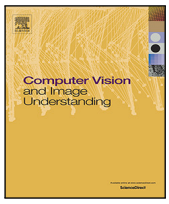
Computer Vision and Image Understanding 253 (2025) 104308

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu



Exploring plain ViT features for multi-class unsupervised visual anomaly detection

Jiangning Zhang [a], Xuhai Chen [b], Yabiao Wang [a], Chengjie Wang [a], Yong Liu [b,*], Xiangtai Li [c], Ming-Hsuan Yang [d], Dacheng Tao [c]

[a] Youtu Lab, Tencent, Shanghai, China
[b] Zhejiang University, Hangzhou, China
[c] Nanyang Technological University, Singapore
[d] Department of Computer Science and Engineering at the University of California, Merced, United States of America

ARTICLE INFO

Communicated by Xuelong Li

Keywords:
Multi-class anomaly detection
Vision transformer
Unsupervised learning
Feature reconstruction

ABSTRACT

This work studies a challenging and practical issue known as multi-class unsupervised anomaly detection (MUAD). This problem requires only normal images for training while simultaneously testing both normal and anomaly images across multiple classes. Existing reconstruction-based methods typically adopt pyramidal networks as encoders and decoders to obtain multi-resolution features, often involving complex sub-modules with extensive handcraft engineering. In contrast, a plain Vision Transformer (ViT) showcasing a more straightforward architecture has proven effective in multiple domains, including detection and segmentation tasks. It is simpler, more effective, and elegant. Following this spirit, we explore the use of only plain ViT features for MUAD. We first abstract a Meta-AD concept by synthesizing current reconstruction-based methods. Subsequently, we instantiate a novel ViT-based ViTAD structure, designed incrementally from both global and local perspectives. This model provide a strong baseline to facilitate future research. Additionally, this paper uncovers several intriguing findings for further investigation. Finally, we comprehensively and fairly benchmark various approaches using seven metrics and their average. Utilizing a basic training regimen with only an MSE loss, ViTAD achieves state-of-the-art results and efficiency on MVTec AD, VisA, and Uni-Medical datasets. *E.g.*, achieving 85.4 mAD that surpasses UniAD by +3.0 for the MVTec AD dataset, and it requires only 1.1 h and 2.3G GPU memory to complete model training on a single V100 that can serve as a strong baseline to facilitate the development of future research. Full code is available at https://zhangzjn.github.io/projects/ViTAD/.

1. Introduction

Visual Anomaly Detection (AD) aims to identify unusual or unexpected patterns within images that deviate significantly from the norm images. This technique helps prevent potential risks, improve safety, or enhance system performance, relying on its ability to reveal critical information across various domains. Thus, it has been widely used in visual inspection (Bergmann et al., 2019a; Liu et al., 2023b), medical image lesion detection (Bao et al., 2023), and video surveillance (Berroukham et al., 2023), to name a few.

As the unsupervised anomaly detection approach does not require high labeling costs, it has received increasing attention in recent years (Liu et al., 2024). Existing methods are generally developed based on a Single-class Unsupervised Anomaly Detection (SUAD) setting (Zavrtanik et al., 2021a; Deng and Li, 2022; Liu et al., 2023b), where each class requires a separate model for training that significantly increases the training and storage costs of the model. To alleviate the above problem, recent UniAD (You et al., 2022a) proposes the multi-class setting for the first time, but significant opportunities still exist for enhancing performance and reducing training costs. This work focuses on tackling this more challenging and practical setting, termed Multi-class Unsupervised Anomaly Detection (MUAD). Its purpose is to address the issue of model deployment across multiple scenarios using a single model, significantly reducing both development and deployment costs. This trend of transitioning from single-class to multi-class experimental settings is also evident in fields such as object detection and semantic segmentation. However, MUAD presents a greater challenge to the generalization ability of models that simultaneously model multiple categories. It forces models to learn general feature representations rather than overfitting to features of a single

* Corresponding author.
E-mail address: yongliu@iipc.zju.edu.cn (Y. Liu).

https://doi.org/10.1016/j.cviu.2025.104308
Received 14 June 2024; Received in revised form 12 November 2024; Accepted 28 January 2025
Available online 4 February 2025

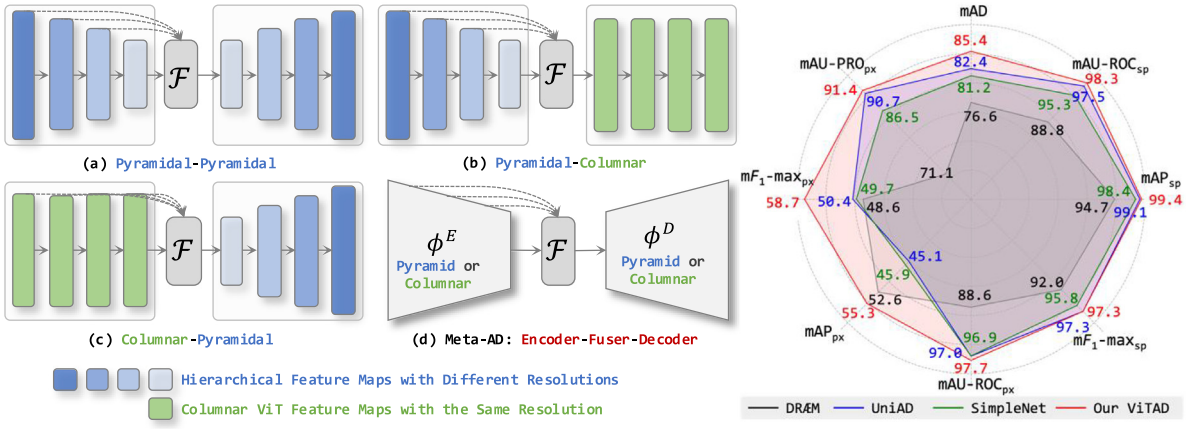


Fig. 1. Left: (a–c) display general reconstruction-based AD frameworks. (d) shows a Meta-AD framework that consists of image Encoder ϕ^E, Fuser $\mathcal{F}$, and Decoder ϕ^D. Blue and green represent columnar ViT features and hierarchical features, respectively. The dashed line indicates that the feature may be used by the Fuser $\mathcal{F}$. Right: Comprehensive quantitative comparison with popular methods by eight metrics on MVTec AD dataset (Bergmann et al., 2019a) (see Section 4.1 and Section 4.2). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

category, thereby demanding higher feature representation capabilities from AD models. The noticeable performance drop when current AD methods are transferred to this more practical setting further illustrates the significant challenge posed by MUAD (Zhang et al., 2024a).

Unsupervised anomaly detection methods in the literature can be broadly categorized as follows. (i) Augmentation-based methods (Li et al., 2021; Liu et al., 2023b; You et al., 2022a) enhance performance by introducing fabricated anomaly information, including CutPaste (Li et al., 2021) at the image level, and noise injection (Liu et al., 2023b) and feature jitter (You et al., 2022a) at the feature level. (ii) Embedding-based methods (Bergmann et al., 2020; Roth et al., 2022) map normal features to a compact space and identify anomalies by feature comparison. (iii) Reconstruction-based methods generally follow an encoder–decoder framework to reconstruct input images or features, and simple reconstruction errors serve as the anomaly map during inference. Thanks to the effectiveness, interpretability, and scalability, numerous subsequent researchers (Deng and Li, 2022; Tien et al., 2023; He et al., 2024b) follow this framework. According to whether the encoder–decoder structure is pyramidal (*e.g.*, multi-resolution ResNet (He et al., 2016)) or columnar (*e.g.*, single-resolution ViT (Dosovitskiy et al., 2021)), existing methods can be categorized as fully pyramidal structures (Zhou and Paffenroth, 2017; Akcay et al., 2019; Deng and Li, 2022) (Fig. 1(a)), pyramidal encoder with a dynamic ViT (You et al., 2022b,a) (Fig. 1(b)), and plain ViT-based encoder with a pyramidal decoder (Mishra et al., 2021)(Fig. 1(c)). However, over-reliance on pyramidal structure (Deng and Li, 2022) lacks long-distance perception in early stages, leading to wrong results casually (see Fig. 4). Thus, some works introduce ViT partially to enhance global modeling capabilities, but their results are still unsatisfactory, and the training cost is expensive (You et al., 2022a; Mishra et al., 2021). This motivates us to explore a more powerful and efficient MUAD model while considering global interactions. Starting by abstracting current methods to a new Meta-AD concept (Fig. 1(d)) that consists of: (i) a feature *Encoder* to map the input to the latent space while compressing the spatial dimensions; (ii) a feature *Fuser* to fuse multi-stage feature maps and generate more compact features as the decoder input; (iii) a feature *Decoder* to reconstruct the original image or features by constraining the reconstruction loss. Existing methods (Liu et al., 2024; Deng and Li, 2022; You et al., 2022a) typically introduce pyramidal networks as encoders or decoders to obtain multi-resolution features for more accurate anomaly locations. Nevertheless, plain ViT serves as a visual foundation model that has been effective for possessing long-distance modeling capabilities in many downstream tasks, *e.g.*, object detection (Li et al., 2022), semantic segmentation (Zhang et al., 2022b), and human pose estimation (Xu et al., 2022). This motivates us to explore the feasibility of exploring only plain ViT features for the anomaly detection field.

Within the Meta-AD framework, we propose an efficient ViT-based AD model that only includes global modeling without a pyramidal encoder and decoder. Notably, we adopt a 12-layer ViT-S evenly divided into four stages to supply multi-depth features like ViTDet (Li et al., 2022). However, directly applying the naive ViT to Meta-AD leads to extremely poor anomaly classification and localization results (see Section 4.3.1). This leads to subsequent explorations to find an effective ViT structure for the MUAD task. Adhering to Occam's Razor principle, we improve plain ViT from global and local perspectives to explore its anomaly detection capability without resorting to complex modules or training strategies. *From a global perspective*, we empirically design three structural improvements that notably boost the model performance: (i) A plain ViT with the same resolution shortens the information flow path, causing the output to be identical to the input that leads to performance degradation. Removing multi-step skip connections and only using the last stage to reduce the loop between input and output improves performance, this manner forces the model to learn more essential feature reconstruction. (ii) Features at different levels can express fine-grained features differently. Unlike current methods that reconstruct features at different resolutions, we explore the impact of columnar feature reconstruction layers with the same resolution on the AD model. The last three stages can provide rich multi-depth information, and using them for loss constraints and anomaly map calculations enables a higher anomaly localization capability of the model. This configuration differs from the default usage of the first three stages in current works (Deng and Li, 2022; You et al., 2022a). ((iii) Due to the gap between AD datasets and ImageNet (Deng et al., 2009), the applicability of pretrained weights obtained through different training methods varies. Considering that ViTAD is based on reconstruction, using ImageNet-1K (Deng et al., 2009) pretrained discriminative features for the backbone would result in the loss of too many reconstructed details, leading to model failure. Therefore, we experimentally select self-supervised DINO features, which possess more general and powerful feature extraction capabilities.

