Remembering Normality: Memory-guided Knowledge Distillation for Unsupervised Anomaly Detection

Zhihao Gu[1]*, Liang Liu[2]*, Xu Chen[2]*, Ran Yi[1], Jiangning Zhang[2],
Yabiao Wang[2], Chengjie Wang[1,2], Annan Shu[3], Guannan Jiang[3], Lizhuang Ma[1]†
[1]Shanghai Jiao Tong University, China [2]Tencent YouTu Lab, China [3]CATL, China

Abstract

Knowledge distillation (KD) has been widely explored in unsupervised anomaly detection (AD). The student is assumed to constantly produce representations of typical patterns within trained data, named "normality", and the representation discrepancy between the teacher and student model is identified as anomalies. However, it suffers from the "normality forgetting" issue. Trained on anomaly-free data, the student still well reconstructs anomalous representations for anomalies and is sensitive to fine patterns in normal data, which also appear in training. To mitigate this issue, we introduce a novel ***Mem****ory-guided* ***K****nowledge-****D****istillation (****MemKD****) framework that adaptively modulates the normality of student features in detecting anomalies. Specifically, we first propose a normality recall memory (NR Memory) to strengthen the normality of student-generated features by recalling the stored normal information. In this sense, representations will not present anomalies and fine patterns will be well described. Subsequently, we employ a normality embedding learning strategy to promote information learning for the NR Memory. It constructs a normal exemplar set so that the NR Memory can memorize prior knowledge in anomaly-free data and later recall them from the query feature. Consequently, comprehensive experiments demonstrate that the proposed MemKD achieves promising results on five benchmarks.*

1. Introduction

Anomaly detection (AD) has attracted increasing attention in recent years for its wide applications, to name a few, defect detection [23], medical diagnosis [34], and video surveillance [10]. The lack of anomalous samples makes it more challenging and it is usually formulated as an unsupervised learning problem, only relying on normal data.

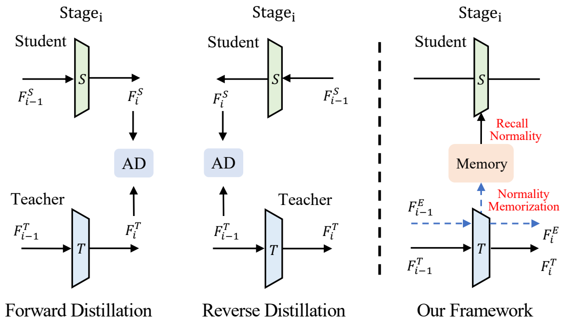


Figure 1. **Left**: Current knowledge distillation based paradigms. **Right**: Our method adopts the memory to strengthen the normality of student-generated features and learns the normal information via the normality embedding learning strategy (dotted lines).

Since anomalous data are unavailable in the training phase, a straightforward choice is to compare the normal data with the given target. To achieve this, memory bank based techniques [6, 24, 32] are proposed. They exploit the memory bank to store normal representations extracted by the ImageNet [9] pre-trained network. These features are then used to measure the normality distribution and outliers are considered to be anomalous. However, to compute anomaly maps, they need to search the entire memory bank through complex formulations, which increases the computational complexity as the size of the training set grows.

Reconstruction based methods [17, 11, 19] are conceptually simple and have been extensively explored for the task. It is expected that the reconstruction error of normal samples is lower than that of abnormal samples and anomalies can be detected by thresholding the difference between the input and the retrieved one. Nevertheless, they may fail to reconstruct subtle details from latent representations, resulting in large reconstruction errors for anomaly-free data.

Recent efforts tend to explore the Knowledge Distillation (KD) [13] and instead detect the anomaly at the feature level (left part in Fig. 1) for unsupervised AD. The student is assumed to constantly produce presentations of typical pat-

*Equal contribution. Work done when Zhihao Gu is an intern at CATL. ✉ : ellery-holmes@sjtu.edu.cn.

†Corresponding author.

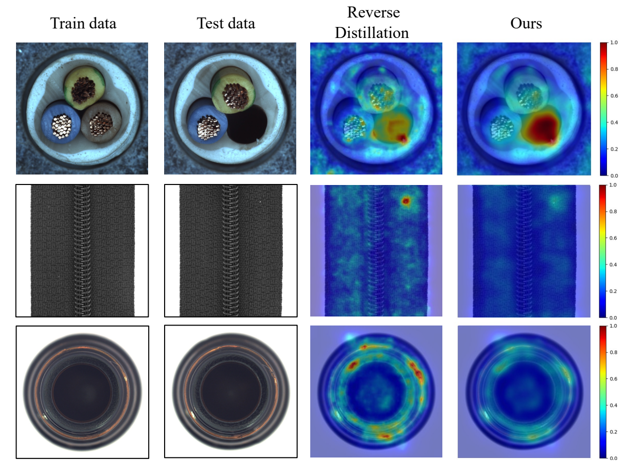


Figure 2. “Normality forgetting” on MVTec AD [23]. As a reference, we show a similar image from the training set for each test sample. Compared to RD [8], our MemKD accurately localizes the ‘missing cable’ on ‘cable’ and also suppresses the response to fine patterns in normal data (tiny dust on the ‘zipper’ and fine textures on the ‘bottle’), which also appear in training data.

terns within trained normal data, which is called “normality”, and the discrepancies between features in the teacher and student (T-S) network are used for anomaly detection. However, the assumption is not always ensured and the student suffers from the “normality forgetting” issue. For example, the student still produces anomalous representations for anomalies, resulting in slight feature discrepancy between the T-S. Finally, the model fails to detect the missing of an axial in the triaxial cable (1st row of Fig. 2). Besides, the capacity discrepancy between the T-S is more likely to make the student unable to capture fine patterns in anomaly-free data and gives inconsistent features to them, becoming sensitive. As shown in the last two rows of Fig. 2, the tiny dust and inessential textures are considered anomalous even if similar patterns have appeared in normal training data.

