, which classifies images as normal and anomalous.

This paper designs a multi-classification anomaly detection (MCAD) framework to realize anomaly classification and anomaly detection. MCAD utilizes a teacher–student model with different structures for anomaly detection. The ResNet18 model serves as the teacher, and the ResNet10 model serves as the student. In the training process, the teacher–student model learns normal image information. To make the student model comparable to the teacher model, the teacher model imparts knowledge to the student model through RKD. During the testing process, the teacher and student models responded differently to the input images since the teacher model used pre-trained parameters on ImageNet. This difference is converted into feature activation values for anomaly detection and anomaly localization. In addition, MCAD employs a multi-classification model for anomaly classification. By transfer learning, the multi-classification model is equipped with pre-trained weights to categorize a limited set of abnormal images. The primary contributions of this study are outlined as follows.

- This paper presents a multi-classification anomaly detection framework, which contains two stages. The first stage performs anomaly detection and anomaly localization through a teacher–student model. The second stage implements anomaly classification through a lightweight model.
- This paper designs a teacher–student model with different structures. Efficient knowledge transfer between teacher–student models through relational knowledge distillation.
- This paper proposes a multi-classification model for anomaly classification. Consider anomaly classification as a traditional image classification task. The intermediate features of the student model are fused as inputs to the classification model through feature fusion.
- Extensive experiments on an industrial anomaly detection dataset are conducted to validate the performance of the method in this paper on anomaly detection and

anomaly classification. Compared with other methods, the method in this paper shows excellent performance.

The structure of the paper for the rest of the content is as follows. Section 2 provides an extensive overview of the prior research related to anomaly detection and anomaly classification. Section 3 describes in detail the MCAD framework proposed in this paper. Section 4 presents datasets, results, ablation study and discussions. Section 5 summarizes the findings of this paper and explores prospects for future research.

2 Related work

2.1 Anomaly detection

The teacher–student model transfers the feature extraction capability for normal data to the student model through the teacher model. During the inference process, anomaly detection is performed using the teacher and student models for feature differences of anomalous images. The utilization of the teacher–student structure for AD was initially proposed by Bergmann et al. [40]. After that, STPM [41] and MKD [42] use different approaches to refine multi-scale features under different network layers. However, the typical image features extracted from the student model features are more similar to those extracted from the teacher model, while the anomaly image features are less similar. Compared to STPM, RSTPM [43, 44] adds a pair of teacher–student models. During the test process, the new teacher model is placed behind the original teacher–student model and is responsible for replicating the features. The student model reconstructs normal features at the appearance of abnormal images and enables the reconstructed features to be distinguished from those of the teacher model. In contrast to RSTPM, RD4AD [45] uses only a pair of teacher–student models but has the same similarities in learning as RSTPM. RD4AD introduces multi-scale feature fusion blocks and one-class bottlenecks to create embeddings that eliminate superfluous features across various scales. This configuration of teacher–student models facilitates proficient feature reconstruction. During the inference process, the anomalous image features extracted by the RD4AD teacher-pupil model varied greatly. Therefore, this paper proposes a differently structured teacher–student model while utilizing a new knowledge distillation approach for knowledge transfer. Efficient knowledge transfer between teacher–student models with different structures improves the performance of model anomaly detection.

2.2 Anomaly classification

Anomaly detection is usually learned only on normal images and is viewed as an OCC problem. Support vector domain description (SVDD) stands as a classical algorithm employed to address the OCC problem. Based on SVDD, researchers have improved SVDD in industrial anomaly detection and propose discriminant SVDD (DSVDD) [31], patch SVDD (PatchSVDD) [28], deep structure preservation SVDD(DSPSVDD) [29] and semantic-enhanced SVDD (SESVDD) [30]. The main of SVDD is to project the images into the feature space and compute the centroids of all sample projections at a specified radius r. Those within r from the centroids are considered normal images; otherwise, they are abnormal. Ruff et al. [31] present DSVDD, a method utilizing DNNs for the mapping process. However, both SVDD and DSVDD have the disadvantage that the whole image is processed during the process, which means that each image corresponds to a point in the feature space during the projection process. The result of this is that it identifies anomalies but does not locate anomalies. Yi et al. [28] propose PatchSVDD to turn the processed objects from the whole image into patches; each patch corresponds to a point in the feature space. In addition, since the features of different patches are different and the patches of normal images may be very far apart in the feature space over time, it is not feasible to use only one centroid. PatchSVDD replaces one centroid of SVDD with multiple centroids formed by clustering. Zhang et al. [29] propose DSPSVDD by designing an improved integrated optimization objective for the DSVDD module, considering both hypersphere volume minimization and network reconstruction error minimization, to extract depth data features more efficiently.

Compared to OCC, the contrastive language-image pretraining (CLIP)-based approach shows excellent performance on the anomaly classification task. Jeong et al. [49] propose a window-based CLIP model, WinCLIP for anomaly classification. With the prompting of the large language model, WinCLIP achieves excellent classification results on the MVTec-AD dataset. Liu et al. [50] design a lightweight and nearly training-free unsupervised semantic segmentation model. The input image is segmented into multiple parts to realize industrial vision inspection. However, the article only mentions good potential for anomaly classification and does not show results. Although anomaly classification has achieved excellent results, anomaly classification relies heavily on large language models. This increases the computational complexity of the model.

The approaches mentioned above have their advantages, but there is still a lot of room for improvement. The current