From a local perspective, we explore four factors to improve the model performance further: (i) whether the output for the encoder goes through the final batch normalization. (ii) whether the feature fuser uses linear feature transformation. (iii) whether the class token is inherited. iv) whether position embeddings are used in the decoder.

Extensive experimental results reveal the effects of the proposed modules with three main interesting findings in Section 4.3. (i) Pyramidal structure for encoder/decoder is **not** necessary for AD models.

The simple plain ViT can yield impressive state-of-the-art (SoTA) results (Section 4.2). (ii) The pre-training weights and model scale of the encoder significantly impact the results, and their performance in classification tasks does not consistently correlate with AD results (Section 4.3.1). (iii) A heavy feature fuser is not necessary that a superficial linear layer suffices, which contradicts the design conclusion of previous works (Mathian et al., 2022; Deng and Li, 2022) (Section 4.3.2).

We make the following three contributions in this work:

- We present a novel ViTAD model inspired by Meta-AD, which explores pure plain ViT as the fundamental structure. Specifically, ViTAD is effectively designed from global and local perspectives (Section 3.3). In addition, we provide in-depth discussions and present some interesting findings.
- We replicate and benchmark state-of-the-art methods for fair comparisons of the MUAD task. We propose using eight metrics to comprehensively evaluate different approaches (Section 4.1).
- Extensive experimental evaluations with state-of-the-art methods on MVTec AD, VisA, and Uni-Medical datasets demonstrate the performance, efficiency, and robustness of our ViTAD. Furthermore, we explore and ablate factors that may affect model performance (Section 4.3).

2. Related work

Visual Anomaly Detection. Numerous methods have been developed to identify abnormal image regions from normal ones, based on supervised, semi-supervised, and unsupervised settings (Liu et al., 2024). Supervised and semi-supervised approaches generally use synthetic data (Zavrtanik et al., 2021a) or few-shot anomaly samples (Xie et al., 2023; Jeong et al., 2023; Chen et al., 2023a,b; Cao et al., 2023; Zhang et al., 2023a; Hu et al., 2024; Gu et al., 2024), which can be seen as a particular case of a binary classification task. Unsupervised AD methods rely only on normal training data to distinguish abnormal regions during testing, which can usually be broadly categorized as: (i) Augmentation-based methods (Li et al., 2021; Zavrtanik et al., 2021a; You et al., 2022a; Liu et al., 2023b; Lu et al., 2023) typically involve synthesizing abnormal regions on normal images or adding anomalous information to normal features to construct pseudo supervisory signals for better one-class classification. Early work operates in the image space. CutPaste (Li et al., 2021) cuts image patches and pastes them at random locations in a larger image, while DRAEM (Zavrtanik et al., 2021a) constructs synthetic anomalies on anomaly-free images automatically to aid training. Subsequent work operates on more powerful feature levels. For instance, SimpleNet (Liu et al., 2023b) adds Gaussian noise to latent embeddings, and UniAD (You et al., 2022a) introduces feature jitter. Recently, RealNet (Zhang et al., 2024c) leverages the high-quality generation ability of the diffusion model for anomaly synthesis. (ii) Embedding-based methods map normal features to a compact space, distancing them from abnormal parts to ensure discriminative capabilities. Existing approaches are based on distribution models (Wan et al., 2022; Lei et al., 2023), teacher student models (Cao et al., 2022; Zhang et al., 2023b; Bergmann et al., 2020), and memory banks (Gu et al., 2023; Roth et al., 2022). (iii) Reconstruction-based methods generally consist of an encoder and a decoder, while some approaches incorporate an additional transformation module. Anomaly localization is achieved by measuring the discrepancy between the input and reconstructed data, and current methods can be roughly categorized into two taxonomies based on the data being reconstructed, *i.e.*, RGB images and deep features. Image-level methods (Pirnay and Chai, 2022; Liang et al., 2023; Madan et al., 2023) essentially are developed based on reconstruction of normal RGB images (Mei et al., 2018; Zavrtanik et al., 2022; He et al., 2024b). A few approaches (Schlegl et al., 2017; Akçay et al., 2019) introduce deep generative adversarial networks to learn normal manifold, while subsequent studies incorporate anomaly data augmentation strategies to enhance pseudo-supervision of the model, such as random noise generation (Deng et al., 2023), random mask generation (Zavrtanik et al., 2021b), cut paste operation (Liang et al., 2023), panel-guided anomaly synthesis (Zhao, 2023), and DRAEM-like predefined anomaly data construction schemes (Zavrtanik et al., 2021a). Feature-level methods reconstruct more expressive deep features that generally achieve better performance (You et al., 2022b; Tien et al., 2023). UniAD (You et al., 2022a) corroborates the pivotal function of query embedding in circumventing shortcut feature-level distribution and further introduces a layer-wise query decoder to model feature-level distribution. On the other hand, RD (Deng and Li, 2022) introduces a novel "reverse distillation" paradigm to reconstruct multi-resolution features, which can be viewed as the extension of multi-resolution representations. Recently, InvAD (Zhang et al., 2024b) has introduced the high-precision reconstruction approach of GAN inversion into anomaly detection and achieved promising results. However, this method involves a high number of parameters and computational complexity. DiAD (He et al., 2024b) is the first to propose feature reconstruction in the latent space of SD, and MambaAD (He et al., 2024a), on the other hand, has achieved impressive results by utilizing the mamba (Gu and Dao, 2023) architecture for the first time. While augmentation-/embedding-based methods achieve satisfactory results, incorporating additional anomaly-related operations leads to complex and high-dimensional embeddings (Liu et al., 2023a) or noise-sensitive models (Jiang et al., 2022). Capitalizing on the simplicity and effectiveness of the reconstruction scheme, we propose an effective and efficient ViT-based method for unsupervised anomaly detection.

Multi-class Unsupervised Anomaly Detection. Most existing methods require individual models for training each category (Roth et al., 2022; Zavrtanik et al., 2021a; Deng and Li, 2022; Liang et al., 2023; Tien et al., 2023; Liu et al., 2023b), *a.k.a.*, Single-class Unsupervised Anomaly Detection (SUAD), which is unsuitable for practical applications. In contrast, UniAD (You et al., 2022a) employs one unified model to cover multiple categories, *a.k.a.*, Multi-class Unsupervised Anomaly Detection (MUAD). A few MUAD methods (Zhao, 2023; You et al., 2022a) have since then been developed. Our method is also formulated within this challenging but practical setting.

Pyramidal Architecture for Anomaly Detection. Numerous methods have demonstrated that multi-resolution (pyramidal) features are effective for anomaly detection (Bergmann et al., 2019b; Liang et al., 2023; Roth et al., 2022; Deng and Li, 2022; You et al., 2022a; Liu et al., 2023b). As these approaches generally include a heavy backbone and local model with a smaller receptive field, large-scale and long-distance defects may not be well detected (Roth et al., 2022; Deng and Li, 2022; Liu et al., 2023b). Recently, a few methods have used a dynamic ViT (Dosovitskiy et al., 2021) to model the global dependence of visual features to improve performance. However, they retain the pyramidal network in the encoder (You et al., 2022b,a; Zhang et al., 2023b) or decoder (Mishra et al., 2021; Zhang et al., 2023b). Unlike these methods, this inspires us to explore solely using pure plain and columnar ViT for the MUAD task, expecting to leverage its long-distance dependence and strong modeling ability verified in other fields (Dosovitskiy et al., 2021; Kirillov et al., 2023; Li et al., 2022).

Plain Vision Transformer. Benefiting from global dynamic modeling capabilities, columnar plain ViT (Dosovitskiy et al., 2021) offers more excellent usability and practical values compared to the more complex pyramidal structures. Although anomaly detection also potentially benefits from this capability, columnar ViT is only exploited in the encoder (Mishra et al., 2021) or decoder (You et al., 2022b,a) of existing methods. This inspires us to break the mold, representing the first exploration of the potential and application value of plain ViT in the challenging MUAD task. In addition, re-trained ViT features have significant effects on downstream tasks. Except for supervisorily-trained on ImageNet (Deng et al., 2009), numerous unsupervised pre-training methods (He et al., 2020; Caron et al., 2021; Oquab et al., 2024;

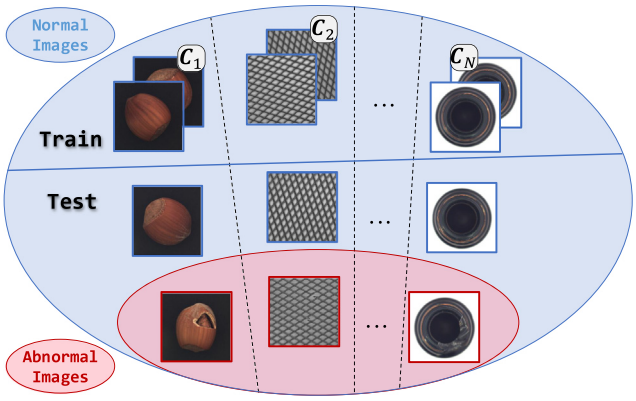


Fig. 2. Diagram of the studied Multi-class Unsupervised AD setting described in Section 3.1. The unified model uses all N classes of normal data $\boldsymbol{C} = \{C_1, C_2, \ldots, C_N\}$ for training, and distinguishes between normal and anomalous data for all classes during testing. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Radford et al., 2021; He et al., 2022) endow ViT with different feature extraction distributions, potentially affecting the effectiveness of AD models. This paper explores the effects of different pre-training manners and shows that DINO-based features perform best for MUAD.

Efficient Network Design for Anomaly Detection. Existing methods pay more attention to accuracy rather than model efficiency. For instance, DRAEM (Zavrtanik et al., 2021a) contains 97.4M parameters and requires 19.6 GPU hours under the condition of 300 epochs of training. It takes 1000 epochs for UniAD (You et al., 2022a) to converge to satisfactory results, and RD (Deng and Li, 2022) requires a large parameter count of 80.6M due to its use of a pyramidal WideResNet-50 encoder/decoder. Recent EfficientAD (Batzner et al., 2024) proposes a lightweight feature extractor to reduce the model cost, but its performance is not satisfactory and needs further improvement for the MUAD setting. Thanks to the simplicity of the basic transformer block, the small-scale ViT has competitive running efficiency. In contrast to counterparts, we use ViT-S (Dosovitskiy et al., 2021) as the backbone, which is more compact and effective. It only requires training of 100 epochs with 1.1 GPU hours and performs favorably against state-of-the-art schemes.