To tackle the above problem, we consider how to integrate normal information into student-generated features so that anomalies will not be presented on the representations and fine patterns can also be described. We take inspiration from the memory process of humans that associate visual cues with memorized knowledge to achieve this.

Following the above intuitions, we propose a novel ***Mem****ory-guided* ***K****nowledge* ***D****istillation* (**MemKD**) framework to handle the “normality forgetting” issue of the student, as demonstrated in the right part of Fig. 1. Specifically, we design a normality recall memory (NR Memory) to adaptively modulate the normality of student features. It stores normal information and recalls it from query features to strengthen the normality. Subsequently, we adopt a normality embedding learning strategy to guide the memory to learn prior knowledge about anomaly-free data. This strategy enables the NR memory to memorize the normality and further integrate it into the query feature by dealing with relevant information. Comprehensive experiments and visualization results validate the effectiveness of the proposed MemKD. In summary, the main contributions are threefold:

- We identify the “normality forgetting” issue of the student in knowledge distillation based anomaly detectors, and propose a novel *Memory-guided Knowledge Distillation* framework to address it.
- We design the NR Memory to recall normal information for strengthening the feature normality in the student network. Besides, we also devise a normality embedding learning strategy to promote the memorization of normal information from anomaly-free data,
- The proposed method outperforms its state-of-the-art competitors on five widely used benchmarks, and extensive experiments further validate its effectiveness.

2. Related Work

2.1. Unsupervised Anomaly Detection

The scarcity of anomalous data makes anomaly detection an unsupervised learning problem. To this end, various techniques are proposed where the most relevant methods are based on knowledge distillation and memory bank.

Knowledge distillation based methods train the student network via feature distillation from pre-trained teachers with anomaly-free data and the generated features from them are assumed to be discrepant for anomalies. To increase their discrimination, US [1] ensembles several models trained on normal data at different scales and MKD [28] distillates features at various stages of a pre-trained expert network. To avoid the structural similarity of the T-S hindering the representation capacity for anomalies, RD [8] instead builds the student upon the teacher model’s output and targets the teacher’s outputs from different stages. However, KD-based methods face the “normality forgetting” issue of the student network. Contrarily, we attempt to mitigate it by introducing a novel normality recall memory module.

Memory bank based methods exploit representations from pre-trained deep neural networks to model the distribution of normality. To achieve this, SPADE [5] proposes the semantic pyramid anomaly detection framework to estimate dense pixel-level correspondence between the target and the normal to detect the anomaly. PaDiM [6] presents a new paradigm for patch distribution modeling which makes use of a pre-trained network for patch embedding and correlations between different semantic levels are considered as cues to identify anomalies. PatchCore [24] further constructs a memory bank to store nominal patch features for comparison between the given target and normal features.

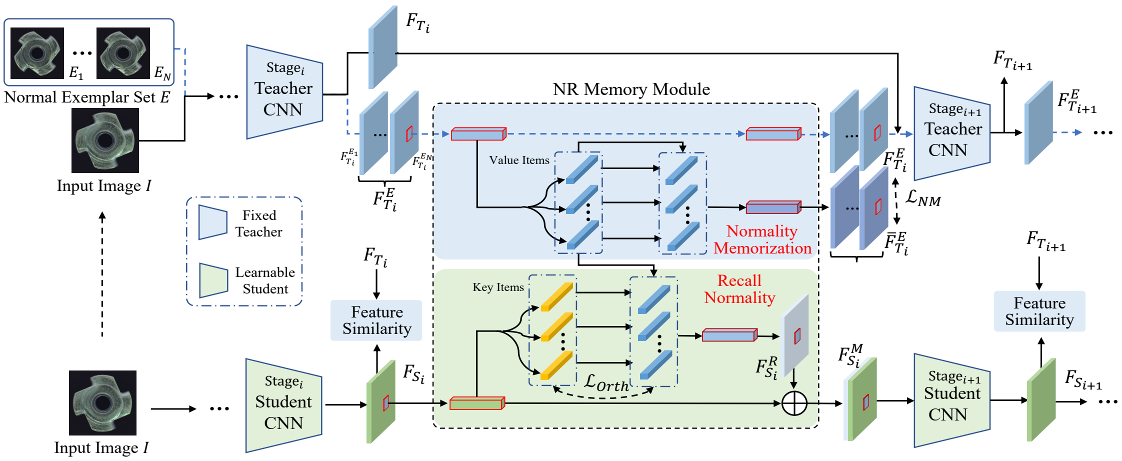


Figure 3. Overview of the proposed MemKD framework. It consists of an anomaly detection path with solid line and a normality embedding learning path with dashed line (used in the training phase only). We first design the NR memory module to strengthen the normality of the query feature F_i^S at the i^{th} stage in the student. Then we construct a normal exemplar set $\mathbb{E}$ to guide the memorization of prior knowledge from anomaly-free data for the memory module. ⊕ represents the concatenation operation. Best viewed in color.

Note that pre-trained models may overestimate the normality of abnormal features, CFA [18] proposes to adapt patch representations to the target dataset and presents the coupled-hypersphere feature adaptation framework. Different from them storing features of training data, our method distills the normality to the NR Memory, significantly reducing the memory consumption for storage (see Tab. 3).

2.2. Memory Module for AD

The memory module [12] attracts much attention and has been introduced into Auto-Encoder (AE) [17] to encode diverse normal patterns from normal training data. MemAE [10] proposes the memory-augmented AE to suppress the model's generalization capacity. Memory items are set as part of the network to automatically learns normal patterns for reconstruction. LMN [22] extends it and updates memory items in both training and testing by aggregating the information from the encoder. Rather than learning memory items via back-propagation, LND [20] designs a lightweight prototype unit to adaptively generate the normal prototypes. DA [15] generalizes MemAE to block-wise modules for maximizing the gap between the reconstruction error. Compared to the above AE-based methods, we specially design a novel NR Memory based on the key-value architecture and a normality embedding learning strategy for knowledge distillation. With the learning strategy, the NR Memory can efficiently memorize normal information and recall them from query features.