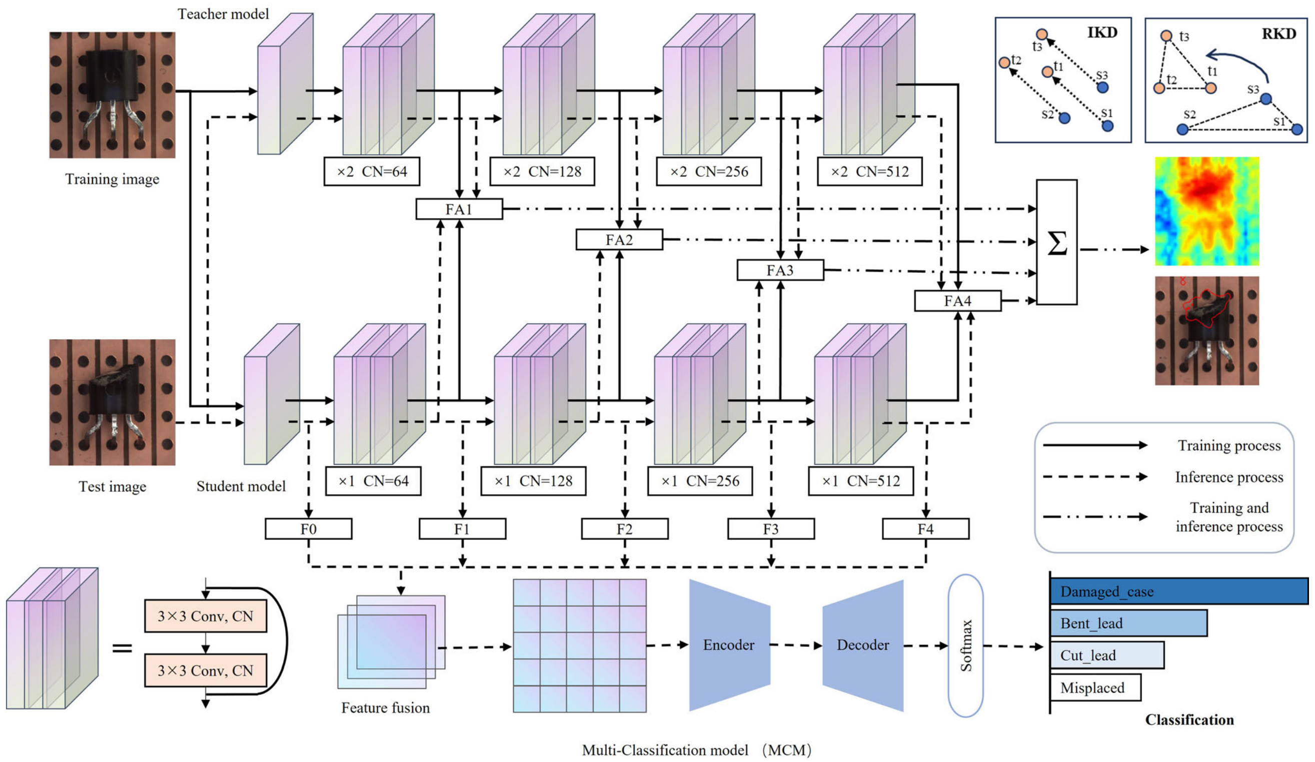

Fig. 1 Overview of multi-classification anomaly detection and localization with relational knowledge distillation framework. *IKD*: Individual knowledge distillation. *RKD*: Relational knowledge distillation. A smaller student model (*S*) is trained to mimic the teacher model (*T*). A lightweight multi-classification model (*MCM*) is used for anomaly classification. *T*: A ResNet18 model pre-trained on ImageNet. *S*: The ResNet10 model without pre-trained. A total loss function for anomaly detection and localization calculates teacher–student intermediate differences. *FA* represents the feature activation value. *F* represents the features of the intermediate critical layer. *MCM*: Simple classification model. The intermediate features of the *S* are used as the input for anomaly classification. × *N* represents the number of times used. *CN* is the number of channels

anomaly detection and anomaly classification models are complex and even require large language models, which increases the complexity of the system and reduces its usefulness. To address these issues, based on some studies [6, 42, 48], this paper proposes MCAD, which is described in Sect. 3.

3 Method

As mentioned above, this paper proposes the MCAD framework for anomaly detection, anomaly localization and anomaly classification. This section discusses the MCAD framework in detail. First, anomaly detection and localization are introduced. It employs a teacher–student model, inspired by RKD [48], and introduces a relational distillation strategy for knowledge transfer. Then, the lightweight classification model proposed in this paper is demonstrated for anomaly classification. MCAD framework is shown in Fig. 1. This section describes the teacher–student model. Meanwhile, the anomaly classification model is introduced in detail. Algorithm 1 shows the process of MCAD anomaly detection and classification.

Algorithm 1 MCAD pseudo-code

```
Input: Normal train data $\mathbb{D}_{train}$, anomaly test data $\mathbb{D}_{test}$, batch size $\mathcal{B}$.
Output: Anomaly localization and classification results.
procedure MCAD ANOMALY DETECTION AND CLASSIFICATION
    Initialize Teacher Network $\mathbf{T}$ with pretrained ImageNet weights.
    Initialize Student Network $\mathbf{S}$ with random weights.
    Phase 1: Training
    for each batch $\mathcal{B}$ in $\mathbb{D}_{train}$ do
        Extract features $F_T$ using $\mathbf{T}$ on $\mathcal{B}$
        Train $\mathbf{S}$ to mimic $F_T$ using $\mathcal{B}$
    end for
    Phase 2: Anomaly detection
    for each image $x \in \mathbb{D}_{test}$ do
        Extract layer-wise features $F_T^{layers}$ using $\mathbf{T}$ on $x$
        Extract layer-wise features $F_S^{layers}$ using $\mathbf{S}$ on $x$
        Compute anomaly score via summing and comparing differences in $F_T^{layers}$ and $F_S^{layers}$
    end for
    Phase 3: Anomaly classification
    for each sample $x \in \mathbb{D}_{test}$ do
        Extract and fuse intermediate features from $\mathbf{S}$
        Feed fused features to $\mathbf{MCM}$ for classification
    end for
    return Anomaly localization and classification results for $\mathbb{D}_{test}$
end procedure
```

3.1 Teacher-student model

The teacher–student model only needs to be trained with normal data during anomaly detection, which addresses the data labeling problem in the anomaly detection task. Meanwhile, the teacher–student model adapts to new images by learning the distributional characteristics of normal data. The teacher–student model also learns and captures the complex patterns of normal data, including the interrelationships and dependencies between the data. In summary, the teacher–student model enables to detection of the distinction between normal and anomaly images during the process of anomaly detection. Compared with the traditional teacher–student model, this paper designs teacher and student models with different structures. As described in MKD [42] and AST [6], the different structures have achieved better performance on the anomaly detection than the teacher–student models with the same network. Moreover, the method of RKD is employed to enhance the accuracy of anomaly detection by transferring the structural information from the teacher to the student.

ResNet18 and ResNet10 models possess relatively shallow network architectures, allowing them to perform effectively even on devices with limited computational resources. As shown in Fig. 1, the teacher model is a ResNet18, loaded with ImageNet pre-training weights, while the student model is an initialized ResNet10. In the training process, normal images are used as inputs for the teacher and student models, respectively. The teacher model imparts the student model's feature extraction ability through RKD. In the upper right corner of Fig. 1, the difference between RKD and individual knowledge distillation is demonstrated. Compared with individual knowledge distillation, RKD better conveys the structural information of teacher models. Conventional knowledge distillation passes the output of the teacher as prior knowledge to the student without focusing on the relationships between feature layers, resulting in poor interpretability of the student model. This paper employs an RKD approach for knowledge transfer to enhance the student model's understanding of the structural information between the feature layers of the teacher model. RKD combines multiple outputs into structural units by using the outputs of multiple teacher models as structural units. This is better able to reflect the structured characteristics of the teacher and enables better teaching of the student model. Unlike individual distillation learning, the loss function of RKD also has a function for constructing structural information, making the student model learn the teacher model's more efficient information representation capability. The loss function in anomaly detection and localization is described as follows.

Distance distillation loss The i_{th} critical layers of the model are represented as CL_i. The feature activation (FA) values for the i-th critical layers of the teacher and student are denoted as $\text{FA}_T^{\text{CL}_\text{i}}$ and $\text{FA}_S^{\text{CL}_\text{i}}$, respectively. Based on this value, distances and angles are introduced to enhance the knowledge transfer from teacher to student. Therefore, this paper defines two loss values, $\mathcal{L}_{\text{Dis}}$ and $\mathcal{L}_{\text{Ang}}$, which represent the loss of the distance term and the angle term, respectively. $\mathcal{L}_{\text{Dis}}$ is mainly concerned with minimizing the Euclidean distance between the teacher and student feature activation values at CL_i. Therefore, $\mathcal{L}_{\text{Dis}}$ is defined as:

$$\mathcal{L}_{\text{Dis}} = \sum_{(x_i,x_j)\in\mathcal{X}^2} l_\delta\big(\psi_\mathcal{D}(t_i,t_j),\psi_\mathcal{D}(s_i,s_j)\big), \tag{1}$$

where $\mathcal{X}^2$ represents a set of 2-tuples of different data instances, $\mathcal{X}^2 = \{(x_i,y_j)|i \neq j\}$. $t_i = \text{FA}_T^{\text{CL}_\text{i}}$ and $s_i = \text{FA}_S^{\text{CL}_\text{i}}$. $l_\delta(a,b)$ is the Huber loss, which is defined as:

$$l_\delta(a,b) = \begin{cases} \frac{1}{2}(a-b)^2, & |a-b| \leq 1, \\ |a-b| - \frac{1}{2}, & \text{otherwise.} \end{cases}, \tag{2}$$

$\psi_\mathcal{D}(t_i,t_j)$ is the calculation of the Euclidean distance of two instances in the output representation space:

$$\psi_{\text{D}}(t_i,t_j) = \frac{1}{\mu}\left\|t_i - t_j\right\|_2, \tag{3}$$

where μ is the distance normalization factor. To consider the distance between multiple pairs, μ is established as the mean distance between pairs of $\mathcal{X}^2$ within the mini-batch:

$$\mu = \frac{1}{|\mathcal{X}^2|}\sum_{(x_i,x_j)\in\mathcal{X}^2}\left\|t_i - t_j\right\|_2. \tag{4}$$

Mini-batch distance normalization proves beneficial in aligning the distances between teacher and student embeddings, especially in the presence of significant scale differences between teacher distances $\left\|t_i - t_j\right\|_2$ and student distances $\left\|s_i - s_j\right\|_2$. It is experimentally demonstrated that normalization makes the model more stable and converges faster during training. The distance distillation loss conveys image relationships by evaluating the disparity in distances within their respective output representation spaces. Unlike traditional knowledge distillation, distance distillation focuses on the distance structure of the teacher's output rather than directly matching the teacher's output.