3. Methodology: Abstract then instantiate

3.1. Task definition of MUAD

Similar to tasks such as object classification (Deng et al., 2009) and detection (Lin et al., 2014), a more valuable anomaly detection setting should involve a model handling multiple categories simultaneously. Given an AD dataset that contains N classes $\boldsymbol{C} = \{C_1, C_2, \ldots, C_N\}$, the Single-class Unsupervised AD (SUAD) setting uses only one class C_i $(i = 1, 2, \ldots, N)$ that contains one-class set $\chi_i = \{(\chi^{Train}_{i,normal}), (\chi^{Test}_{i,normal}, \chi^{Test}_{i,anomaly})\}$, while the Multi-class Unsupervised AD (MUAD) setting covers all classes $\boldsymbol{C}$ that contain all-class sets $\chi = \sum_{i=1}^{N} \chi_i$ in one unified model. As shown in Fig. 2, the normal images of all classes *marked in* ***blue*** are used for training simultaneously without any extra labeled samples, *i.e.*, *no anomalous and defective images marked in* ***red*** are used. On the other hand, normal and abnormal samples are used together for performance evaluation. This challenging task is described in UniAD (You et al., 2022a) and adopted in recent works (Yao and Wang, 2022; Zhao, 2023). To avoid confusion, we use image-level detection and pixel-level location to refer to classification and segmentation as used in prior works.

3.2. Formulation of Meta-AD

As shown in Fig. 1, we abstract existing reconstruction-based approaches by a meta framework that contains a feature *Encoder*, a *Fuser*, and a *Decoder*. Fig. 3-Left details the specific process of its structural treatment and symbols.

***(1)* Feature Encoder.** This sub-module maps the input stem feature map F_0 to multi-depth deep features, which usually uses a frozen network trained on ImageNet (Deng et al., 2009). The encoder ϕ^E consists of N stages $\{\boldsymbol{\phi}^E_1, \boldsymbol{\phi}^E_2, \ldots, \boldsymbol{\phi}^E_N\}$ with each containing a number of basic blocks This module can be either a pyramidal structure with a decreasing resolution (*e.g.*, ResNet (He et al., 2016) and EfficientNet (Tan and Le, 2019)) or a columnar structure with a consistent resolution (*e.g.*, ViT (Dosovitskiy et al., 2021)); For the local component, it can be a CNN (Deng and Li, 2022) or a transformer (Chen et al., 2022). Pyramidal structures (Deng and Li, 2022; You et al., 2022a; Chen et al., 2022) are more commonly used for the strong ability to extract rich multi-depth features $F = \{F_0, F_1, \ldots, F_N\}$. Denote ith feature extracted by the encoder $\boldsymbol{\phi}^E_i$ as $F_i \in \mathcal{R}^{C_i \times H_i \times W_i}$, where C_i, H_i, and W_i represent channel, height, and weight, respectively. The encoding process can be represented as follows:

$$F_i = \boldsymbol{\phi}^E_i(F_{i-1}), \quad i = 1, 2, \ldots, N, \tag{1}$$

where the resolution of ϕ^E_i equals ϕ^E_{i-1} if columnar encoder is employed, and vice versa.

***(2)* Feature Fuser.** This module integrates multi-layer features $\{F_0, F_1, \ldots, F_N\}$ from different stages and generates a more compact feature for the decoder. There are two main designs for this module: (i) a light structure to fuse multi-depth features efficiently (You et al., 2022a); and (ii) a heavy structure to obtain better representation (Deng and Li, 2022). Without loss of generality, the Fuser $\mathcal{F}$ takes multi-layer features as the input to obtain the fused feature $\hat{F}_f$, denoted as:

$$\hat{F}_f = \mathcal{F}(\theta_0(F_0), \theta_1(F_1), \ldots, \theta_N(F_N)), \tag{2}$$

where θ_i is used to adjust ith feature to the desired size for subsequent fusion processing, *e.g.*, up-sampling and de-convolution operations. Specifically, θ_i does not change the resolution of the ith feature in the case of the columnar encoder. The model degenerates into an *Auto Encoder* when only the last compressed feature F_N is used, *i.e.*, $\hat{F}_f$ equals $\hat{F}_N$.

***(3)* Feature Decoder.** It reconstructs original images or encoded features from the fused feature $\hat{F}_f$. Similar to the encoder, the decoder ϕ^D can be a pyramidal/columnar structure. A pyramidal CNN is the most commonly used structure because the fused feature usually needs to be up-sampled to match the resolution of encoded features for accurate anomaly localization. The reconstruction process can be formulated as follows:

$$\hat{F}_{i-1} = \boldsymbol{\phi}^D_i(\hat{F}_i), \quad i = 1, 2, \ldots, N, \tag{3}$$

where the resolution of ϕ^D_i equals ϕ^D_{i-1} if columnar encoder is employed, and vice versa.

***(4)* Anomaly Map Estimation.** Reconstruction-based AD methods assume that anomalies cannot be well reconstructed by features extracted from normal images during the training process. For the ith stage, the anomaly map A_i is computed by:

$$A_i = \mathcal{L}_i(F_i, \hat{F}_i), \quad i = 1, 2, \ldots, N, \tag{4}$$

where $\mathcal{L}_i$ can be L1, Mean Square Error (MSE), Cosine Distance, and other metrics. During the training phase, $A = \Sigma^N_i A_i$ is generally used as the loss to backward the gradient, while $A = \Sigma^N_i A_i$ serves as the anomaly map for testing. Nevertheless, some reconstruction-based works employ extra losses (Tien et al., 2023; Liang et al., 2023) for further improvement, but they suffer from complicated modeling and implementation issues. In this paper, our work uses only one simplest pixel-level loss sufficient for Meta-AD.

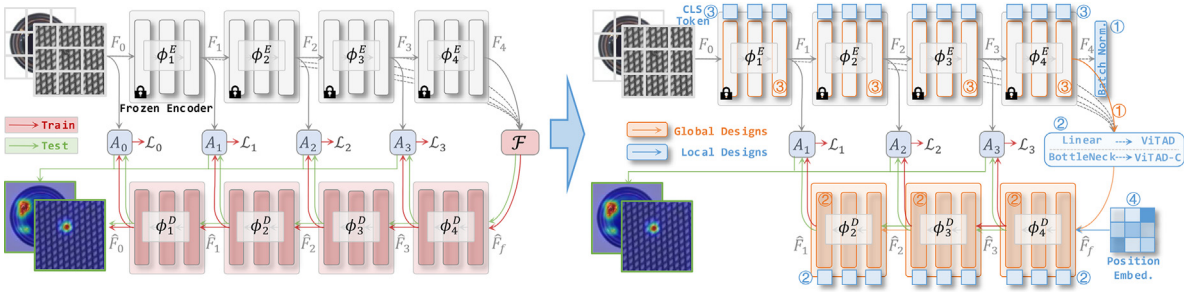


Fig. 3. Left: Reconstruction-based Meta-AD paradigm, which consists of a pretrained image encoder ϕ^E to obtain features at different depths from the patch embedding input, a feature fuser $\mathcal{F}$ to aggregate extracted multiple features, and a decoder ϕ^D that has the same structure with the encoder to reconstruct multi-depth features. During the training phase, $\hat{F}_i$ is constrained by F_i with loss function $\mathcal{L}_i$ to update ϕ^D, while both $\hat{F}_i$ and F_i are used to calculate anomaly map A_i for inference. **Right: Detailed structure of instantiated ViTAD from Meta-AD** that contains global and local designs over MetaAD. **Orange** and **blue** numbers indicate the adaptations of the plain ViT at the global and local levels for MUAD. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

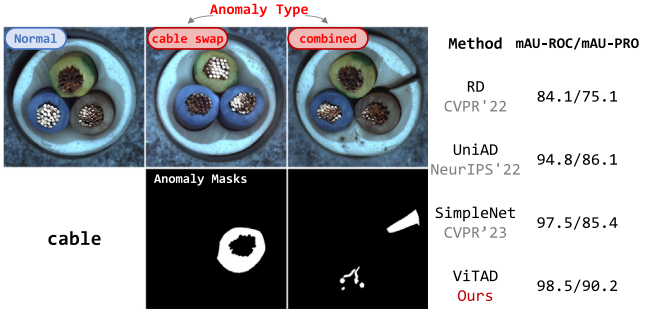


Fig. 4. Left: Pilot study for the necessity of global dependence on "cable" category in MVTec AD dataset, *e.g.*, logically-dependent "cable swap" and long-distance dependent "combined" defects. Right: Quantitative evaluation results of different methods on this category. Our ViTAD markedly mitigates these challenges over other methods by leveraging ViT's global modeling ability.

***(5)* Generalized Extension.** In the generalized concept, encoder features can skip directly to multiple decoder stages without going through the Fuser (Akçay et al., 2019). Nevertheless, this manner tends to fall into an "identity shortcut" that appears to return an unmodified input, disregarding its input content. However, it can be mitigated through specific training strategies and data augmentation methods (You et al., 2022a). Nevertheless, this paper does not discuss this manner, but the abstracted Meta-AD framework can be easily extended to include that case. In addition, effective plug-and-play defect construction (Cimpoi et al., 2014; Zavrtanik et al., 2021a), data augmentation (Liu et al., 2023b), and loss functions (Li et al., 2021; Tien et al., 2023) techniques can potentially enhance performance further.

3.3. Plain vision transformer for AD

3.3.1. Motivation for exploring plain ViT for the MUAD task

Thanks to the global modeling capability and powerful feature representations, ViT exhibits remarkable achievements for diverse scenes, which has been proven effective and applied in various domains, *i.e.*, object detection (Li et al., 2022), semantic segmentation (Kirillov et al., 2023), in-context visual learning (Wang et al., 2023), and multi-modal fusion (Kim et al., 2021). Nevertheless, *using only plain ViT without additional structures has never been explored in the AD field.* Recently, some AD methods (Mishra et al., 2021; You et al., 2022b,a) have introduced powerful ViT as part of the model (*e.g.*, encoder or decoder) to improve performance. Although some progress has been made, *relying on only a portion of global ViT modeling still leads to misdetection of certain types of defects like previous methods without ViT*, *e.g.*, logical errors ("cable swap") and shortcomings in long-distance interactions ("combined") in cable category on MVTec AD (Bergmann et al., 2019a) in Fig. 4. This inspires us to explore the feasibility of using pure ViT for AD tasks. Specifically, we take a pair of normal and abnormal images for a toy experiment shown in Fig. 5. The ViT features are more prosperous and diversified than those from CNN at each stage, and the difference between normal and abnormal images is more significant. The phenomenon demonstrates the stronger modeling capability and larger receptive field of ViT structure for the potential applications to the AD task.

3.3.2. Progressive design of ViTAD

As multi-resolution features are necessary to model anomaly location accurately, existing AD methods use pyramidal networks in encoder (You et al., 2022b,a) or decoder (Mishra et al., 2021; Zhang et al., 2023b) to extract multi-resolution features. In contrast, we develop a non-pyramidal plain yet effective ViTAD for the MUAD task from Meta-AD step by step.