3. Preliminaries

Assume that there exists a training set $\mathcal{S}_{train}$ with sufficient anomaly-free samples and a test set $\mathcal{S}_{test}$ containing both normal and abnormal samples, which belong to the same category and are sampled from the same distribution. The goal of this task is to learn a model on $\mathcal{S}_{train}$ so that anomalies in $\mathcal{S}_{test}$ can be detected and localized. In this work, we adopt the knowledge distillation for unsupervised AD and briefly depict it as follows.

3.1. Knowledge Distillation for Anomaly Detection

In the context of unsupervised AD, the student model exposed to normal samples during training is expected to produce constantly anomaly-free representations and the discrepancies between features in the teacher and student (T-S) network provide essential evidence for anomaly detection.

Formally, given an input image $I \in \mathbb{R}^{C\times H\times W}$ (C, H, and W is the channel, height, and width of I), a frozen pre-trained teacher network T is exploited to extract features for I from multiple levels, denoted as $\{F_{T_i}\}_{i=1}^{K} \in \mathbb{R}^{C_i\times H_i\times W_i}$, where the index i indicates the i^{th} stage. Then a learnable student network S is required to reconstruct them. Let $\{F_{S_i}\}_{i=1}^{K} \in \mathbb{R}^{C_i\times H_i\times W_i}$ be the reconstructed features. To optimize the student, the cosine distance between F_{S_i} and F_{T_i} is measured [28]:

$$d(F_{S_i}, F_{T_i}) = 1 - \frac{\text{flat}(F_{S_i})}{\|\text{flat}(F_{S_i})\|_2} \cdot \frac{\text{flat}(F_{T_i})^{\mathsf{T}}}{\|\text{flat}(F_{T_i})\|_2}, \tag{1}$$

where $\text{flat}(\cdot) : \mathbb{R}^{C_i\times H_i\times W_i} \rightarrow \mathbb{R}^{C_iH_iW_i}$ is the vectorization function and $\|\cdot\|_2$ is the l_2 norm. Finally, the supervision for the knowledge distillation from the teacher to the student is given by accumulating $d(F_{S_i}, F_{T_i})$:

$$\mathcal{L}_{KD} = \sum_{i=1}^{K} d(F_{S_i}, F_{T_i}), \tag{2}$$

where K is the number of stages used for distillation.

At test time, the anomaly map $s_i \in \mathbb{R}^{H_i \times W_i}$ defined as the pixel-wise similarity between F_{S_i} and F_{T_i} at the i^{th} stage is first calculated:

$$s_i(h,w) = 1 - \sum_{c=1}^{C_i} \text{Sim}(F_{S_i}(c,h,w), F_{T_i}(c,h,w)), \quad (3)$$

where $\text{Sim}(\cdot,\cdot)$ is the cosine similarity. $s_i(h,w)$ is then up-sampled to $H \times W$ and pixel-wise accumulated to form the finally anomaly map $M \in \mathbb{R}^{H \times W}$ for the input image:

$$M(h,w) = g(\sum_{i=1}^{K} \text{Up}(s_i(h,w))), \quad (4)$$

where $\text{Up}(\cdot)$ denotes the bi-linear up-sampling and $g(\cdot)$ represents the Gaussian filter operation [24]. A larger score in the anomaly map means a higher probability of anomaly for that position, and the maximum in the anomaly map is defined as the image-level anomaly score [28, 8, 24].

Although the student network merely trained on normal data is expected to produce constantly anomaly-free representations, it suffers from the "normality forgetting" issue that still generates anomalous representations for anomalies and is sensitive to fine patterns in normal data. Therefore, we design the normality recall memory module to remit it.

4. Memory-guided Knowledge Distillation

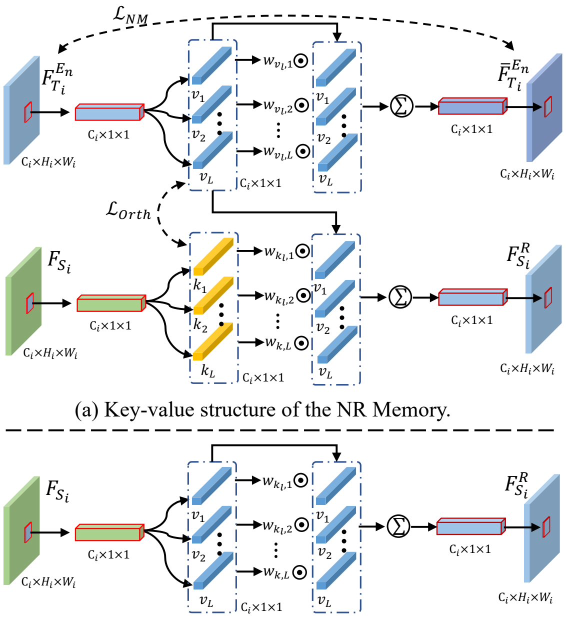
The primary problems of KD-based methods for AD lie in the lack of a mechanism that provides the student with normal information as an inference to produce constantly anomaly-free representations. Thus we consider how to integrate normal information into student-generated features.

The overall architecture of the proposed framework is demonstrated in Fig. 3. When obtaining a query feature F_{S_i} generated by the student, the goal is to recall the normal information via the normality recall memory (NR Memory). Then F_{S_i} passes through the memory module to generate the normalized feature $F_{S_i}^R$, concatenated with query F_{S_i} later to give $F_{S_i}^M$. The student network of next stage encodes $F_{S_i}^M$ to calculate similarity with $F_{T_{i+1}}$ via Eq. (1).

Moreover, we also employ a normality embedding learning (NEL) strategy to help the NR Memory learn prior knowledge in normal data during training. A normal exemplar set $\mathbb{E}$ with N randomly sampled anomaly-free images is first built. Then the teacher encodes $\mathbb{E}$ into the exemplar features $F_{T_i}^E = \{F_{T_i}^{E_n}\}_{n=1}^N$, named normality embedding, to train the memory module so that the learned knowledge can be recalled from the query feature F_{S_i}.