Angle distillation loss The angle distillation loss conveys the relationships among training instance embeddings by assessing the variance in angles. Compared to distance, angles are more intricate and impart relational information more effectively, thereby affording students greater flexibility in model training. $\mathcal{L}_{\text{Ang}}$ is defined as:

$$\mathcal{L}_{\text{Ang}} = \sum_{(x_i,x_j,x_k)\in\mathcal{X}^3} l_\delta\big(\psi_{\mathbf{A}}(t_i,t_j,t_k),\psi_\mathcal{A}(s_i,s_j,s_k)\big), \tag{5}$$

where $\mathcal{X}^3 = \{(x_i,y_j,x_k)|i \neq j \neq k\}$, $l_\delta(x,y)$ is the Huber loss. In addition, the angle similarity between activation vectors is increased using $\mathcal{L}_{\text{Ang}}$, which is even more critical in the ReLU network because the neurons in a ReLU network are only activated when surpassing a threshold of zero value. This implies that two activation vectors exert contrasting influences on the activation of subsequent neurons, despite having an equal Euclidean distance from the object vector. To solve this problem, employ the cosine similarity metric. Equation (6) shows the relational potential on an angle measuring the angles representing the angle formed by the three examples in space.

$$\psi_A(t_i,t_j,t_k) = \cos\angle t_i t_j t_k = \langle \mathbf{e}^{ij}, \mathbf{e}^{kj}\rangle, \tag{6}$$

where

$$\mathbf{e}^{ij} = \frac{t_i - t_j}{\left\|t_i - t_j\right\|_2}, \mathbf{e}^{kj} = \frac{t_k - t_j}{\left\|t_k - t_j\right\|_2}. \tag{7}$$

The RKD is designed to transfer structural knowledge using the interrelationship of data instances output by the instructor. RKD is defined as follows:

$$\mathcal{L}_{\text{RKD}} = \mathcal{L}_{\text{Dis}} + \lambda \cdot \mathcal{L}_{\text{Ang}}, \tag{8}$$

where λ is the hyperparameter for the $\mathcal{L}_{\text{Dis}}$ and $\mathcal{L}_{\text{Ang}}$. The impact of λ of anomaly detection result is discussed in Sect. 4.4.

3.2 Anomaly detection

Given a dataset $\mathbb{D}_{\text{train}} = \{I_1, I_2, \ldots, I_N\}$ for training, where $\mathbb{D}_{\text{train}}$ contains only normal images. A student model is trained to identify anomalous images within the test dataset $\mathbb{D}_{\text{test}}$. The teacher only transfers the knowledge of normal images to the student model in the knowledge distillation process, as shown in Fig. 1. Therefore, the feature activation values of anomalous images are not in the range of normal images in the inference process. Feature activation values for anomalous images are novel values for the student model. Because the teacher model is a pre-trained model on ImageNet, there are differences between the teacher and the student for anomalous images. The threshold value of this difference is used for anomaly detection as in Eq. (8). In Fig. 1, FA1 represents the result of feature activation values for the first critical layer of the teacher–student model. MCAD utilizes the teacher–student model with different intermediate feature activation values for anomaly detection. The effect of different feature activation values on the anomaly detection results is discussed in Sect. 4.4.

3.3 Anomaly localization

As shown in Fig. 1, each test image is fed into the teacher–student model to locate anomalous areas. MKD [42] has demonstrated that the derivative of the loss function with respect to the input yields valuable insights into the significance of individual pixels. This paper obtains the gradient information of each input dimension by calculating the derivative of the loss L_{RKD} concerning the input x_i. The gradient information value signifies the extent of the impact of this dimension on optimization and the presence of anomalies in the loss. Therefore, the gradient of L_{RKD} is used to find the anomalous areas that cause its value to increase. However, to obtain the localization feature map of input x_i, the attribution map of x_i is first obtained. Equation (9) shows how to calculate the attribution map $\mathcal{A}$:

$$\mathcal{A} = \frac{\partial L_{\mathrm{RKD}}}{\partial x_i}. \tag{9}$$

The attribution map $\mathcal{A}$ is obtained by introducing Gaussian blur and open morphological filters. The attribution map $\mathcal{A}$ reduces the natural noise in the feature maps. Thus, the localization feature map L_{map} is represented as:

$$\mathbb{M} = \mathcal{G}_\sigma(\mathcal{A}), \tag{10}$$

$$L_{\mathrm{map}} = (\mathbb{M} \ominus \mathcal{B}) \oplus \mathcal{B}, \tag{11}$$

where $\mathcal{G}$ is a Gaussian filter with a standard deviation σ. $\ominus$ and $\oplus$ denote the morphological operators of shape erosion and dilation through the structural element $\mathcal{B}$, respectively. These operations are known as morphology-open. $\ominus$ removes the light-colored noise. $\oplus$ increases the light-colored component. $\mathcal{B}$ is a simple binary mapping, usually oval or circular disk-shaped. Compared to Eq. (9), Eq. (11) not only enhances the clarity of each pixel's impact on the loss value but also enables more accurate identification of anomalous areas.

3.4 Anomaly classification

In the method of classification using deep learning, customary approaches involve the design of DNNs to learn image representation and characterization. The residual structure is proposed to give ideas for designing DL networks. As a result of the skip-connect within the residual structure, the input from the preceding layer is directly added to the output of the subsequent layer. Meanwhile, the residual structure improves information to pass more freely in the network while retaining the original input information, thus improving the expressiveness of the network. Therefore, this paper considers this in designing the anomaly classification model, and a multi-classification model with residual structure is designed. The multi-classification model that we designed is shown in Fig. 2, which corresponds to the encoder–decoder part in Fig. 1. Since the anomaly images are only in the inference process, the multi-classification model is trained by transfer learning.

At the bottom of Fig. 1, multiple intermediate features of the student model are fused and used as inputs to the multi-classification model for anomaly classification. Feature fusion is done by feature concatenation. Feature concatenation enables multiple intermediate features to be concatenated together to form a large feature vector, thus extending the feature representation capability while preserving the information of each intermediate feature. Moreover, feature concatenation fuses different levels of features while also fusing features at different scales. This captures richer feature information and improves the model's classification ability. The student model's five intermediate features (F0–F4) are feature fused in Fig. 1. However, in the process of experimentation, different features are selected for feature fusion. For example, only F0 is used as a feature for anomaly classification. Similarly, feature fusion is performed using F0 + F1. The impact of using features from different intermediate critical layers fused as input to a multi-classification model on the classification model on classification results is explored in Sect. 4.3. In addition, Table 4 provides the classification results after the fusion of different critical layer features.

4 Experiments

This section proves the effectiveness of the proposed method in this paper by extensive experiments. First, the experimental environment and datasets are introduced. Subsequently, we present the outcomes of the MCAD framework proposed in this paper across various datasets. After that, an ablation study is conducted. Finally, the whole experiment is discussed.