Based on the formulation of Meta-AD, we employ columnar plain ViT as the instantiated structure and divide it into four stages as pyramid RD (Deng and Li, 2022) for both encoder $\{\boldsymbol{\phi}_1^E, \boldsymbol{\phi}_2^E, \boldsymbol{\phi}_3^E, \boldsymbol{\phi}_4^E\}$ and decoder $\{\boldsymbol{\phi}_1^D, \boldsymbol{\phi}_2^D, \boldsymbol{\phi}_3^D, \boldsymbol{\phi}_4^D\}$. Each stage contains the same number of layers, *i.e.*, 3 layers for each stage with 12-layer ViT-B trained on ImageNet-1K, and the decoder is randomly initialized with the same ViT structure. For the feature Fuser $\boldsymbol{F}$, we employ one superficial linear layer to aggregate concatenated multi-stage features with the same resolution to ensure that the channel number of $\hat{F}_f$ matches the decoder input:

$$\hat{F}_f = \boldsymbol{\mathcal{F}}(F_1, \ldots, F_4) = \text{Linear}([F_1, \ldots, F_4]), \tag{5}$$

where θ_i in Eq. (2) degenerates into identity operation for simplicity. Nevertheless, a naive implementation yields a significant gap compared to the state-of-the-art methods for the MUAD task, *e.g.*, the image-level and pixel-level mAU-ROC metrics are only 92.4/95.9, which are significantly lower than the SoTA UniAD of 97.5/97.0. Thus, we make AD-specific improvements to the implementation details at both global and local levels in Fig. 3-Right, ultimately achieving obvious superiority over SoTA pyramidal approaches (Section 4.2).

Global Design of ViTAD. As shown by the **orange** numbers in Fig. 3-Right, we explore three effective designs at the global level for the ViT-based AD model. First, the Fuser removes the multi-depth feature skip connection and only uses the last stage F_4 as input, *i.e.*, $\hat{F}_f = \text{Linear}(F_4)$ for Eq. (5). This modification structurally reduces the identity short path described in UniAD (You et al., 2022a), thereby significantly improving model performance (see Table 8), *e.g.*, the mAD increases from 84.7 to 85.4. The reason is that the deep features F_4 of the columnar ViT are sufficient to contain rich texture and semantic attributes. The injection of shallow stage features would shorten the information flow path that constructs a shortcut for feature transformation, leading to potential information leakage that enables

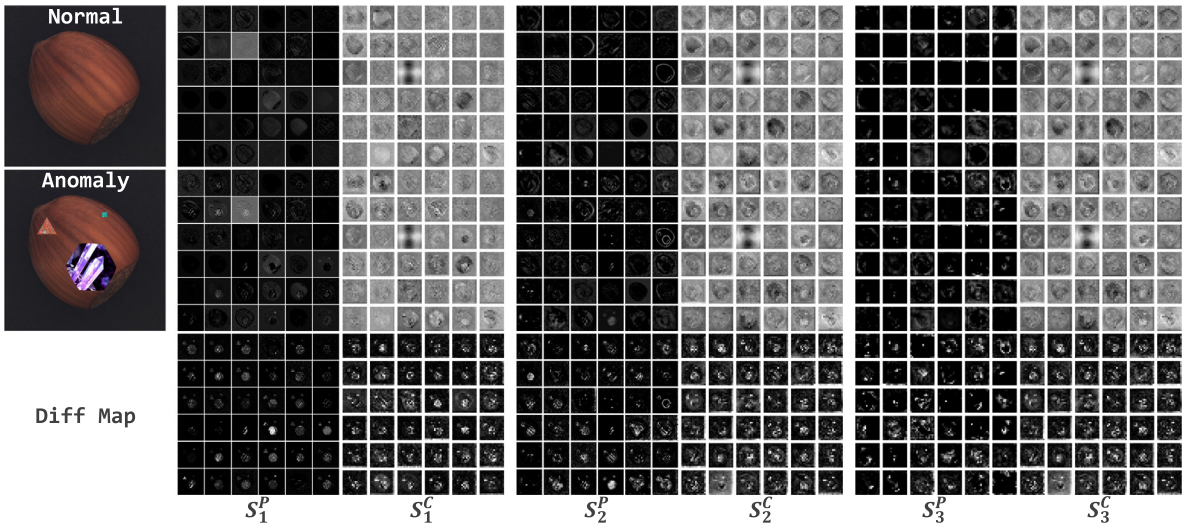


Fig. 5. First 36 visualized feature maps of different stages (S_i, $i = 1, 2, 3$) **for** *pyramid* (Deng and Li, 2022) (S^P) **and** *columnar* (Dosovitskiy et al., 2021) (S^C) **backbones.** The first two rows show the results of normal and anomaly images in the first column, and the last row shows differential maps. Results demonstrate that ViT excels at capturing a wider range of features and identifying more distinct anomalous areas.

the model to learn the identity mapping (Deng and Li, 2022; You et al., 2022a) and affecting the model's judgment ability at the image level. Using only the last stage reduces the loop between input and output, forcing the model to learn more essential feature reconstruction. Furthermore, we observe that a heavy fuser is not necessary and a superficial linear layer suffices, which contradicts the design conclusion of previous works (Mathian et al., 2022; Deng and Li, 2022) (Section 4.3.2). Second, features at different levels can express fine-grained features differently, as demonstrated by some works in the AD field. Unlike current reconstruction-based methods (Deng and Li, 2022; He et al., 2024a) that recover features at different resolutions, we explore the impact of columnar feature reconstruction layers with the same resolution on the AD model. In this work, we use $F_1/F_2/F_3$ for training constraints and compute anomaly maps $A_1/A_2/A_3$ during inference. (see Table 14) This configuration differs from the default usage of the first three stages in works (Deng and Li, 2022; You et al., 2022a), providing richer multi-depth information to produce more accurate anomaly localization. Third, existing AD methods would use ImageNet-1K (Deng et al., 2009) pretrained vision backbone as part of the model by default. However, this straightforward approach performs poorly due to the domain gap between ImageNet-1K and downstream AD datasets. We analyze various self-supervised training schemes, empirically observing that pretrained weights from more general unsupervised training could mitigate this gap. DINO (Caron et al., 2021) has stronger semantic granularity and performs better than other pre-training methods (see Table 10). Using pretrained DINO weights stems from the fact that current unsupervised AD methods heavily rely on pretrained weights, and the applicability of pretrained weights obtained through different training methods varies. Considering that ViTAD is based on reconstruction, using ImageNet-1K (Deng et al., 2009) pretrained discriminative features for the backbone would result in the loss of too many details, leading to model failure. Therefore, we empirically select self-supervised DINO features, which possess more general and powerful feature extraction capabilities (detailed ablation study in Table 10).

Local Design of ViTAD. To further improve performance, we empirically analyze four finer structural details that improve the performance, as indicated by the **blue** numbers in Fig. 3-Right. First, using features before normalization, rather than after this operation, as the fusion features slightly decrease image-level performance. The above modification is empirical structural improvements aimed at further enhancing the model's performance in finer details. Second, we use a lightweight linear one-layer as the Fuser structure that discards heavy transformations. When we remove this linear layer, *i.e.*, $F_4 = \hat{F}_4$, it leads to performance loss, indicating the necessity of one linear layer. Third, maintaining the class token throughout the process slightly increases computational load and potentially affects the interaction between normal spatial structures, which causes a slight performance degradation. Thus, this modification enhances the modeling of spatial feature interactions while reducing the influence of the global CLS token on the model. Therefore, we remove the class token, similar to previous works (Liu et al., 2021; Zhang et al., 2023c) in downstream task experiments. Fourth, maintaining the position embedding of the ViT-based decoder also brings slight performance gain. Detailed ablation study can be viewed in Table 12.

3.3.3. ViTAD summary

Starting from the abstract Meta-AD framework, we gradually improve plain ViT from global and local perspectives to realize the adaptation to the AD task. As shown in Table 1, each design would incrementally boost the performance, and we finally obtain a very competitive AD model for the challenging MUAD setting. The pre-trained weights from DINO (Caron et al., 2021) significantly enhance model performance. However, the structure of Vanilla ViT and other global/local designs are crucial for the results. For instance, the vanilla ViT achieves 80.6 mAP on the MVTec AD dataset, surpassing DeSTseg (Zhang et al., 2023b), which has 77.1 mAP. When using ImageNet-1K pre-trained weights, the model achieves 82.4 mAP, improving by 1.8 points over the strong vanilla ViT baseline. Additionally, Table 1 demonstrates that each global and local design represents incremental improvements based on modifications to the original vanilla ViT structure. Replacing with global DINO weights does not alter the model's parameters and FLOPs. The other two designs reduce model complexity while enhancing performance. Each component in the local designs has a negligible computational expense, less than 0.1, with the model parameters totaling 38.6M and FLOPs at 10.7G. We use the entry at the end of Table 1 as our final Meta-AD instantiation termed ViTAD, the first ViT-based powerful model customized for the AD domain. The detailed structure can be viewed in the attached source code.

3.3.4. Training constraints

We aim to explore the effectiveness of plain ViT for the AD task and provide a strong baseline to facilitate future research. Therefore, we use only the cosine distance loss during training.

Table 1
Overall incremental trajectory from naive ViT-adapted Meta-AD to ViTAD. Each line is based on a modification of the immediately preceding line with seven metrics as well as their average. We also report parameter and FLOPs for each model. Detailed ablations in Section 4.3.