4.1. Normality Recall Memory Module

The purpose of the NR Memory module is to adaptively modulate the normality of student features. To model the process that recalling the memorized prior knowledge from the query feature, we design NR Memory as L key-value pairs $\mathbb{M} = \{(k_l, v_l) | k_l \in \mathbb{R}^C,\ v_l \in \mathbb{R}^C\}_{l=1}^L$. The key items are responsible for generating dynamical weights based on the query in order to recall the normal information stored in value items. Fig. 4 (a) illustrates its detailed structure.



(a) Key-value structure of the NR Memory.

(b) Memory module in MemAE.

Figure 4. Structure of (a) the proposed NR Memory and (b) the memory module in MemAE [10].

Concretely, given the query feature $F_{S_i} \in \mathbb{R}^{C_i \times H_i \times W_i}$, it is first flattened to $\hat{F}_{S_i} \in \mathbb{R}^{C_i \times H_i W_i}$ and the cosine similarity between position $\hat{F}_{S_i}(:,j)$ and each key item k_l is calculated to give the similarity vector $w_{k_l,j} \in \mathbb{R}^L$:

$$w_{k_l,j} = \frac{\exp(d(\hat{F}_{S_i}(:,j), k_l))}{\sum_{l=1}^{L} \exp(d(\hat{F}_{S_i}(:,j), k_l))}, \quad (5)$$

where $d(\cdot,\cdot)$ is the cosine similarity. The weight $w_{k_l,j}$ controls how much relevant normality needed to be recalled for integration at that localization. Then we aggregate value v_l by $w_{k_l,j}$ to obtain the normalized feature $\bar{F}_{S_i} \in \mathbb{R}^{C_i \times H_i W_i}$:

$$\bar{F}_{S_i}(:,j) = \sum_{l=1}^{L} w_{k_l,j} \cdot v_l. \quad (6)$$

Finally, $\bar{F}_{S_i}$ is reshaped to $F_{S_i}^R \in \mathbb{R}^{C_i \times H_i \times W_i}$, which is further concatenated with F_{S_i} to form the input for the student network of next stage.

Due to the different roles the key and value play, each k_i should be as independent as possible from each v_j. Therefore, a pairwise orthogonal loss is proposed:

$$\mathcal{L}_{Orth} = \frac{1}{L^2}\sum_{i=1}^{L}\sum_{j=1}^{L} d(k_i, v_j). \tag{7}$$

4.2. Normality Embedding Learning

To ensure that the NR Memory is able to memorize and recall normal information, we adopt a normality embedding learning strategy to model the process. During training, we randomly sample N normal images from the training set at every iteration to construct the normal exemplar set $\mathbb{E} = \{E_1, E_2, \ldots, E_N\}$. Then the teacher T encodes them into features of the i^{th} stage as $\mathbb{F}_{T_i}^{E} = \{F_{T_i}^{E_1}, F_{T_i}^{E_2}, \ldots, F_{T_i}^{E_N}\}$, where $F_{T_i}^{E_n} \in \mathbb{R}^{C_i \times H_i \times W_i}$, named normality embedding. By using exemplars, the essential information of normality is preserved in v_l and can be efficiently recalled.

Specifically, each exemplar feature $F_{T_i}^{E_n}$ is first flattened to $\hat{F}_{T_i}^{E_n} \in \mathbb{R}^{C_i \times H_i W_i}$ and the cosine similarity between position $\hat{F}_{T_i}^{E_n}(:, j)$ and v_l is measured. With the similarity, we obtain the weight $w_{v_l,j} \in \mathbb{R}^L$ via the softmax activation:

$$w_{v_l,j} = \frac{\exp(d(\hat{F}_{T_i}^{E_n}(:, j), v_l))}{\sum_{l=1}^{L}\exp(d(\hat{F}_{T_i}^{E_n}(:, j), v_l))}. \tag{8}$$

The weight $w_{v_l,j}$ decides how much stored normality needs to be used for retrieving $F_{T_i}^{E_n}$. Therefore, the reconstructed $\bar{F}_{T_i}^{E_n}$ is given by aggregating v_l with $w_{v_l,j}$:

$$\bar{F}_{T_i}^{E_n} = \sum_{l} w_{v_l,j} \cdot v_l. \tag{9}$$

To make sure that the v_l memorizes normal information from those normality embedding, we employ the normality memorization loss to minimize their difference:

$$\mathcal{L}_{NM} = \frac{1}{N}\sum_{n=1}^{N}\|F_{T_i}^{E_n} - \bar{F}_{T_i}^{E_n}\|_2^2. \tag{10}$$

Since the current sample is different from the N normal samples, the prior knowledge of normality embedded in v_l is more general with the restraint from Eq. (10).

Finally, the overall loss function is formulated as:

$$\mathcal{L} = \mathcal{L}_{KD} + \lambda_1\mathcal{L}_{NM} + \lambda_2\mathcal{L}_{Orth}, \tag{11}$$

where λ_1 and λ_2 are balancing hyper-parameters.

Discussion. It should be emphasized that the proposed memory is different from [10] (Fig. 4 (b)) in several aspects. First, [10] is AE-based and ours is designed for KD. Second, we define our NR Memory as the key-value structure to model the recall process and specially devise the NEL strategy to guide the learning of general prior knowledge from normal data, which is more comprehensive than their mechanism. Therefore, our method achieves better results. Structural differences are demonstrated in Fig. 4 and experimental comparisons are presented in Tab. 3 and Fig. 7.

5. Experiments

5.1. Dataset

To evaluate the effectiveness of the proposed method, we perform state-of-the-art comparisons on five benchmarks, *i.e.*, MVTec AD [23], VisA [35], MPDD [16], MVTec 3D-AD [2] and Eyecandies [4].

MVTec AD is a well-studied benchmark for anomaly detection and contains more than 5000 images for 15 classes. Anomalies are mainly in various structural changes.