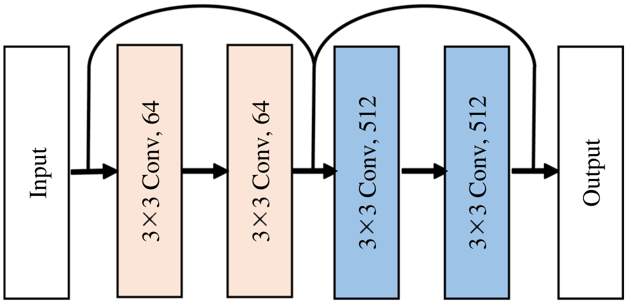

Fig. 2 Multi-classification model

4.1 Experiment setup

Environment The experiments are executed using Python 3.9 and PyTorch 1.13.1, utilizing an Intel(R) Core i7-13700KF CPU and an NVIDIA GeForce RTX 4090 GPU. For MNIST, FashionMNIST and CIFAR10 experiments, the original image size is used as input. For the MVTec-AD dataset [51, 52], resize all images to 224 × 224.

Datasets MNIST: The MNIST dataset comprises 60,000 training images and 10,000 test images, with each image being a grayscale handwritten digit of dimensions 28 × 28 pixels. *FashionMNIST*: FashionMNIST serves as an image dataset that substitutes MNIST, containing a total of 70,000 images showcasing various products across 10 distinct categories. *CIFAR10*: CIFAR10 contains 60,000 color images in 10 categories, with the size of 32 × 32. *MVTec-AD*: MVTec-AD dataset is proposed by MVTec for industrial anomaly detection. The training set of MVTec-AD exclusively comprises normal images. The test set includes normal and anomaly images. In the anomaly images, the different classes of abnormalities are classified. The dataset mimics scenarios from industrial production, encompassing a variety of domains with five textures and ten objects. It also provides pixel-level annotations for anomaly regions and is a comprehensive multi-object, multi-anomaly dataset. MVTec-AD is a popular dataset in the field of AD.

4.2 Results of MNIST & FashionMNIST & CIFAR10

The proposed MCAD framework is contrasted against other state-of-the-art (SOTA) AD methods in this paper. The area under the receiver operating characteristic curve (AUROC) approach is used for outcome evaluation. Since the MNIST, FashionMNIST, and CIFAR10 datasets contain ten categories, this paper uses Class 0–9 to represent the ten categories. Table 1 compares the method described in this paper with other approaches. ARAE [53] improves anomaly detection by adversarially training autoencoders. LSA [54] introduces latent space autoregression as a method for AD. The latent space refers to the lower-dimensional representation created by the encoder to encapsulate the fundamental attributes of the input data. Anomaly detection is achieved by modeling the correlation between data points in latent space. MKD [42] employs multiresolution knowledge distillation to address AD. Anomaly detection performance is improved by knowledge transfer between teacher–student models and incorporates representations at different scales to better capture anomalous behavior in the data. The approach in this paper uses a teacher–student model for anomaly detection. Knowledge transfer between the teacher and the student is performed by relational knowledge as a manner to improve AD. In the context of the MNIST dataset, the method in this paper achieved several best results and ended up with the best average result of 98.95%. Compared with MKD and AREA, the method in this paper improved the detection accuracy by 0.24 and 1.45%, respectively. For the FashionMNIST dataset, MCAD achieved an average result of 96.04%. It is better than the detection of MKD. This is because some categories in FashionMNIST are more closely related, and MCAD does not perform well on multiscale problems. CIFAR10 is a commonly used dataset in computer vision. It is also utilized within the realm of AD. On the CIFAR10 dataset, MCAD achieved the best detection results in several categories. MCAD achieves the best average result of 92.24%, outperforming the MKD [42] by 5.06%.

4.3 Experimental results of MVTec-AD

This subsection discusses the anomaly detection and classification results for the MVTec-AD dataset. The various categories of the MVTec-AD dataset are divided into two parts, texture and object classes. Consequently, Table 2 presents the outcomes of anomaly detection, and Table 3 exhibits the results of anomaly localization across diverse categories. Meanwhile, the average results of the texture class and object class are also shown. The anomaly classification results are displayed in Table 4.

Anomaly detection This section showcases the outcomes of AD performed on the MVTec-AD dataset. Table 2 shows the comparative results between the approach presented in this paper and other methods. For the texture and object classes of MVTec-AD, MCAD achieved AUROC results of 98.83 and 96.12%. At the same time, MCAD achieved an average AUROC result of 97.58% for all classes. This is 11.32 and 8.61% higher than US [40] and MKD [42]. US [40] and MKD [42] also used the teacher–student model. US [40] utilizes the uninformative student model and discriminative embedding loss function for AD. It is noteworthy that US [40] utilizes the exact structure of the teacher–student model. MKD [42] achieved better detection results with different teacher–student structures than US [40]. However, MKD [42] employs a DNN with a structure VGG, where the teacher–student models respond similarly to the same anomalous region. The residual structure provides better extraction of anomalous areas. Therefore, MCAD employs ResNet18 as the teacher model, a deep neural network with a residual structure. Moreover, the student model is ResNet10 to ensure that the teacher–student model has a different structure. In summary, the MCAD framework proposed in this paper outperforms most models in AD, particularly achieving SOTA

Table 1 AUROC results for anomaly detection on MNIST, FashionMNIST and CIFAR10

Datasets	Method	Class0	Class1	Class2	Class3	Class4	Class5	Class6	Class7	Class8	Class9	Mean
MNIST	ARAE	**99.8**	**99.9**	96.0	97.25	97.0	97.40	**99.50**	96.9	92.40	98.50	97.50
	AnoGAN	96.6	99.2	85.0	88.7	89.4	88.3	94.7	93.5	84.9	92.4	91.3
	DSVDD	98.0	99.7	91.7	91.9	94.9	88.5	98.3	94.6	93.9	96.5	94.8
	LSA	99.3	99.9	95.9	96.6	95.6	96.4	99.4	98.0	95.3	98.1	97.5
	MKD	99.82	99.82	97.79	98.75	98.43	98.16	99.43	98.38	**98.41**	98.10	98.71
	MCAD (**Ours**)	98.89	98.91	**98.92**	**99.12**	**98.85**	**98.85**	98.92	**99.09**	98.83	**99.11**	**98.95**
FashionMNIST	ARAE	93.7	99.1	91.1	94.40	92.30	91.40	83.60	98.90	93.90	97.90	93.60
	DSVDD	**98.2**	90.3	90.7	94.2	89.4	91.80	83.40	98.8	91.90	**99.00**	92.80
	LSA	91.6	98.3	87.8	92.3	89.7	90.7	84.1	97.7	91.02	98.4	92.2
	DAGMM	30.3	31.1	47.5	48.10	49.90	41.30	42.00	37.40	51.80	37.80	41.70
	MKD	92.5	**99.21**	92.48	93.8	92.95	**98.21**	84.87	**99.02**	94.33	97.51	**97.49**
	MCAD (**Ours**)	96.34	96.37	**94.4**	**96.12**	**96.61**	96.35	**97.59**	95.97	**95.53**	95.15	96.04
CIFAR10	ARAE	72.2	43.1	69.0	55.00	75.20	54.70	70.10	51.00	72.20	40.00	60.23
	AnoGAN	67.10	54.70	52.90	54.50	65.10	60.30	58.50	62.50	75.80	66.50	61.79
	DSVDD	61.70	65.90	50.80	59.10	60.90	65.70	67.70	67.30	75.90	73.10	64.81
	LSA	73.50	58.00	69.00	54.20	76.10	54.60	75.10	53.50	71.10	54.80	64.10
	MKD	90.53	90.35	79.66	77.02	86.71	91.40	88.98	86.78	**91.45**	88.91	87.18
	MCAD (**Ours**)	**93.49**	**93.58**	**92.34**	**92.97**	**94.27**	**94.02**	**93.25**	**91.96**	90.96	**92.52**	**92.94**