	Method	Image-level			Pixel-level			mAU-PRO	mAD	Parameters	FLOPs
		mAU-ROC	mAP	mF_1-max	mAU-ROC	mAP	mF_1-max				
	Baseline	92.4	96.4	94.2	95.9	46.9	51.8	86.5	80.6	44.2M	12.3G
Global	+ Remove multi-depth features	93.3	96.8	94.8	96.1	47.6	52.6	86.9	81.2	43.9M	12.2G
Global	+ Use late three stage features	93.8	97.3	95.2	96.4	48.6	53.6	87.6	81.8	44.2M	12.3G
Global	+ Use pretrained DINO weights	97.6	99.0	96.8	97.5	55.0	58.2	91.0	85.0	38.6M	10.7G
Local	+ Before normalization	97.9	99.1	97.0	97.5	54.8	58.1	91.1	85.1	38.6M	10.7G
Local	+ Add linear	98.1	99.2	97.0	97.6	55.1	58.6	91.4	85.3	38.6M	10.7G
Local	+ Remove class token	98.0	99.1	97.2	97.7	55.3	58.5	91.3	85.3	38.6M	10.7G
Local	+ Use position embedding	98.3	99.4	97.3	97.7	55.3	58.7	91.4	85.4	38.6M	10.7G

Specifically, ViTAD uses the fused feature $\hat{F}_f$ to reconstruct multiple features $F_i \in \mathcal{R}^{C_i \times H_i \times W_i}$ by the predicted $\hat{F}_i \in \mathcal{R}^{C_i \times H_i \times W_i}$ with decoder. Let $F_i(h,w) \in \mathcal{R}^{C_i}$ and $\hat{F}_i(h,w) \in \mathcal{R}^{C_i}$ be the ith stage feature vector at position (h,w) in both encoder and decoder, we use the cosine distance at position (h,w) as the anomaly score $A_i(h,w)$ (Deng and Li, 2022):

$$A_i(h,w) = 1 - \frac{F_i(h,w)^T \cdot \hat{F}_i(h,w)}{\|F_i(h,w)\| \|\hat{F}_i(h,w)\|}, i \in \{1,2,3\}. \tag{6}$$

Anomaly scores of all positions construct the final anomaly map. Since the model is only trained on normal samples, the large pixel values in the anomaly map suggest anomalies. For the ith stage, the corresponding loss term $\mathcal{L}_i$ is calculated by:

$$\mathcal{L}_i = \frac{1}{H_i W_i} \sum_{h=1}^{H_i} \sum_{w=1}^{W_i} A_i(h,w). \tag{7}$$

The overall loss $\mathcal{L}_{All} = \Sigma_1^3 \mathcal{L}_i$ is the simple sum of all $\mathcal{L}_i$ terms.

3.3.5. Anomaly segmentation and classification

Anomaly segmentation aims to provide the pixel-level anomaly score map to determine the specific locations of the anomalies. multi-depth anomaly maps $\{A_1, A_2, A_3\}$ calculated by the restrained features are summed up to form the final anomaly map $A = \text{sum}(A_1, A_2, A_3)$ for evaluation. In addition, anomaly classification requires an image-level anomaly score to indicate whether the image is anomalous. Similar to You et al. (2022a), we first apply an average pooling operation to the anomaly score map A and then take its maximum value as the anomaly score.

4. Experiments

4.1. Setups for multi-class unsupervised AD

Task Setting. The single-class setting usually requires training a model for each class separately. This work focuses on the more challenging and practical multi-class setting described in Section 3.1. Experiments for ablation and interpretability are primarily conducted on the MVTec AD dataset.

Datasets. We evaluate ViTAD and state-of-the-art methods on the widely-used industrial MVTec AD (Bergmann et al., 2019a, 2021) and VisA (Zou et al., 2022) datasets for anomaly classification and segmentation: The MVTec AD dataset contains 15 industrial products in 2 types, with 3629 normal images for training and 467/1258 normal/anomaly images for testing (5354 images in total); The VisA dataset covers 12 objects in 3 types, with 8659 normal images for training and 962/1200 normal/anomaly images for testing (10,821 images in total). In addition, we unify Brain MRI (7500 normal images for training and 640/3075 normal/anomaly images for testing), Liver CT (1542 normal images for training and 660/1258 normal/anomaly images for testing), and Retinal OCT (4297 normal images for training and 1041/764 normal/anomaly images for testing) to establish a Uni-Medical (Bao et al., 2023) benchmark in the medical field. All datasets provide ground-truth anomaly maps at the pixel level for evaluation.

Evaluation Metrics for AD. Similar to Deng and Li (2022), Zavrtanik et al. (2021a), Zou et al. (2022) and Bergmann et al. (2020), we use threshold-independent measures, including mean Area Under the Receiver Operating Curve (mAU-ROC) to evaluate binary classification ability, mean Average Precision (Zavrtanik et al., 2021a) (mAP) to calculate the area under the PR curve, and mean Area Under the Per-Region-Overlap (Bergmann et al., 2020) (mAU-PRO) to weigh regions of different size equally. In addition, threshold-dependent mean F_1-score at optimal threshold (Zou et al., 2022) (mF_1-max) is employed to relieve the potential data imbalance problem. Note that mAU-ROC, mAP, and mF_1-max are used in image-level (anomaly classification) and pixel-level (anomaly segmentation) evaluations. The maximum pixel-level value is regarded as the image-level anomaly score (Deng and Li, 2022; You et al., 2022a). The models are evaluated ten times evenly for all methods, and the result corresponding to the maximum pixel-level mAU-ROC value is taken as the final result. We demonstrate and emphasize using all metrics for evaluation, and the mean of all the metrics termed mAD is reported to indicate the overall performance.

Comparison Methods. As MUAD is a relatively new task, we mainly evaluate the published UniAD methods (You et al., 2022a). We also compare with the latest Augmentation-based DRAEM (Zavrtanik et al., 2021a), Reconstruction-based RD (Deng and Li, 2022), and Embedding-based DeSTSeg (Zhang et al., 2023b)/SimpleNet (Liu et al., 2023b). Since the above methods only report results under the SUAD setting, we retrain them to obtain MUAD results by official codes. In addition, we replace the ViT (Dosovitskiy et al., 2021) encoder of our ViTAD with various pretrained manners to ensure fair and comprehensive evaluations, including pretrained models by ImageNet (Deng et al., 2009) and self-supervised MoCo (He et al., 2020), MAE (He et al., 2022), CLIP (Radford et al., 2021), DINOv2 (Oquab et al., 2024), and DINO (Caron et al., 2021). We use five criteria to compare current concurrent to show differences among them clearly. As shown in Table 2, comparison methods more or less require (1) pyramidal encoder, (2) heavy fuser to aggregate multi-depth features, (3) pyramidal decoder, (4) multi-resolution features to keep accurate anomaly location ability, and (5) feature augmentation to obtain better results. The design of ViTAD is simple but effective without resorting to elaborate structure or complex augmentations.

Training. ViTAD chooses ViT-S with DINO-S weights as the structure of the encoder and decoder, and it is trained on images of 256×256 pixels without other data augmentations for all experiments. The AdamW optimizer (Loshchilov and Hutter, 2019) is used with an initial learning rate of $1e^{-4}$, a weight decay of $1e^{-4}$, and a batch size of 8. Our model only requires 100 training epochs on a single GPU in all experiments, and the learning rate drops by 0.1 after 80 epochs. All images are trained together without any categorical labels for the MUAD setting, and the weights of the encoder are frozen by default as previous methods (Deng and Li, 2022; You et al., 2022a; Liu et al., 2023b). Note that the above hyperparameters are fixed in all experiments under different settings and datasets for our approach without elaborate fine-tuning.

Table 2
Comparison for current powerful anomaly detection methods in terms of different characteristics. ✔: Satisfied; ✘: Unsatisfied; ✚: Partially satisfied; ○: Inapplicable. Aug., Emb., and Rec. respectively represent augmentation-based, embedding-based, and reconstruction-based methods..

Criterion → / Method ↓	Pyramidal Encoder	Heavy Fuser	Pyramidal Decoder	Multi-Resolution Features	Image/Feature Augmentation	Category			Reproduction Code
						Aug.	Emb.	Rec.	
DRAEM (Zavrtanik et al., 2021a)	✔	✘	○	✔	✔	✔	✚	✚	github.com/VitjanZ/DRAEM
RD (Deng and Li, 2022)	✔	✔	✔	✔	✘	✘	✘	✔	github.com/hq-deng/RD4AD
UniAD (You et al., 2022a)	✔	✘	✘	✔	✔	✔	✘	✔	github.com/zhiyuanyou/UniAD
DeSTSeg (Zhang et al., 2023b)	✔	✘	○	✔	✔	✔	✔	✘	github.com/apple/ml-destseg
SimpleNet (Liu et al., 2023b)	✔	✘	○	✔	✔	✔	✔	✘	github.com/DonaldRR/SimpleNet
RealNet (Zhang et al., 2024c)	✔	✔	○	✔	✔	✔	✚	✚	github.com/cnulab/RealNet
MambaAD (He et al., 2024a)	✔	✔	✔	✔	✘	✘	✘	✔	github.com/lewandofskee/MambaAD
ViTAD (Ours)	✘	✘	✘	✘	✘	✘	✘	✔	github.com/zhangzjn/ADer

Table 3
Image-level multi-class anomaly classification results with mAU-ROC/mAP/mF_1-max metrics on MVTec AD. Note that only one model is trained to detect anomalies for all categories. **Superscript *: Re-trained under the multi-class setting by the official code. †: Reproduced results no less than the original paper. *Bold*** and ***Underline*** indicate optimal and sub-optimal results, respectively. The circled number represents the category to which the method belongs (*c.f.*, Section 2): ①-Image/Feature augmentation based; ②-Embedding based; ③-Image/Feature Reconstruction based, and the **red** circle indicate the main category to which the method belongs. ***Subsequent tables follow the consistent presentations.***.