VisA is the largest industrial anomaly detection dataset to date, which is composed of 10,821 high-resolution color images of 12 objects. The anomaly type covers both surface and structural defects.

MPDD is specially collected for defects produced during painted metal part fabrication and owns 6 classes of about 1300 images. Objects have various spatial orientations, light intensities, and non-homogeneous backgrounds.

MVTec 3D-AD includes 4,147 scans captured by a 3D sensor from 10 object categories, each of them providing both RGB and point cloud data. Defects mainly lie in the geometric structure.

Eyecandies owns 10 classes of 15000 synthetic images and challenges like complex textures, self-occlusions, and specularities are presented. It provides surface normal maps and depth for each image.

5.2. Implementation Details.

Evaluation Metrics. The Area Under the Receiver Operator Curve (AUROC) and Precision Recall (AUPR/AP) are adopted to measure the ability of anomaly detection and localization. For localization, PRO [1] is also calculated.

Implementation Details. All experiments are implemented using Pytorch. For MVTec 3D-AD and Eyecandies datasets, we only use color images for experiments. During training, images are resized into 256×256 and Adam is used as the optimizer with a learning rate of 0.005. We train the model for 100 epochs with batch size 16 and no augmentations are applied. N, L, λ_1 and λ_2 are set 16, 50, 0.1 and 0.1, respectively. Unless otherwise specified, the teacher network is an ImageNet [9] pre-trained WideResNet50. Following the common practice [24, 8], features from the first three stages are used for anomaly detection and thus $K = 3$. We choose RD [8] as the baseline model and insert three memory modules into the last three stages of the student and randomly initialize them.

5.3. Main Results

Anomaly detection and localization. Tab. 1 shows the comparison results of anomaly detection and localization on (a) MVTec AD [23], (b) VisA [35] and (c) MPDD [16] benchmarks. Since our MemKD recalls normal information to strengthen the feature normality, it improves the

	Normalizing Flows	Reconstruction Based		Memory Based				KD Based		
Metric	**FastFlow** [33]	**FAVAE** [7]	**DRAEM** [30]	**SPADE** [31]	**PaDiM** [6]	**CFA** [18]	**PatchCore** [24]	**ST** [5]	**RD** [8]	**Ours**
I-AUC	90.5	79.3	98.1	85.4	90.8	98.1	99.2	92.4	98.4	**99.6**
I-AP	94.5	91.3	99.0	94.0	95.4	99.3	99.8	95.7	99.5	**99.9**
P-AUC	95.5	88.9	97.5	95.5	96.6	97.1	98.1	95.4	97.8	**98.2**
P-AP	39.8	30.7	**68.9**	47.1	45.2	53.8	56.1	51.8	58.0	59.6
P-PRO	85.6	74.9	92.1	89.5	91.3	89.8	93.4	87.9	93.9	**94.5**

(a) Anomaly detection and localization performance on the MVTec AD dataset.

I-AUC	82.2	80.3	88.7	82.1	89.1	92.0	95.1	83.3	96.0	**97.6**
I-AP	84.3	84.3	90.5	84.7	89.5	93.5	96.2	87.3	96.5	**98.6**
P-AUC	88.2	88.0	93.5	85.6	98.1	84.3	**98.8**	83.4	90.1	98.4
P-AP	15.6	21.3	26.5	21.5	30.9	26.8	40.1	16.9	27.7	**44.1**
P-PRO	59.8	67.9	72.4	65.9	85.9	55.1	91.2	62.0	70.9	**94.9**

(b) Anomaly detection and localization performance on the VisA dataset.

I-AUC	88.7	57.0	94.1	78.4	70.6	92.3	94.8	87.6	92.7	**95.4**
I-AP	88.1	70.5	96.1	81.5	78.4	92.2	97.0	91.4	95.3	**97.3**
P-AUC	80.8	90.6	91.8	98.2	95.5	94.8	**99.0**	98.1	98.7	98.4
P-AP	11.5	8.8	28.8	34.2	15.5	28.3	43.2	35.4	45.5	**46.1**
P-PRO	49.8	70.6	78.1	92.6	84.8	83.2	93.9	93.9	95.3	**95.9**

(c) Anomaly detection and localization performance on the MPDD dataset.

Table 1. Quantitative results for anomaly detection and localization on (a) MVTec AD [23], (b) VisA [35] and (c) MPDD [16] benchmarks. We report Image AUC ↑, Image AP ↑, Pixel AUC ↑, Pixel AP ↑ and Pixel PRO ↑ for each method. Best results are highlighted in bold.

knowledge distillation-based methods (*e.g.,* ST and RD) for anomaly detection by 1.2%, 0.4% on MVTec AD, 1.6%, 1.1% on VisA and 2.7%, 2.0% on MPDD for image AUC and AP, respectively. Besides, the MemKD also outperforms its memory bank based counterparts, *e.g.,* CFA and PatchCore. It is reasonable that the learned prior knowledge from normal data is more general than that stored in the memory bank and thus the resulting features can better describe the normal distribution for detection. Without the mechanism to access normal information, normalizing flow and reconstruction-based methods give inferior results. Besides, MemKD also achieves leading anomaly localization performance on these benchmarks, especially on the MPDD dataset. Although falling a little (0.4%) behind PatchCore on P-AUC, it respectively outperforms the second-best method on more challenging P-AP and P-PRO by 4.0% and 3.9%, implying a better performance of localizing both small and large anomalies.

We also evaluate the MemKD on two 3D anomaly detection datasets: MVTec 3D-AD [2] and Eyecandies [4], where some anomalies are imperceptible only from RGB images. Color images are adopted and only image-level AUC is reported. The quantitative results are listed in Tab. 2. Remarkably, the proposed method still outperforms other methods and improves the baseline by 1.6% and 2.7%, respectively. All these results show the effectiveness of our method and indicate the essence of increasing the normality of student features for KD-based anomaly detection.