The MNIST, FashionMNIST and CIFAR10 datasets all contain ten categories, and this paper uses Class 0–9 to represent the ten categories

Bold represents the best results

Table 2 Anomaly detection results with AUROC on MVTec-AD

Category	GeoTrans [55]	GANormaly [56]	DGAD [57]	OCG [58]	ITAE [59]	MKD [42]	US [40]	RIAD [60]	CFLOW-AD [35]	DRAEM [61]	NSA [62]	MCAD(Ours)
Carpet	43.7	69.9	52.0	98.9	70.6	79.25	69.5	84.20	98.20	97.00	95.60	**99.39**
Grid	61.9	70.8	67.0	99.6	88.3	78.01	81.9	99.60	98.97	**99.90**	**99.90**	98.83
Leather	84.1	84.2	94.0	97.1	86.2	95.05	81.9	**100.0**	**100.0**	100.0	99.90	**100.0**
Tile	41.7	79.4	83.0	92.2	73.5	91.57	91.2	98.70	98.40	99.60	**100.0**	97.61
Wood	61.1	83.4	72.0	95.8	92.3	94.29	72.5	93.00	98.59	**99.10**	97.50	98.60
Mean_tex	58.5	77.54	73.6	96.6	82.2	87.63	79.4	95.10	98.83	**99.12**	98.60	98.89
Bottle	74.4	89.2	97.0	99.6	94.1	99.39	91.8	99.90	**100.0**	99.20	97.70	**100.0**
Cable	78.3	75.7	90.0	99.2	83.2	89.19	86.5	81.90	97.62	91.80	94.50	**98.08**
Capsule	67.0	73.2	60.0	95.4	68.1	80.46	91.6	88.40	93.15	**98.50**	95.20	95.81
Hazelnut	35.9	78.5	80.0	88.2	85.5	98.37	93.7	83.30	99.91	**100.0**	94.70	99.93
Metal_nut	81.3	70.0	95.0	98.7	66.7	73.58	89.5	88.50	98.45	**98.70**	**98.70**	98.62
Pill	63.0	74.3	76.0	98.5	78.6	82.70	93.5	83.80	93.02	98.90	**99.20**	98.20
Screw	50.0	74.6	67.0	**100.0**	**100.0**	93.31	92.8	84.50	85.94	93.90	90.20	88.75
Toothbrush	97.2	65.3	93.0	98.2	**100.0**	92.17	86.3	**100.0**	99.86	**100.0**	**100.0**	98.89
Transistor	86.9	79.2	88.0	94.9	84.3	85.55	70.1	90.90	93.04	93.10	**95.10**	94.42
Zipper	82.0	74.5	82.0	97.6	87.6	93.24	93.3	98.10	96.15	**100.0**	99.80	96.53
Mean_obj	71.6	75.45	82.8	97.0	84.8	88.80	88.9	89.90	95.71	**97.41**	96.50	96.92
Mean	67.23	76.15	80.0	96.9	83.9	88.41	85.7	91.70	96.75	**97.98**	97.20	97.58

Bold represents the best results

results in categories such as carpet, leather, bottle, and cable.

Anomaly localization This part examines the outcomes of anomaly localization utilizing the MCAD approach. Table 3 presents the outcomes of anomaly localization on the MVTec-AD dataset in comparison with alternative methods. For the texture and object classes, the average result for the text category differs from the best result by

Table 3 Anomaly localization results with AUPRO on MVTec-AD

Category	AE-SSIM [63]	VEVAE [64]	AnoGAN [16]	CFLOW-AD [35]	US [40]	RIAD [60]	DRAEM [61]	NSA [62]	MCAD (Ours)
Carpet	87.0	78.0	54.0	99.29	93.5	96.30	95.50	95.50	**99.30**
Grid	94.0	73.0	58.0	98.53	89.9	98.80	**99.70**	99.20	98.55
Leather	78.0	95.0	64.0	**99.51**	97.8	99.40	98.60	99.50	99.50
Tile	59.0	92.5	50.0	95.80	92.5	89.10	99.20	**99.30**	95.94
Wood	73.0	77.0	62.0	95.00	92.1	85.80	**96.40**	90.70	94.96
Mean_tex	78.2	83.1	57.6	97.63	93.16	93.90	**97.88**	96.84	97.65
Bottle	93.0	87.0	86.0	98.64	97.8	98.40	**99.10**	98.30	98.63
Cable	82.0	90.0	78.0	96.75	91.9	84.20	94.70	96.00	**96.89**
Capsule	94.0	74.0	84.0	98.62	96.8	92.80	94.30	97.60	**98.67**
Hazelnut	97.0	98.0	87.0	98.81	98.2	96.10	**99.70**	97.60	98.88
Metal_nut	89.0	94.0	76.0	97.59	97.2	92.50	**99.50**	98.40	97.61
Pill	91.0	83.0	87.0	98.34	96.5	95.70	97.60	**98.50**	98.35
Screw	96.0	97.0	80.0	98.40	97.4	**98.80**	97.60	96.50	98.35
toothbrush	92.0	94.0	90.0	99.00	97.9	98.90	98.10	94.90	**99.01**
Transistor	80.0	93.0	80.0	97.69	73.7	87.70	90.90	88.00	**97.84**
Zipper	88.0	78.0	78.0	98.98	95.6	97.80	98.80	94.20	**99.00**
Mean_obj	90.2	88.8	82.6	98.28	94.3	94.30	97.03	96.00	**98.32**
Mean	86.2	86.9	74.27	98.06	93.92	94.20	97.31	96.28	**98.10**

Bold represents the best results

0.23%, reaching 97.65%. The object category achieved the best average localization result of 98.32%. Meanwhile, MCAD achieved a final average localization result of 98.10% for all classes. MCAD has achieved SOTA localization results in multiple classes, for instance *carpet, cable, capsule, toothbrush, transistor, zipper*. Although DRAEM [61] achieved the best localization results in several classes (grid, wood, bottle, hazelnut and metal_-nut). However, MCAD has achieved excellent performance at these levels as well. Moreover, the localization results of MCAD in the *carpet, capsule and transistor* classes are much better than those of DRAEM. The outcomes of MCAD anomaly localization on the MVTec-AD dataset are depicted in Fig. 3. For complex anomalies like texture classes, MCAD locates the anomalous areas accurately. Similarly, the anomalous areas of the object class are also achieved to locate accurately. As a result, MCAD achieves excellent localization performance of the MVTec-AD dataset.

Anomaly classification This part discusses anomaly classification. As shown in Fig. 1, the MCAD framework contains a multi-classification model for anomaly classification. Since anomalous images exist only in the inference process, the multi-classification model classifies only in the inference process. The results of anomaly classification on the MVTec-AD dataset are presented in Table 4. The uppermost row illustrates the count of anomalies for each data type within the MVTec-AD dataset. For example, the category *carpet* contains five categories of exceptions, namely *color, cut, hole, metal_contamination and thread.*

In Sect. 3.4, the multi-classification model uses feature concatenation. Therefore, the input F0 in Table 4 represents the features using the 0-th critical layer. Similarly, F0, 1, 2, 3, 4 denotes the utilization of all feature maps from the critical layers within the student model as input for the multi-classification model. Figure 4 provides an in-depth presentation of the classification result for each class within the context of the MVTec-AD dataset. Furthermore, the recognition accuracy increases with the addition of features. The multi-classification model achieved more than 90% AUROC classification results in the categories *leather, tile, hazelnut and zipper*. However, the classification results are unsuitable for complex textures and unusually obscure classes such as *grid, capsule and pill*. The anomalous features of these categories are similar to the normal features and are challenging to classify. Nevertheless, the multi-classification model achieved an average AUROC classification result of 76.37% in the MVTec-AD dataset. Consequently, MCAD is well-positioned to classify each anomaly for industry applications accurately.