Method →		① ② ③ DRAEM* (Zavrtanik et al., 2021a)	③ RD* (Deng and Li, 2022)	① ③ UniAD† (You et al., 2022a)	① ② DeSTSeg* (Zhang et al., 2023b)	① ② SimpleNet* (Liu et al., 2023b)	① ② RealNet* (Zhang et al., 2024c)	③ MambaAD* (He et al., 2024a)	③ ViTAD
Category ↓		ICCV'21	CVPR'22	NeurIPS'22	CVPR'23	CVPR'23	CVPR'24	NeurIPS'24	(Ours)
Texture	Carpet	97.2/99.1/96.7	98.5/99.6/97.2	**99.8/99.9/99.4**	97.6/99.3/96.6	95.9/98.8/94.9	96.5/99.1/96.0	**99.8/99.9/99.4**	99.5±.00/**99.9**±.00/**99.4**±.00
	Grid	99.2/99.7/98.2	98.0/99.4/96.6	99.3/99.8/**99.1**	97.9/99.2/96.6	97.6/99.2/96.4	97.2/99.2/96.4	**99.8/99.9/99.1**	99.7±.14/**99.9**±.05/**99.1**±.50
	Leather	97.7/99.3/95.0	**100./100./100.**	**100./100./100.**	99.2/99.8/98.9	**100./100./100.**	**100./100./100.**	**100./100./100.**	**100.**±.00/**100.**±.00/**100.**±.00
	Tile	**100./100./100.**	98.3/99.3/96.4	99.9/99.9/99.4	97.0/98.9/95.3	99.3/99.8/98.8	97.5/99.3/97.6	96.8/98.8/93.8	**100.**±.00/**100.**±.00/**100.**±.00
	Wood	**100./100./100.**	99.2/99.8/98.3	98.9/99.7/97.5	99.9/**100.**/99.2	98.4/99.5/96.7	99.6/99.9/99.2	98.5/99.5/96.0	98.7±.10/99.6±.03/96.7±.47
Object	Bottle	97.3/99.2/96.1	99.6/99.9/98.4	**100./100./100.**	98.7/99.6/96.8	**100./100./100.**	95.6/98.8/95.1	99.8/**100.0**/99.2	**100.**±.00/**100.**±.00/**100.**±.00
	Cable	61.1/74.0/76.3	84.1/89.5/82.5	94.8/97.0/90.7	89.5/94.6/85.9	97.5/98.5/94.7	70.0/83.0/76.0	**99.3/99.6/97.4**	98.5±.15/99.1±.08/95.7±.86
	Capsule	70.9/92.5/90.5	94.1/96.9/**96.9**	93.7/98.4/96.3	82.8/95.9/92.6	90.7/97.9/93.5	64.4/90.3/90.5	91.8/98.2/93.2	**95.4**±.46/**99.0**±.12/95.5±.05
	Hazelnut	94.7/97.5/92.3	60.8/69.8/86.4	**100./100./100.**	98.8/99.2/98.6	99.9/**100.**/99.3	**100./100./100.**	**100./100./100.**	99.8±.13/99.9±.07/98.6±.82
	Metal Nut	81.8/95.0/92.0	**100./100./99.5**	98.3/99.5/98.4	92.9/98.4/92.2	96.9/99.3/96.1	78.6/94.7/89.4	99.7/99.9/98.4	99.7±.07/99.9±.02/98.4±.30
	Pill	76.2/94.9/92.5	**97.5/99.6/96.8**	94.4/99.0/95.4	77.1/94.4/91.7	88.2/97.7/92.5	68.5/92.8/91.6	94.7/99.0/96.5	96.2±.50/99.3±.10/96.4±.18
	Screw	87.7/95.7/89.9	**97.7/99.3/95.8**	95.3/98.5/92.9	69.9/88.4/85.4	76.7/90.5/87.7	71.3/86.6/87.0	92.7/96.8/94.0	91.3±.39/97.0±.16/93.0±.88
	Toothbrush	90.8/96.8/90.0	97.2/99.0/94.7	89.7/95.3/95.2	71.7/89.3/84.5	89.7/95.7/92.3	76.4/91.5/83.3	96.4/98.5/96.8	98.9±.32/99.6±.12/96.8±.92
	Transistor	77.2/77.4/71.1	94.2/95.2/90.0	99.8/**99.8**/97.5	78.2/79.5/68.8	99.2/98.7/97.6	79.7/83.0/73.3	**99.9/99.8/98.7**	98.8±.57/98.3±.85/92.5±.72
	Zipper	**99.9/100./99.2**	99.5/99.9/**99.2**	98.6/99.6/97.1	88.4/96.3/93.1	99.0/99.7/98.3	77.3/94.1/88.1	98.2/99.5/96.7	97.6±.08/99.3±.03/97.1±.46
Average		88.8/94.7/92.0	94.6/96.5/95.2	97.5/99.1/**97.3**	89.2/95.5/91.6	95.3/98.4/95.8	84.8/94.1/90.9	97.8/99.3/**97.3**	**98.3**±.02/**99.4**±.05/**97.3**±.20

Table 4
Pixel-level multi-class anomaly segmentation with mAU-ROC/mAP/mF_1-max/mAU-PRO on MVTec AD. The last row presents the averaged mAD metric across seven metrics to provide a comprehensive evaluation.

Method →		DRAEM* (Zavrtanik et al., 2021a)	RD* (Deng and Li, 2022)	UniAD† (You et al., 2022a)	DeSTSeg* (Zhang et al., 2023b)	SimpleNet* (Liu et al., 2023b)	RealNet* (Zhang et al., 2024c)	MambaAD* (He et al., 2024a)	ViTAD
Category ↓		ICCV'21	CVPR'22	NeurIPS'22	CVPR'23	CVPR'23	CVPR'24	NeurIPS'24	(Ours)
Texture	Carpet	98.1/**78.7/73.1**/93.1	**99.0**/58.5/60.5/95.1	98.4/51.4/51.5/94.4	93.6/59.9/58.9/89.3	97.4/38.7/43.2/90.6	89.2/69.0/64.3/84.0	**99.2**/64.0/63.8/**97.3**	**99.0**±.01/60.5±.60/64.1±.22/94.7±.19
	Grid	99.0/44.5/46.2/92.1	**99.2**/46.0/47.4/**97.0**	97.7/23.7/30.4/92.9	97.0/42.1/46.9/86.8	96.8/20.5/27.6/88.6	82.6/41.2/45.7/77.7	98.9/**48.4/48.5**/96.9	98.6±.01/31.2±.16/36.7±.12/95.8±.82
	Leather	98.9/60.3/57.4/88.5	99.3/38.0/45.1/97.4	98.8/34.2/35.5/96.8	99.5/**71.5/66.5**/91.1	98.7/28.5/32.9/92.7	97.9/70.4/68.0/98.0	99.3/50.6/50.4/**98.7**	**99.6**±.01/52.1±.76/55.8±.66/97.9±.30
	Tile	**99.2/93.6/86.0/97.0**	95.3/48.5/60.5/85.8	92.3/41.5/50.3/78.4	93.0/71.0/66.2/87.1	95.7/60.5/59.9/90.6	93.9/84.1/76.8/90.5	93.0/43.9/52.6/79.5	96.6±.01/56.4±.26/68.8±.14/87.0±.44
	Wood	**96.9/81.4/74.6/94.2**	95.3/47.8/51.0/90.0	93.2/37.4/42.8/86.7	95.9/77.3/71.3/83.4	91.4/34.8/39.7/76.3	90.4/76.1/71.9/88.8	94.0/46.9/48.4/92.0	96.4±.03/60.6±.52/58.3±.39/88.0±.69
Object	Bottle	91.3/62.5/56.9/80.7	97.8/68.2/67.6/94.0	98.0/67.0/67.9/93.1	93.3/61.7/56.0/67.5	97.2/53.8/62.4/89.0	69.8/53.9/46.6/60.9	98.7/79.7/**75.7/96.0**	**98.8**±.03/**79.9**±.57/75.6±.56/94.3±.12
	Cable	75.9/14.7/17.8/40.1	85.1/26.3/33.6/75.1	**96.9/45.4**/50.4/86.1	89.3/37.5/40.5/49.4	96.7/42.4/**51.2**/85.4	61.5/23.5/25.9/33.3	95.2/42.0/47.9/**90.4**	96.2±.25/43.1±.54/47.4±.65/90.2±.74
	Capsule	50.5/ 6.0/10.0/27.3	**98.8**/43.4/**50.1/94.8**	**98.8**/45.6/47.7/92.1	95.8/**47.9**/48.9/62.1	98.5/35.4/44.3/84.5	54.6/23.7/12.1/23.4	98.3/43.5/47.8/**93.0**	98.3±.02/42.7±.58/47.8±.28/92.0±.94
	Hazelnut	96.5/**70.0**/60.5/78.7	97.9/36.2/51.6/92.7	98.0/53.8/56.3/94.1	98.2/65.8/61.6/84.5	98.4/44.6/51.4/87.4	77.5/44.2/48.9/75.4	**99.1**/67.0/**66.1/95.3**	99.0±.01/64.6±.46/64.0±.25/95.2±.08
	Metal Nut	74.4/31.1/21.0/66.4	93.8/62.3/65.4/91.9	93.3/50.9/63.6/81.8	84.2/42.0/22.8/53.0	**98.0/83.1/79.4**/85.2	52.5/32.3/21.0/39.6	**96.7**/74.2/78.3/**92.9**	96.4±.14/75.1±.52/77.3±.83/92.4±.13
	Pill	93.9/59.2/44.1/53.9	97.5/63.4/65.2/**95.8**	96.1/44.5/52.4/95.3	96.2/61.7/41.8/27.9	96.5/72.4/67.7/81.9	54.4/47.8/8.9/35.1	96.2/55.0/58.9/95.2	**98.7**±.01/**77.8**±.27/**75.2**±.23/95.3±.15
	Screw	90.0/33.8/40.7/55.2	**99.4**/40.2/44.7/96.8	99.2/37.4/42.3/95.2	93.8/19.9/25.3/47.3	96.5/15.9/23.2/84.0	51.8/15.4/4.5/18.5	**99.3/45.3/45.1/97.0**	99.0±.03/34.0±.81/41.0±.95/93.5±.53
	Tooth-brush	97.3/**55.2**/55.8/68.9	99.0/53.6/58.8/**92.0**	98.4/37.8/49.1/87.9	96.2/52.9/58.8/30.9	98.4/46.9/52.5/87.4	84.8/50.1/56.1/34.1	98.9/47.5/59.7/**92.0**	**99.1**±.04/51.3±.95/**61.9**±.50/90.9±.17
	Transistor	68.0/23.6/15.1/39.0	85.9/42.3/45.2/74.7	**97.4**/61.2/**63.0/93.5**	73.6/38.4/39.2/43.9	95.8/58.2/56.0/83.2	60.9/40.2/28.3/44.6	96.0/**63.8**/61.6/90.4	93.9±.13/58.4±.36/55.3±.42/76.8±.88
	Zipper	**98.6/74.3/69.3**/91.9	98.5/53.9/60.3/94.1	98.0/45.0/51.9/92.6	97.3/64.7/59.2/66.9	97.9/53.4/54.6/90.7	67.6/51.9/42.6/47.7	98.1/55.0/58.9/**94.5**	95.9±.07/42.6±.43/50.8±.30/87.2±.82
Average		88.6/52.6/48.6/71.1	96.1/48.6/53.8/91.2	97.0/45.1/50.4/90.7	93.1/54.3/50.9/64.8	96.9/45.9/49.7/86.5	72.6/48.2/41.4/56.8	97.4/55.1/57.6/**93.4**	**97.7**±.02/**55.3**±.11/58.7±.10/91.4±.21
mAD		76.6	82.3	82.4	77.1	81.2	69.8	**85.4**	**85.4**

4.2. Performance evaluation

Quantitative Evaluations on MVTec AD. We evaluate the ViTAD method with state-of-the-art approaches using both image-level (Table 3) and pixel-level (Table 4) metrics on the MVTec AD dataset. The proposed ViTAD method performs favorably against all the evaluated schemes, even when trained solely with the plain columnar ViT and a simple cosine similarity loss. ViTAD achieves better image-level results than UniAD with SoTA results on mAU-ROC/mAP/mF_1-max of 98.3/99.4/97.3. In addition, ViTAD achieves performance gain of +0.7 ↑/+10.2 ↑/+8.3 ↑/+0.7 ↑ using pixel-level metrics (mAU-ROC/mAP/mF_1-max of 97.7/55.3/58.7/91.4) and performance gain of +3.0 ↑ using the mean metric (85.4). Even compared to the recent MambaAD (He et al., 2024a) (accepted for NeurIPS 2024 but not yet

Table 5
Image-level multi-class anomaly classification (left) and pixel-level multi-class anomaly segmentation (right) results on VisA. Our approach yields significantly better results compared to the benchmark methods.