Complexity analysis. We measure the model complexity from the perspective of inference time (second on Intel i7) and memory consumption (MB) of the memory bank MPDD [16] dataset. Tab. 3 summarizes the results. Memory bank based methods store normal features from the training set and compare them with representations of the target at test time. Though owning fast inference speed and good performance, they consume more memory. In comparison, our MemKD performs better depending only on limited memory usage (0.3 MB) and the additionally consumed time (0.02s) is neglectable.

Discussion. We focus on improving the effectiveness of KD-based methods and the main time consumption lies in the propagation of the S-T. How to efficiently reduce it is an interesting problem and we leave it as our future work.

5.4. Ablation Study

We conduct comprehensive ablation studies to explore the effectiveness of each component on MVTec AD. More ablation studies can be found in the supplementary material.

Study on key components. The key elements of the proposed method include the NR Memory module, the pairwise orthogonal loss $\mathcal{L}_{Orth}$, and the normality memorization loss $\mathcal{L}_{NM}$. We investigate them and report the numerical results in Tab. 4 (a). The baseline (first row) owns inferior performance relying on the vanilla knowledge distillation architecture. Simply employing the NR Memory with $\mathcal{L}_{Orth}$ gives slight improvement. Nevertheless, more performance gains are derived from using $\mathcal{L}_{NM}$ (0.7% versus 0.3% for I-AUC, 0.2% versus 0.1% for P-AUC, and 0.4% versus 0.2% for P-PRO). Combining them all contributes to the best results and they have more significant impacts on image-level anomaly detection than pixel-level localization.

Study on the number of memory items. The number of items controls the amount of normality to be stored from normal data. Tab. 4 (b) studies its effects on different stages.

Method	Bagel	Cable Gland	Carrot	Cookie	Dowel	Foam	Peach	Potato	Rope	Tire	Mean
GANomaly [14]	48.5	51.2	53.2	50.4	55.8	48.6	46.7	51.1	48.1	52.8	50.7
DifferNet [21]	85.9	70.3	64.3	43.5	79.7	79.0	78.7	64.3	71.5	59.0	69.6
PADiM [6]	97.5	77.5	69.8	58.2	95.9	66.3	85.8	53.5	83.2	76.0	76.4
CS-Flow [26]	94.1	93.0	82.7	79.5	99.0	88.6	73.1	47.1	98.6	74.5	83.0
AST [27]	94.7	92.8	85.1	**82.5**	98.1	**95.1**	89.5	61.3	99.2	82.1	88.0
PatchCore [24]	87.6	88.0	79.1	68.2	91.2	70.1	69.5	61.8	84.1	70.2	77.0
RD [8]	98.7	93.7	94.3	69.0	98.1	84.7	91.3	**69.3**	99.3	85.3	88.4
Ours	**99.1**	**95.3**	**95.7**	71.6	**99.3**	87.5	**95.0**	67.9	**99.8**	**88.9**	**90.0**

(a) Anomaly detection performance on the MVTec 3D-AD dataset.

Method	Candy Cane	Chocolate Cookie	Chocolate Praline	Confetto	Gummy Bear	Hazelnut Truffle	Licorice Sandwich	Lollipop	Marsh.	Peppermint Candy	Mean
ST [5]	55.1	65.4	57.6	78.4	73.7	**79.0**	77.8	62.0	84.0	74.9	70.8
PADiM [6]	53.1	81.6	82.1	85.6	82.6	72.7	78.4	66.5	**98.7**	92.4	79.4
PatchCore [24]	52.5	95.4	53.4	90.7	64.6	46.6	76.2	68.2	94.4	91.5	73.4
AST [25]	57.9	87.4	68.8	97.3	78.9	54.7	85.6	87.9	96.5	96.3	81.1
AE [4]	52.7	84.8	77.2	73.4	59.0	50.8	69.3	76.0	85.1	73.0	70.1
RD [8]	56.5	94.7	84.3	96.7	84.5	62.5	85.6	81.4	95.2	99.1	84.0
Ours	**62.1**	**99.6**	**86.5**	**98.0**	**86.4**	65.8	**87.3**	**84.2**	97.3	**99.8**	**86.7**

(b) Anomaly detection performance on the Eyecandies dataset.

Table 2. Quantitative results for unsupervised anomaly detection on (a) MVTec 3D-AD [2] and (b) Eyecandies [4] dataset. We report the image-level AUROC (%) for RGB data. Methods achieving the top AUROC are highlighted.

Method	Infer. Time	Mem. Size	I-AUC/ P-AP/P-PRO
PaDiM	0.95s	41.2M	89.1 / 30.9 / 85.9
PatchCore	**0.22s**	48.4M	95.1 / 40.1 / 91.2
CFA	0.31s	29.2M	92.0 / 28.3 / 83.2
Baseline	0.31s	**0.0**M	96.0 / 27.7 / 70.9
MemAE	0.32s	0.2M	96.5 / 32.8 / 86.2
Ours	0.33s	0.3M	**97.6 / 44.1 / 94.9**

Table 3. Comparison of mean inference time (second) and memory consumption (MB) of memory bank on MPDD [16].

NR Mem.	$\mathcal{L}_{Orth}$	$\mathcal{L}_{NM}$	I-AUC	P-AUC	P-PRO
			98.4	97.8	93.9
✓	✓		98.7	97.9	94.1
✓		✓	99.1	98.0	94.3
✓	✓	✓	**99.6**	**98.2**	**94.5**

(a) Study on key components.

L_1	L_2	L_3	I-AUC	P-AUC	P-PRO
0	0	0	98.9	97.9	94.0
10	10	10	99.3	98.1	94.2
50	50	50	**99.6**	**98.2**	94.5
100	100	100	99.4	98.1	94.3
100	50	10	99.4	**98.2**	**94.6**

(b) Study on the number of memory item.

Table 4. Ablation study on key components and the item number.