4.4 Ablation study

The influence of different critical layer features of the student model on the anomaly classification results is discussed in Sect. 4.3. Within this section, we analyze the

Table 4 Anomaly classification results with accuracy on MVTec-AD

	Input	Carpet	Grid	Leather	Tile	Wood	Bottle	Cable	Capsule	Hazelnut	Metal_nut	Pill	Screw	Toothbrush	Transistor	Zipper	Mean
Classes		5	5	5	5	5	3	8	5	4	4	7	5	1	4	7	
Accuracy	F^{0}	79.51	37.63	89.70	86.37	80.22	82.30	76.26	37.59	84.11	66.14	41.52	57.40	81.18	79.91	88.15	71.20
Accuracy	$F^{0,1}$	81.74	40.59	90.71	87.22	81.33	81.62	78.41	39.74	87.55	69.53	43.64	59.60	83.68	82.84	89.85	73.20
Accuracy	$F^{0,1,2}$	83.96	41.36	93.19	87.83	83.65	84.34	79.55	40.16	87.89	70.39	44.70	61.46	85.68	83.67	90.76	74.57
Accuracy	$F^{0,1,2,3}$	84.85	41.97	94.16	88.78	84.29	85.97	80.66	40.37	88.71	70.67	45.63	61.57	86.07	84.38	91.44	75.30
Accuracy	$F^{0,1,2,3,4}$	85.00	42.86	**95.24**	**90.48**	85.71	87.50	81.48	41.67	**90.00**	71.43	46.67	62.07	87.50	85.00	**92.96**	**76.37**

Bold represents the best results

impact of feature activation values across different critical layer features on the results of anomaly detection and localization. Besides, for the hyperparameter λ in Eq. (8), we discuss the anomaly detection result for different λ.

The incorporation of feature activation values from distinct critical layers exerts a discernible influence on the ultimate outcome, during the course of anomaly detection and localization. Table 5 shows the anomaly detection and localization results using different network layers feature activation values. Where FA1 represents the feature activation value representing the first critical layer. With single-layer feature activation values, FA4 achieved the best anomaly detection and localization results with 96.83 and 97.32%, respectively. When using two feature activation values, FA3 + FA4 achieves the best anomaly detection and localization results of 96.92 and 97.23%, respectively. Compared to using FA1 + FA2, the AUROC results for anomaly detection and localization improved by 1.74 and 2.40%, respectively. The best results are achieved in the FA2 + FA3 + FA4 with 97.58 and 98.10%. This result exceeds the results using FA1 + FA2 + FA3 + FA4, which shows that more feature activation values do not necessarily yield better results. This is also reflected in the experiments using FA3 + FA4 versus FA1 + FA2 + FA3. FA1 + FA2 + FA3 achieves 94.67 and 95.96% detection and localization results. And FA3 + FA4 achieves 96.92 and 97.23% detection and localization results. Although the feature activation values of FA4 are based on FA1, the feature extraction capability is enhanced as the depth of the network increases. Thus, FA4 gives better results than other methods in using only a single value.

Different angle distillation loss weights show a significant difference in anomaly detection results in Fig. 5. As the hyperparameter λ increases, there is a significant improvement in the detection results. However, this improvement is not endless, the accuracy instead decreases as λ exceeds 0.6.

4.5 Discussions

The MCAD framework proposed is an anomaly detection and localization model containing multi-classification in this paper. The teacher–student model is used in the anomaly detection and localization process. Utilizing ResNet18 and ResNet10 as the teacher and student models, respectively, we apply relational knowledge distillation to transfer knowledge from teacher to student. And, leveraging feature activation values from pivotal layers in both models to enhance anomaly detection and localization. The multi-classification model is a lightweight model containing residual structure. Experimental results show that MCAD simultaneously achieves anomaly classification with excellent anomaly detection and localization.

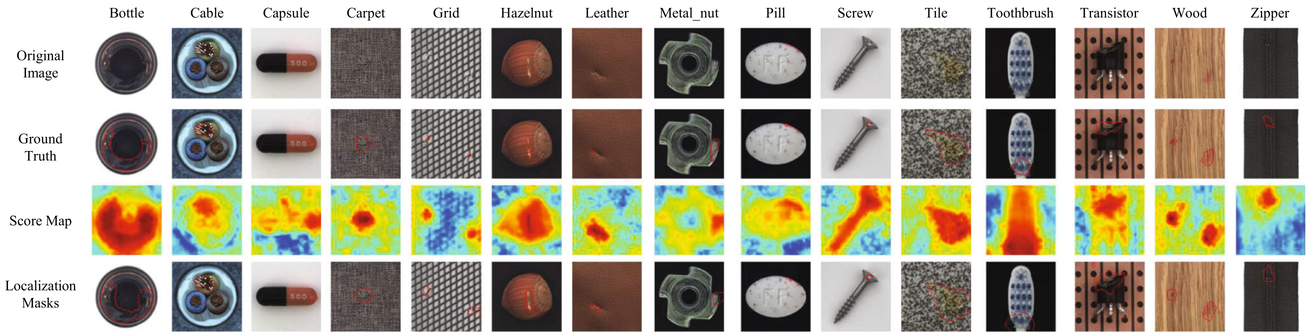

Fig. 3 The original images (*1st row*) of each class in MVTec-AD dataset. The input images of each sample with the ground truth (*2nd row*) anomaly mask. The anomaly score map (*3rd row*) is estimated using the MCAD framework and the localization mask is generated (*4th row*)

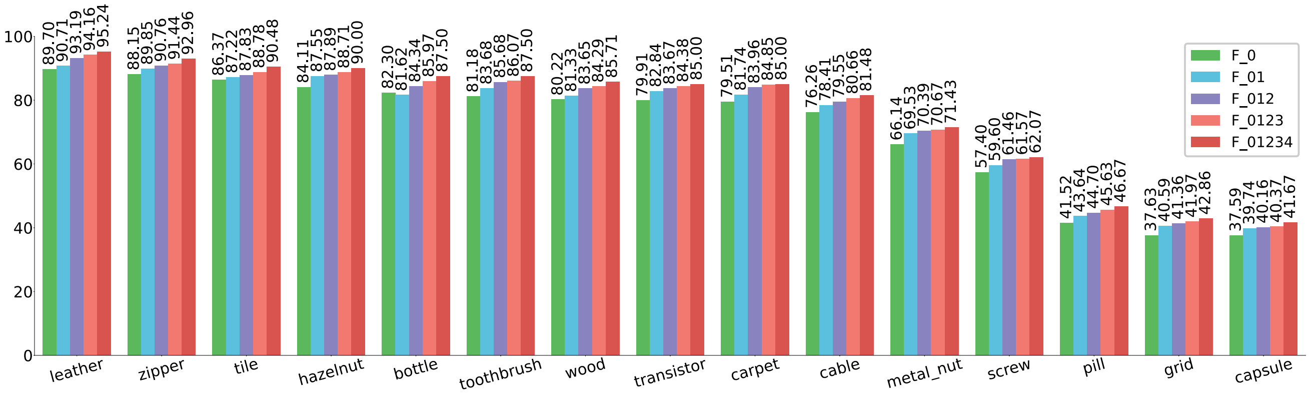

Fig. 4 AUROC results of anomaly classification for different classes

Table 5 Anomaly detection and localization AUROC(%) results for different feature activation (FA) values

FA1	FA2	FA3	FA4	Anomaly detection	Anomaly localization
✓				90.34	91.42
	✓			92.51	90.37
		✓		93.97	94.10
			✓	96.83	97.32
✓	✓			95.18	94.83
	✓	✓		94.58	95.64
		✓	✓	96.92	97.23
✓	✓	✓		94.67	95.96
	✓	✓	✓	**97.58**	**98.10**
✓	✓	✓	✓	96.89	98.08

Bold represents the best results

Although MCAD has demonstrated outstanding performance in anomaly detection, localization, and classification, the results on the FashionMNIST dataset could be better than MKD [42]. In addition, the results of MCAD are less well than the method of reconstruction-based DRAEM [61] in anomaly detection. DRAEM achieved the best anomaly detection results across multiple categories in the MVTec-AD dataset. Despite the promising results obtained by MCAD on anomaly localization, the results on some categories could be better than those of DRAEM. At the same time, feature fusion capabilities must be improved for anomaly classification to enable better performance on complex textures and categories of classification ambiguity, such as *grid, capsule and pill*.