	Method →	**Image-level** mAU-ROC/mAP/mF_1-max				**Pixel-level** mAU-ROC/mAP/mF_1-max/mAU-PRO			
		DRAEM* (Zavrtanik et al., 2021a)	UniAD† (You et al., 2022a)	SimpleNet* (Liu et al., 2023b)	ViTAD	DRAEM* (Zavrtanik et al., 2021a)	UniAD† (You et al., 2022a)	SimpleNet* (Liu et al., 2023b)	ViTAD
	Category ↓	① ICCV'21	③ NeurIPS'22	② CVPR'23	③ (Ours)	① ICCV'21	③ NeurIPS'22	② CVPR'23	③ (Ours)
Complex Structure	PCB1	71.9/72.3/70.0	94.2/92.9/90.8	91.6/91.9/86.0	**95.8**±.22/**94.7**±.43/**91.8**±.95	94.7/31.9/37.3/52.9	99.2/59.6/59.6/88.8	99.2/**86.1**/**78.8**/83.6	**99.5**±.01/64.5±.89/61.7±.76/**89.6**±.81
	PCB2	78.5/78.3/76.3	91.1/91.6/**85.1**	**92.4**/**93.3**/84.5	90.6±.12/89.9±.36/**85.3**±.82	92.3/10.0/18.6/66.2	**98.0**/ 9.2/16.9/82.2	96.6/ 8.9/18.6/**85.7**	97.9±.06/**12.6**±.40/**21.2**±.49/82.0±.53
	PCB3	76.6/77.5/74.8	82.2/83.2/77.5	89.1/91.1/82.6	**90.9**±.25/**91.2**±.28/**83.9**±.21	90.8/14.1/24.4/43.0	**98.2**/13.3/24.0/79.3	97.2/**31.0**/**36.1**/85.1	**98.2**±.03/22.4±.87/26.4±.95/**88.0**±.71
	PCB4	97.4/97.6/93.5	99.0/**99.1**/95.5	97.0/97.0/93.5	**99.1**±.13/98.9±.18/**96.6**±.27	94.4/31.0/37.6/75.7	97.2/29.4/33.5/82.9	93.9/23.9/32.9/61.1	**99.1**±.03/**42.9**±.47/**48.3**±.51/**91.8**±.79
Multiple Instances	Macaroni1	69.8/68.6/70.9	82.8/79.3/75.7	**85.9**/82.5/73.1	85.8±.40/**83.9**±.74/**76.7**±.57	95.0/**19.1**/**24.1**/67.0	**99.0**/ 7.6/16.1/**92.6**	98.9/ 3.5/ 8.4/92.0	98.5±.01/ 8.0±.73/19.3±.88/89.2±.47
	Macaroni2	59.4/60.7/68.1	76.0/**75.8**/70.2	68.3/54.3/59.7	**79.1**±.79/74.7±.85/**74.9**±.83	94.6/ 3.9/**12.5**/65.3	97.3/ **5.1**/12.2/87.0	93.2/ 0.6/ 3.9/77.8	**98.1**±.03/ 3.6±.02/10.4±.24/**87.2**±.03
	Capsules	**83.4**/**91.1**/**82.1**	70.3/83.2/77.8	74.1/82.8/74.6	79.2±.62/87.6±.17/79.8±.72	97.1/27.8/33.8/62.9	97.4/40.4/44.7/72.2	97.1/**52.9**/**53.3**/73.7	**98.2**±.03/30.4±.82/41.4±.81/**75.1**±.09
	Candle	69.3/73.9/68.1	**95.8**/**96.2**/**90.0**	84.1/73.3/76.6	90.4±.45/91.2±.53/83.7±.91	82.2/10.1/19.0/65.6	**99.0**/**23.6**/**32.6**/**93.0**	97.6/ 8.4/16.5/87.6	96.2±.09/16.8±.19/26.4±.21/85.2±.21
Single Instance	Cashew	81.7/89.7/87.3	**94.3**/**97.2**/**91.1**	88.0/91.3/84.7	87.8±.83/94.2±.38/86.1±.47	80.7/ 9.9/15.8/38.5	**99.0**/56.2/58.9/**88.5**	98.9/**68.9**/**66.0**/84.1	98.5±.06/63.9±.38/62.7±.24/78.8±.83
	Chewing Gum	93.7/97.2/91.0	**97.5**/**98.9**/**96.4**	96.4/98.2/93.8	94.9±.18/97.7±.06/91.4±.63	91.1/**62.4**/**63.3**/41.0	**99.1**/59.5/58.0/**85.0**	97.9/26.8/29.8/78.3	97.8±.04/61.6±.83/58.7±.88/71.5±.47
	Fryum	89.2/95.0/86.6	86.9/93.9/86.0	88.4/93.0/83.3	**94.3**±.28/**97.4**±.11/**90.9**±.26	92.4/38.8/38.6/69.5	97.3/46.6/**52.4**/82.0	93.0/39.1/45.4/85.1	**97.5**±.02/**47.1**±.08/50.3±.21/**87.8**±.76
	Pipe Fryum	82.8/91.2/84.0	95.3/97.6/92.9	90.8/95.5/88.6	**97.8**±.11/**99.0**±.05/**94.7**±.37	91.1/38.2/39.7/61.9	99.1/53.4/58.6/93.0	98.5/65.6/63.4/83.0	**99.5**±.02/**66.0**±.92/**66.5**±.52/**94.7**±.34
Average		79.5/82.8/79.4	88.8/90.8/85.8	87.2/87.0/81.7	**90.5**±.09/**91.7**±.22/**86.3**±.19	91.4/24.8/30.4/59.1	**98.3**/33.7/39.0/**85.5**	96.8/34.7/37.8/81.4	98.2±.00/**36.6**±.21/**41.1**±.15/85.1±.25
mAD		–	–	–	–	63.9	74.5	72.4	**75.6**

Table 6
Image-level multi-class anomaly classification (left) and pixel-level multi-class anomaly segmentation (right) results on Uni-Medical. Our approach yields significantly better results compared to the benchmark methods.

	Method →	Image-level mAU-ROC/mAP/mF_1-max				Pixel-level mAU-ROC/mAP/mF_1-max/mAU-PRO			
		DRAEM* (Zavrtanik et al., 2021a)	UniAD† (You et al., 2022a)	SimpleNet* (Liu et al., 2023b)	ViTAD	DRAEM* (Zavrtanik et al., 2021a)	UniAD† (You et al., 2022a)	SimpleNet* (Liu et al., 2023b)	ViTAD
	Category ↓	① ICCV'21	③ NeurIPS'22	② CVPR'23	③ (Ours)	① ICCV'21	③ NeurIPS'22	② CVPR'23	③ (Ours)
	Brain	69.2/90.1/90.7	89.9/**97.5**/92.6	82.3/95.6/90.9	**90.1**±.33/**97.5**±.28/**93.1**±.56	52.0/ 4.8/ 6.1/12.5	97.4/55.7/55.7/82.4	94.8/42.1/42.4/73.0	**97.8**±.03/**65.3**±.38/**61.8**±.26/**84.0**±.41
	Liver	59.1/52.7/60.7	61.0/48.8/63.2	55.8/47.6/60.9	**64.2**±.63/**55.4**±.72/**65.1**±.49	52.9/ 1.1/ 1.3/ 6.6	97.1/ 7.8/13.7/**92.7**	97.4/13.2/20.1/86.3	**98.0**±.05/**14.1**±.73/**22.0**±.37/90.5±.53
	Retinal	51.7/43.8/59.6	84.6/79.4/73.9	88.8/87.6/78.6	**92.1**±.28/**90.1**±.38/**82.0**±.27	57.4/ 6.6/11.3/ 0.9	94.8/49.3/51.3/79.9	95.5/59.5/56.3/82.1	**95.9**±.02/**70.3**±.32/**65.0**±.51/**83.9**±.40
	Average	60.0/62.2/70.3	78.5/75.2/76.6	75.6/76.9/**76.8**	**82.2**±.25/**81.0**±.37/**80.1**±.31	54.1/ 4.1/ 6.2/ 6.6	96.4/37.6/40.2/85.0	95.9/38.3/39.6/80.5	**97.2**±.01/**49.9**±.35/**49.6**±.32/**86.1**±.37
	mAD	–	–	–	–	37.7	69.9	69.1	**75.2**

officially published), our ViTAD still achieves highly competitive results that achieves higher results on several metrics other than mAD-PRO.

We have a few findings from these empirical results. First, models with higher single-class results do not necessarily perform better in multi-class scenarios, as shown in the comparison between UniAD (You et al., 2022a) and SimpleNet (Liu et al., 2023b). This could be due to model over-fitting or single-class-specific training strategies. Second, different methods yield similar results for categories under texture but show significant differences for categories with semantic objects, reflecting the diversity and effectiveness of different methods. Third, our method performs favorably across all categories, with no categories scoring particularly low (for instance, all mAU-ROC scores are above 90.0), demonstrating the effectiveness and generalizability of our method. In contrast, other methods invariably underperform in specific categories. Fourth, even without employing pyramidal encoders and decoders, our method still achieves state-of-the-art anomaly segmentation results, demonstrating the inherent fine-grained multi-scale modeling capability of the plain ViT.

Quantitative Evaluations on VisA. The VisA dataset contains more complex structures, multiple and large variations of objects, and more images. We use one of the most advanced and powerful methods from each category for comprehensive performance evaluations. The quantitative results in Table 5 show that ViTAD consistently performs well against state-of-the-art schemes. ViTAD surpasses UniAD by mAU-ROC/mAP/mF_1-max of +1.7 ↑/+0.9 ↑/+0.5 ↑. In addition, it obtains 75.6 on the comprehensive mAD metric, exceeding UniAD by +1.1 ↑.

Quantitative Evaluations on Uni-Medical. Compared to industrial AD datasets, Uni-Medical presents more significant challenges due to the more difficult anomaly types and a more comprehensive range of anomaly areas. Following the fair training and evaluation setting, Table 6 benchmarks quantitative results compared to the state-of-the-art methods. ViTAD achieves a significant advantage, reaching 75.2 mAD that surpasses the second-best UniAD by +5.3 ↑, while the performance of DRAEM significantly decreases. This indicates that our ViTAD has strong generalization ability across different types of datasets.

Qualitative Evaluations on Three AD Datasets. We conduct qualitative experiments to analyze the anomaly localization performance of evaluated methods. Fig. 6 shows the anomaly detection results on various object categories in MVTec AD (Bergmann et al., 2019a), VisA (Zou et al., 2022), and Uni-Medical (Bao et al., 2023) datasets. Compared to the augmentation-based DRAEM, reconstruction-based UniAD, and the embedding-based SimpleNet, our method can find more accurate and compact anomalous areas with less edge uncertainty and shows fewer false positives in normal areas. Using examples of a textural carpet with large-scale anomalies and a capsule object with irregularly shaped anomalies, ViTAD segments anonymous areas more accurately with fewer false positive responses.