Distillation Type	I-AUC	P-AUC	P-PRO
RD [8]	98.4	97.8	93.9
MemRD	**99.6**	**98.2**	**94.5**
MKD [28]	97.5	96.4	91.7
MemMKD	**98.7**	**97.6**	**93.0**

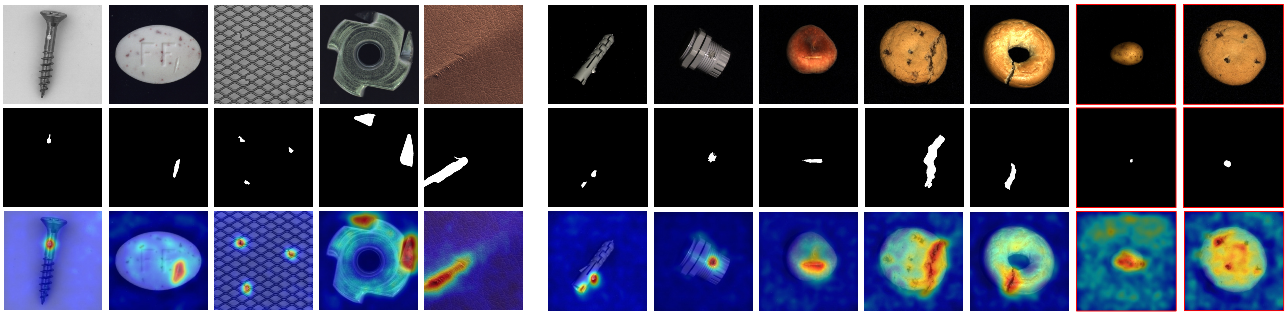
Table 5. Ablation study on the generalization of NR Memory. The prefix 'Mem' refers to methods with the NR Memory.

First of all, remembering and recalling the normality is beneficial for both anomaly detection and localization. More storage of normality contributes to better performance. Instead, a large number may introduce more parameters and lead to optimization difficulty. Finally, we increase L_1 and decrease L_3, resulting in a performance drop on I-AUC and slight improvement on P-PRO. For the sake of higher I-AUC, we set all of them 50 in this paper.

Study on the generalization of NR memory. Apart from the reverse distillation paradigm [8], we also apply the proposed framework to the forward distillation [28] architecture. Accordingly, we exploit the WideResNet50 as the teacher and the vanilla ResNet50 as the student. Specific architecture is included in the supplementary material. Tab. 5 lists the results. Our method improves the overall performance for different distillation paradigms and thus owns the favorable generalization ability.

5.5. Visualization Analysis

Anomaly localization. To intuitionally illustrate the performance of the proposed method, we visualize anomaly maps in Fig. 5 on (a) MVTec AD [23] and (b) MVTec 3D-AD [3] benchmarks. Although abnormal data is unavailable during training, our MemKD still accurately localizes anomalies of various sizes on anomalous images in both datasets. However, as demonstrated in the last two columns in Fig. 5 (b), anomalous images with geometrical anomalies are difficult to detect with only RGB images. More modalities may help tackle this issue. We will explore it later.



(a) Visualization on MVTec AD [23].

(b) Visualization on MVTec 3D-AD [3]. The last two columns are failure cases.

Figure 5. Qualitative results for anomaly localization. From top to bottom: RGB image, ground truth, and the predicted anomaly map. The proposed MemKD localizes tiny and conspicuous anomalies in both benchmarks. However, some anomalies can not be perceived only from the RGB data (last two columns in (b)). Best viewed in color.

Recall the normality. The k_l in the NR Memory are designed to assign the weights for aggregating the normal information stored in v_l. To comprehensively demonstrate it, we show the statistics of feature distance before (B) and after (A) recalling the normality in Fig. 6. Before querying the key, the averaged distance between anomalous and averaged normal features (Anomalous-B) in the test set is larger than that between normal and averaged normal ones (Normal-B). Once performing the recall operation, these distance coherently decreases and the averaged distance descent for Anomalous-A remains larger, implying that through assigning weights for recalling stored information, the feature normality of anomalous data indeed increases.

Learned normality. We also adopt the t-SNE [29] to visualize normality learned on the MVTec AD [23] dataset, as illustrated in Fig. 7. Each dot here represents the value item v_l. It can be observed that the distribution of normality from MemAE [10] presents inter- and intra-category disorder. Contrarily, the NEL strategy promotes the learning of normal information and makes the majority of value items within each category compact. And a few items are spread for dealing with diverse normal patterns in each category.

6. Conclusion

This paper presents a novel Memory-guided Knowledge Distillation framework for unsupervised anomaly detection, which handles the "normality forgetting" issue of the student network. We first design a normality recall memory to adaptively modulate the normality of student features by recalling normal information from query features for both normal and anomalous data. To guide the memory to memorize prior knowledge about normal data, we adopt a normality embedding learning strategy, which enables the NR memory to integrate normality by dealing with the relevant information. Consequently, the proposed method achieves promising results on five benchmarks.

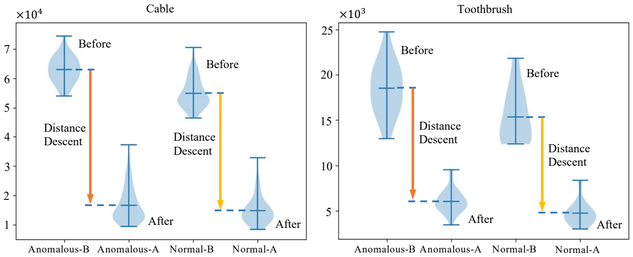
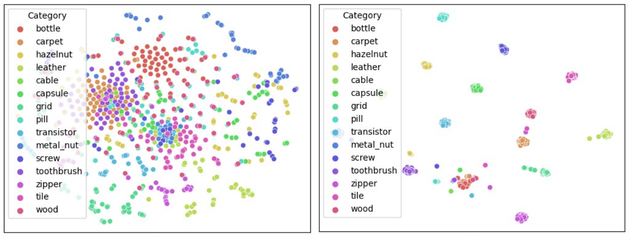


Figure 6. Statistics of feature distance before (B) and after (A) recalling the normality. The distance (y-axis) refers to the l_2 norm between the anomalous (or normal) features and averaged normal features in the test set. The NR Memory increases the feature normality and consistently reduces the distance.