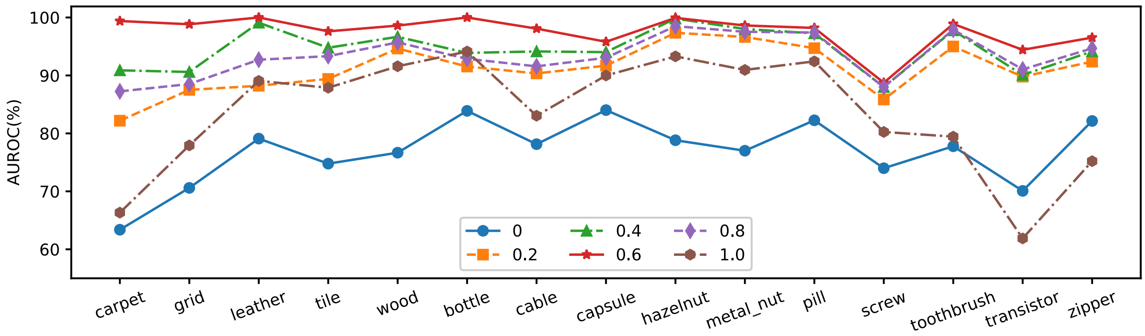

Fig. 5 Anomaly detection accuracy of different λ (Eq. 8) on different categories of MVTec-AD

5 Conclusion

This paper proposes an MCAD framework for anomaly detection, localization, and classification. First, the teacher–student model is applied for anomaly detection and localization. Improving teacher-to-student model knowledge transfer utilizing RKD. Subsequently, a multi-classification model is employed to accomplish the classification of anomalous images. MCAD achieves 97.58% AUROC and 98.10% AUROC in MVTecAD data on anomaly detection and localization, respectively. In terms of anomaly classification, the multi-classification model secures an average classification accuracy of 76.37% on the MVTec-AD dataset. The experimental findings demonstrated that MCAD achieves outstanding anomaly detection and localization performance with anomaly classification. MCAD addresses the requirements in industrial scenarios, especially for some texture class localization problems and object class classification problems. However, the multi-classification model still has limitations, for example, the class of *grid, capsule, and pill*. This paper believes that these limitations are addressed by treating complex backgrounds. In future work, the primary focus lies in enhancing feature fusion capabilities, especially within complex backgrounds, to elevate the model's performance in anomaly classification.

Funding Open Access funding provided by Ritsumeikan University.

Data availability The data that support the findings of this study are openly available in MVTec Software at MVTec AD, reference number [51, 52].

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Paula Monteiro R, Lozada MC, Mendieta DRC, Loja RVS, Bastos Filho CJA (2022) A hybrid prototype selection-based deep learning approach for anomaly detection in industrial machines. Expert Syst Appl 204:117528
2. Creswell A, White T, Dumoulin V, Arulkumaran K, Sengupta B, Bharath AA (2018) Generative adversarial networks: an overview. IEEE Signal Process Mag 35(1):53–65
3. Roth K, Pemula L, Zepeda J, Schölkopf B, Brox T, Gehler P (2022) Towards total recall in industrial anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 14318–14328
4. Kim D, Park C, Cho S, Lee S (2023) Fapm: Fast adaptive patch memory for real-time industrial anomaly detection. In: ICASSP 2023-2023 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, pp 1–5
5. Jang J, Hwang E, Park S-H (2023) N-pad: Neighboring pixel-based industrial anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 4364–4373
6. Rudolph M, Wehrbein T, Rosenhahn B, Wandt B (2023) Asymmetric student–teacher networks for industrial anomaly detection. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp 2592–2602
7. Hooshmand MK, Hosahalli D (2022) Network anomaly detection using deep learning techniques. CAAI Trans Intell Technol 7(2):228–243
8. Flusser M, Somol P (2022) Efficient anomaly detection through surrogate neural networks. Neural Comput Appl 34(23):20491–20505
9. Shi Y, Shen H (2022) Unsupervised anomaly detection for network traffic using artificial immune network. Neural Comput Appl 34(15):13007–13027

10. Zavrak S, Iskefiyeli M (2023) Flow-based intrusion detection on software-defined networks: a multivariate time series anomaly detection approach. Neural Comput Appl 35(16):12175–12193
11. Li Z, Wang Y, Xiao C, Ling Q, Lin Z, An W (2023) You only train once: learning a general anomaly enhancement network with random masks for hyperspectral anomaly detection. IEEE Trans Geosci Remote Sens 61:1–18
12. Wang D, Gao L, Qu Y, Sun X, Liao W (2023) Frequency-to-spectrum mapping GAN for semisupervised hyperspectral anomaly detection. CAAI Trans Intell Technol
13. Shi Y, Yang J, Qi Z (2021) Unsupervised anomaly segmentation via deep feature reconstruction. Neurocomputing 424:9–22
14. Ristea N-C, Madan N, Ionescu RT, Nasrollahi K, Khan FS, Moeslund TB, Shah M (2022) Self-supervised predictive convolutional attentive block for anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 13576–13586
15. Liu R, Liu W, Zheng Z, Wang L, Mao L, Qiu Q, Ling G (2023) Anomaly-GAN: a data augmentation method for train surface anomaly detection. Expert Syst Appl 228:120284
16. Schlegl T, Seeböck P, Waldstein SM, Schmidt-Erfurth U, Langs G (2017) Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: Information processing in medical imaging—25th international conference, IPMI 2017, Boone, NC, USA, June 25–30, 2017, proceedings, vol 10265. Springer, Berlin, pp 146–157
17. Akcay S, Atapour-Abarghouei A, Breckon TP (2019) Ganomaly: Semi-supervised anomaly detection via adversarial training. In: Computer vision—ACCV 2018: 14th Asian conference on computer vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part III 14. Springer, Berlin, pp 622–637
18. Akçay S, Atapour-Abarghouei A, Breckon TP (2019) Skip-ganomaly: skip connected and adversarially trained encoder–decoder anomaly detection. In: 2019 International joint conference on neural networks (IJCNN). IEEE, pp 1–8
19. Devlin J, Chang M, Lee K, Toutanova K (2019) BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2–7, 2019, volume 1 (long and short papers). Association for Computational Linguistics, pp 4171–4186
20. Yue X, Li H, Meng L (2023) An ultralightweight object detection network for empty-dish recycling robots. IEEE Trans Instrum Meas 72:1–12
21. Pirnay J, Chai K (2022) Inpainting transformer for anomaly detection. In: Image analysis and processing—ICIAP 2022: 21st international conference, Lecce, Italy, May 23–27, 2022, proceedings, Part II. Springer, Berlin, pp 394–406
22. Yue X, Meng L (2023) YOLO-MSA: A multi-scale stereoscopic attention network for empty-dish recycling robots. IEEE Trans Instrum Meas
23. Chen J, Lu Y, Yu Q, Luo X, Adeli E, Wang Y, Lu L, Yuille AL, Zhou Y (2021) Transunet: transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306
24. Mousakhan A, Brox T, Tayyub J (2023) Anomaly detection with conditioned denoising diffusion models. arXiv preprint arXiv:2305.15956
25. Teng Y, Li H, Cai F, Shao M, Xia S (2022) Unsupervised visual defect detection with score-based generative model. arXiv preprint arXiv:2211.16092
26. Zhang H, Wang Z, Wu Z, Jiang Y-G (2023) Diffusionad: denoising diffusion for anomaly detection. arXiv preprint arXiv:2303.08730
27. Wyatt J, Leach A, Schmon SM, Willcocks CG (2022) Anoddpm: anomaly detection with denoising diffusion probabilistic models using simplex noise. In: 2022 IEEE/CVF conference on computer vision and pattern recognition workshops (CVPRW), pp 649–655
28. Yi J, Yoon S (2020) Patch SVDD: Patch-level SVDD for anomaly detection and segmentation. In: Proceedings of the Asian conference on computer vision
29. Zhang Z, Deng X (2021) Anomaly detection using improved deep SVDD model with data structure preservation. Pattern Recogn Lett 148:1–6
30. Hu C, Chen K, Shao H (2021) A semantic-enhanced method based on deep SVDD for pixel-wise anomaly detection. In: 2021 IEEE International conference on multimedia and expo (ICME). IEEE, pp 1–6
31. Ruff L, Vandermeulen R, Goernitz N, Deecke L, Siddiqui SA, Binder A, Müller E, Kloft M (2018) Deep one-class classification. In: Proceedings of the 35th international conference on machine learning, pp 4393–4402
32. Kobyzev I, Prince SJ, Brubaker MA (2020) Normalizing flows: an introduction and review of current methods. IEEE Trans Pattern Anal Mach Intell 43(11):3964–3979
33. Rudolph M, Wandt B, Rosenhahn B (2021) Same same but different: semi-supervised defect detection with normalizing flows. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp 1907–1916
34. Rudolph M, Wehrbein T, Rosenhahn B, Wandt B (2022) Fully convolutional cross-scale-flows for image-based defect detection. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp 1088–1097
35. Gudovskiy D, Ishizaka S, Kozuka K (2022) CFLOW-AD: Real-time unsupervised anomaly detection with localization via conditional normalizing flows. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision (WACV), pp 98–107
36. Yu J, Zheng Y, Wang X, Li W, Wu Y, Zhao R, Wu L (2021) Fastflow: unsupervised anomaly detection and localization via 2d normalizing flows. arXiv preprint arXiv:2111.07677
37. Yan R, Zhang F, Huang M, Liu W, Hu D, Li J, Liu Q, Jiang J, Guo Q, Zheng L (2022) Cainnflow: convolutional block attention modules and invertible neural networks flow for anomaly detection and localization tasks. arXiv preprint arXiv:2206.01992
38. Lee S, Lee S, Song BC (2022) CFA: Coupled-hypersphere-based feature adaptation for target-oriented anomaly localization. IEEE Access 10:78446–78454
39. Bae J, Lee J-H, Kim S (2022) Image anomaly detection and localization with position and neighborhood information. arXiv preprint arXiv:2211.12634
40. Bergmann P, Fauser M, Sattlegger D, Steger C (2020) Uninformed students: student–teacher anomaly detection with discriminative latent embeddings. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)
41. Wang G, Han S, Ding E, Huang D (2021) Student–teacher feature pyramid matching for anomaly detection. arXiv preprint arXiv:2103.04257
42. Salehi M, Sadjadi N, Baselizadeh S, Rohban MH, Rabiee HR (2021) Multiresolution knowledge distillation for anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 14902–14912
43. Yamada S, Hotta K (2021) Reconstruction student with attention for student-teacher pyramid matching. arXiv preprint arXiv:2111.15376
44. Yamada S, Kamiya S, Hotta K (2022) Reconstructed student–teacher and discriminative networks for anomaly detection. In: 2022 IEEE/RSJ International conference on intelligent robots and systems (IROS). IEEE, pp 2725–2732

45. Deng H, Li X (2022) Anomaly detection via reverse distillation from one-class embedding. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 9737–9746
46. Cao Y, Wan Q, Shen W, Gao L (2022) Informative knowledge distillation for image anomaly segmentation. Knowl Based Syst 248:108846
47. Li Z, Li H, Meng L (2023) Model compression for deep neural networks: a survey. Computers 12(3):60
48. Park W, Kim D, Lu Y, Cho M (2019) Relational knowledge distillation. In: IEEE Conference on computer vision and pattern recognition, CVPR 2019, Long Beach, CA, USA, June 16–20, 2019. Computer Vision Foundation/IEEE, pp 3967–3976
49. Jeong J, Zou Y, Kim T, Zhang D, Ravichandran A, Dabeer O (2023) Winclip: Zero-/few-shot anomaly classification and segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 19606–19616
50. Liu T, Li B, Du X, Jiang B, Jin X, Jin L, Zhao Z (2023) Component-aware anomaly detection framework for adjustable and logical industrial visual inspection. arXiv preprint arXiv:2305.08509
51. Bergmann P, Batzner K, Fauser M, Sattlegger D, Steger C (2021) The MVTEC anomaly detection dataset: a comprehensive real-world dataset for unsupervised anomaly detection. Int J Comput Vis 129(4):1038–1059
52. Bergmann P, Fauser M, Sattlegger D, Steger C (2019) Mvtec AD—a comprehensive real-world dataset for unsupervised anomaly detection. In: IEEE Conference on computer vision and pattern recognition, CVPR 2019, Long Beach, CA, USA, June 16–20, 2019, pp 9592–9600
53. Salehi M, Arya A, Pajoum B, Otoofi M, Shaeiri A, Rohban MH, Rabiee HR (2021) ARAE: Adversarially robust training of autoencoders improves novelty detection. Neural Netw 144:726–736
54. Abati D, Porrello A, Calderara S, Cucchiara R (2019) Latent space autoregression for novelty detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)
55. Golan I, El-Yaniv R (2018) Deep anomaly detection using geometric transformations. In: Advances in neural information processing systems 31: annual conference on neural information processing systems 2018, NeurIPS 2018, December 3–8, 2018, Montréal, Canada, pp 9781–9791
56. Akcay S, Atapour-Abarghouei A, Breckon TP (2018) Ganomaly: semi-supervised anomaly detection via adversarial training. In: Asian conference on computer vision. Springer, Berlin, pp 622–637
57. Xia X, Pan X, He X, Zhang J, Ding N, Ma L (2021) Discriminative-generative representation learning for one-class anomaly detection. arXiv preprint arXiv:2107.12753
58. Liang Y, Zhang J, Zhao S, Wu R, Liu Y, Pan S (2023) Omni-frequency channel-selection representations for unsupervised anomaly detection. IEEE Trans Image Process
59. Huang C, Cao J, Ye F, Li M, Zhang Y, Lu C (2019) Inverse-transform autoencoder for anomaly detection. arXiv:1911.10676
60. Zavrtanik V, Kristan M, Skočaj D (2021) Reconstruction by inpainting for visual anomaly detection. Pattern Recogn 112:107706
61. Zavrtanik V, Kristan M, Skocaj D (2021) Draem—a discriminatively trained reconstruction embedding for surface anomaly detection. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV), pp 8330–8339
62. Schlüter HM, Tan J, Hou B, Kainz B (2022) Natural synthetic anomalies for self-supervised anomaly detection and localization. In: Computer vision—ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, proceedings, Part XXXI. Springer, Berlin, pp 474–489
63. Bergmann P, Löwe S, Fauser M, Sattlegger D, Steger C (2019) Improving unsupervised defect segmentation by applying structural similarity to autoencoders. In: Proceedings of the 14th international joint conference on computer vision, imaging and computer graphics theory and applications, VISIGRAPP 2019, volume 5: VISAPP, Prague, Czech Republic, February 25–27, 2019. SciTePress, pp 372–380
64. Liu W, Li R, Zheng M, Karanam S, Wu Z, Bhanu B, Radke RJ, Camps O (2020) Towards visually explaining variational autoencoders. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)