(a) Value from MemAE.

(b) Value from our NR Memory.

Figure 7. t-SNE visualization [29] on value items from (a) MemAE and (b) the proposed NR Memory module. Items learned by the normality memorization loss are compact within each category.

Acknowledgements

This research is supported in part by National Natural Science Foundation of China (No. 61972157 and 72192821), Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102), Shanghai Science and Technology Commission (21511101200), Shanghai Sailing Program (22YF1420300 and 23YF1410500), CCF-Tencent Open Research Fund (RAGR20220121) and Young Elite Scientists Sponsorship Program by CAST (2022QNRC001).

References

[1] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *CVPR*, 2020.

[2] Paul Bergmann, Xin Jin, David Sattlegger, and Carsten Steger. The mvtec 3d-ad dataset for unsupervised 3d anomaly detection and localization. 2021.

[3] Paul Bergmann, Xin Jin, David Sattlegger, and Carsten Steger. The mvtec 3d-ad dataset for unsupervised 3d anomaly detection and localization. In *ICCVTA*, 2022.

[4] Luca Bonfiglioli, Marco Toschi, Davide Silvestri, Nicola Fioraio, and Daniele De Gregorio. The eyecandies dataset for unsupervised multimodal anomaly detection and localization. 2022.

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

[6] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: a patch distribution modeling framework for anomaly detection and localization. In *ICPR*, 2021.

[7] David Dehaene and Pierre Eline. Anomaly localization by modeling perceptual features. 2020.

[8] Hanqiu Deng and Xingyu Li. Anomaly detection via reverse distillation from one-class embedding. In *CVPR*, 2022.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[10] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *ICCV*, 2019.

[11] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[12] Alex Graves, Greg Wayne, alMalcolm Reynolds, and Tim et al. Harley. Hybrid computing using a neural network with dynamic external memory. In *Nature*, 2016.

[13] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NIPS Workshop*, 2015.

[14] Eliahu Horwitz and Yedid Hoshen. An empirical investigation of 3d anomaly detection and segmentation. 2022.

[15] Jinlei Hou, Yingying Zhang, Qiaoyong Zhong, Di Xie, Shiliang Pu, and Hong Zhou. Divide-and-assemble: Learning block-wise memory for unsupervised anomaly detection. In *ICCV*, 2021.

[16] Stepan Jezek, Martin Jonák, Radim Burget, Pavel Dvorak, and Milos Skotak. Deep learning-based defect detection of metal parts: evaluating current methods in complex conditions. 2021.

[17] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *Arxiv*, 2014.

[18] Sungwook Lee, Seunghyun Lee, and Byung Cheol Song. Cfa: Coupled-hypersphere-based feature adaptation for target-oriented anomaly localization. In *Access*, 2022.

[19] Yufei Liang, Jiangning Zhang, Shiwei Zhao, Runze Wu, Yong Liu, and Shuwen Pan. Omni-frequency channel-selection representations for unsupervised anomaly detection. In *TIP*, 2022.

[20] Hui Lv, Chen Chen, Zhen Cui, Chunyan Xu, Yong Li, and Jian Yang. Learning normal dynamics in videos with meta prototype network. In *CVPR*, 2021.

[21] Rudolph Marco, Wandt Bastian, and Rosenhahn Bodo. Same same but differnet: Semi-supervised defect detection with normalizing flows. In *WACV*, 2021.

[22] Hyunjong Park, Jongyoun Noh, and Bumsub Ham. Learning memory-guided normality for anomaly detection. In *CVPR*, 2020.

[23] Bergmann Paul, Fauser Michael, Sattlegger David, and Steger Carsten. Mvtec ad – a comprehensive real-world dataset for unsupervised anomaly detection. In *CVPR*, 2019.

[24] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *CVPR*, 2022.

[25] Marco Rudolph, Tom Wehrbein, Bodo Rosenhahn, and Bastian Wandt. Asymmetric student-teacher networks for industrial anomaly detection. 2022.

[26] Marco Rudolph, Tom Wehrbein, Bodo Rosenhahn, and Bastian Wandt. Fully convolutional cross-scale-flows for image-based defect detection. In *WACV*, 2022.

[27] Marco Rudolph, Tom Wehrbein, Bodo Rosenhahn, and Bastian Wandt. Asymmetric student-teacher networks for industrial anomaly detection. In *WACV*, 2023.

[28] Mohammadreza Salehi, Niousha Sadjadi, Soroosh Baselizadeh, Mohammad H Rohban, and Hamid R Rabiee. Multiresolution knowledge distillation for anomaly detection. In *CVPR*, 2021.

[29] Laurens van der Maaten and Geoffrey E. Hinton. Visualizing data using t-sne. 2008.

[30] Zavrtanik Vitjan, Kristan Matej, and Skočaj Danijel. Draem – a discriminatively trained reconstruction embedding for surface anomaly detection. In *ICCV*, 2021.

[31] Guodong Wang, Shumin Han, Errui Ding, and Di Huang. Student-teacher feature pyramid matching for anomaly detection. 2021.

[32] Yue Wang, Jinlong Peng, Jiangning Zhang, Ran Yi, Yabiao Wang, and Chengjie Wang. Multimodal industrial anomaly detection via hybrid fusion. In *CVPR*, 2023.

[33] Jiawei Yu1, Ye Zheng, Xiang Wang, Wei Li, Yushuang Wu, Rui Zhao, and Liwei Wu. Fastflow: Unsupervised anomaly detection and localization via 2d normalizing flows. 2021.

[34] Jianpeng Zhang, Yutong Xie, Yi Li, Chunhua Shen, and Yong Xia. Covid-19 screening on chest x-ray images using deep learning based anomaly detection. In *ArXiv*, 2020.

[35] Yang Zou, Jongheon Jeong, Latha Pemula, Dongqing Zhang, and Onkar Dabeer. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. 2022.