Efficiency Comparison. We evaluate the model efficiency in five aspects: (1) number of parameters, (2) FLOPs, (3) train memory, (4) run-time (evaluated on a V100 GPU with a batch size of 8), and (5) train epoch. Table 7 shows that ViTAD requires significantly smaller memory for training and run-time while achieving SoTA performance. It is worth noting that ViTAD only requires 1.1 h and 2.3G of GPU memory for training while requiring very competitive parameters and FLOPs.

Summary. Results demonstrate that using only non-pyramidal ViT is sufficient to achieve state-of-the-art performance, and our ViTAD can be well generalized to various AD datasets in different domains. This indicates that a pyramidal structure for the encoder/decoder is unnecessary for constructing AD models.

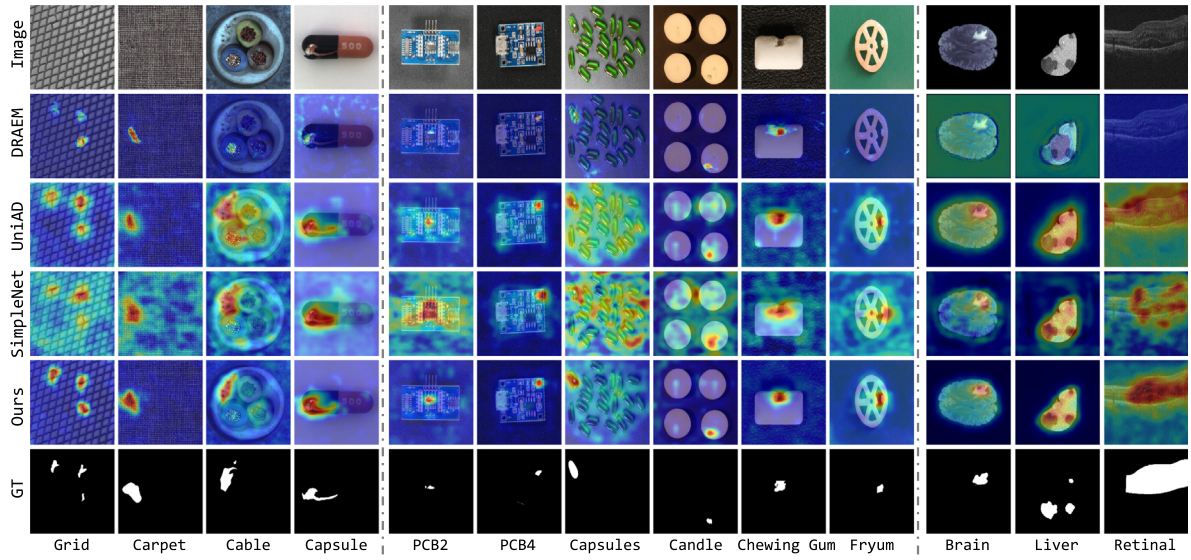


Fig. 6. Qualitative visualized results for anomaly segmentation. Compared with the latest augmentation-based DRAEM (Zavrtanik et al., 2021a) (2nd row), reconstruction-based UniAD (You et al., 2022a) (3rd row), and embedding-based SimpleNet (Liu et al., 2023b) (4th row) on **MVTec AD** (Bergmann et al., 2019a) (Left), **VisA** (Zou et al., 2022) (Middle), and **Uni-Medical** (Bao et al., 2023) (Right) datasets, our ViTAD (5th row) has a more accurate and compact anomaly location capability.

Table 7
Efficiency comparison of different methods on MVTec AD dataset in terms of six dimensions. *bold*, *underline*, and *wavy line* represent the best, second best, and third best results, respectively.

Method	Parameters	FLOPs	Train Memory	Train Time	Train Epoch	FPS
DRAEM	97.4 M	198.0 G	19,852 M	19.6 H	700	54.0
RD	80.6 M	28.4 G	3,872 M	4.1 H	200	90.6
UniAD	**24.5 M**	**3.6 G**	6,844 M	13.4 H	1,000	56.8
DeSTSeg	35.2 M	122.7 G	3,562 M	2.5 H	660	74.7
SimpleNet	72.8 M	16.1 G	5,488 M	11.8 H	200	49.3
RealNet	591.0 M	115.0 G	14,004 M	2.6 H	100	41.1
MambaAD	25.7 M	8.3 G	6,542 M	2.4 H	100	49.5
ViTAD (Ours)	38.6 M	10.7 G	**2,300 M**	**1.1 H**	**100**	**112.3**

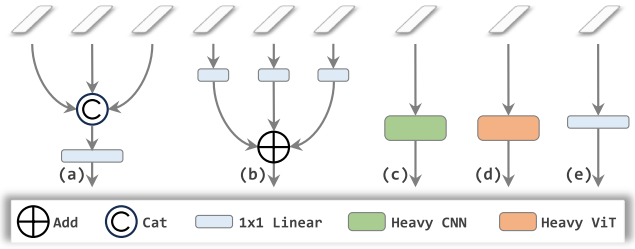


Fig. 7. Alternative Fuser $\mathcal{F}$. (a) Only the last feature F_4. (b) multi-depth features with concatenation (c) multi-depth features with addition; **(d)** F_4 followed by a heavy CNN network; **(e)** F_4 followed by a heavy ViT network.

4.3. Ablation studies

4.3.1. Global structural designs for ViTAD

Structural Variants of Fuser $\mathcal{F}$. In addition to the structure described in Section 3.3, we analyze different Fuser structures by exploiting two fusion schemes of multi-layer features in Fig. 7(a)–(b). Table 8 shows corresponding quantitative evaluation results. Concatenation and addition schemes exhibit negligible result differences. In addition, gradually integrating multi-depth features initially decreases and then slightly increases performance, indicating that multi-depth features are not the optimal choice for AD tasks. The phenomenon differs from the structural design of RD (Deng and Li, 2022) and UniAD (You et al., 2022a), where multi-scale features are deemed necessary for performance gains. Considering model complexity and performance, we only use the output feature of the last layer, *i.e.*, F_4, as the input to the decoder. This strikes a good balance between the accuracy and performance of the AD model. Furthermore, we explore the effect of deeper sub-models with CNN-based BottleNeck (Fig. 7(c)) and ViT block (Fig. 7(d)) as the input to the decoder. Additional CNN-based structure can significantly enhance the model performance in all metrics, termed ViTAD-C in orange background. These results indicate that complementary basic structures can bring additional gains for our ViTAD, even though the plain structure has already achieved impressive SoTA results. For instance, the average mAD increases from 85.4 to 86.2 (+0.8 ↑) after adding one BottleNeck layer, but more layers do not significantly improve the performance. In contrast, using ViT does not bring about a noticeable improvement in model performance. Following the principle of simple and effective structural design, ViTAD uses one linear layer as the Fuser structure (Fig. 7(e)) and keeps this design in Eq. (5).

Model Depth and Division Analysis. Considering the computational cost and performance of the model, a 12-layer ViT-S with three layers for each division is utilized for experiments. Only the last three divisions, *a.k.a.* stages in other contexts, are employed for the decoder. In addition to this division pattern, we experiment with different depth and division combinations, as shown in Table 9. (1) The top part demonstrates the effect of different depths of the encoder and decoder on the results, with shallower model depths leading to poorer performance. (2) The middle part shows the effect of an asymmetric decoder, which is found to have a negligible impact on the results, which also indirectly demonstrates the stability and robustness of our approach. (3) The bottom part indicates that different division methods have a minimal effect on the results, as these models fully consider both deep and shallow features. When using i divisions in the encoder, the output of the last division is fed into the Fuser, while the outputs of other $i-1$ divisions are used as restraint features. Since the features of the first stage are not used, the number of divisions in the decoder is always one less than that in the encoder. Considering both performance and structural generality, we choose the configuration from the last row as the final structure that is illustrated above (Section 3.3.2).

Effect of pretrained ViT. We analyze the effect of different pretrained ViT models on anomaly detection. Table 10 shows that the pretrained models are crucial for the subsequent anomaly detection. (1) The model with random weights still performs reasonably (first

Table 8
Quantitative evaluation of different Fuser $\mathcal{F}$ variants. The numbers in the "Cat" and "Add" rows at the top represent the usage of corresponding stage features, while the numbers in the "ViT" and "Conv" rows at the bottom denote the number of ViT (Dosovitskiy et al., 2021) and BottleNeck (He et al., 2016) layers used respectively. Methods in blue and orange represent our ViTAD and ViTAD-C, respectively.

	Fuser Variant	Image-level			Pixel-level				mAD
		mAU-ROC	mAP	mF_1-max	mAU-ROC	mAP	mF_1-max	mAU-PRO	
Cat	01 234	97.5	99.0	96.4	97.8	55.3	59.1	91.5	85.2
	1234	97.6	99.0	96.4	97.8	55.2	59.1	91.2	85.2
	234	97.4	98.8	96.3	97.2	54.3	57.7	91.0	84.7
	34	97.8	98.9	96.8	97.4	54.7	58.3	91.4	85.0
Add	01 234	97.5	99.0	96.5	97.8	55.4	59.3	91.3	85.3
	1234	97.5	99.0	96.5	97.8	55.2	59.1	91.2	85.2
	234	97.3	98.6	96.1	97.2	54.3	58.0	90.8	84.6
	34	98.2	99.3	97.1	97.3	54.4	58.2	91.1	85.1
ViT	1	98.2	99.3	97.2	97.6	55.4	58.5	91.2	85.4
	3	98.4	99.4	97.3	97.7	55.7	58.6	91.3	85.5
	5	98.3	99.3	97.5	97.7	55.4	58.7	91.3	85.5
Conv	1	98.4	99.4	97.6	98.0	57.6	60.2	91.9	86.2
	3	97.9	99.2	97.1	98.0	57.8	60.7	91.9	86.1
	5	98.1	99.3	97.1	98.1	58.5	60.7	91.8	86.2
	ViTAD	98.3	99.4	97.3	97.7	55.3	58.7	91.4	85.4

Table 9
Ablation study on model depth and division. $i \times j$ indicates that the network contains i divisions with each owing j layers for the decoder. $a-b-c$ means that the decoder has 3 divisions with a, b, and c layers, respectively.

Encoder	Decoder	Image-level	Pixel-level	mAD
1 × 3	1 × 3	69.9/83.6/86.4	81.1/21.9/27.1/58.4	61.2
2 × 3	1 × 3	93.6/96.9/93.6	93.8/48.6/52.3/87.1	80.8
3 × 3	2 × 3	98.1/99.2/97.2	97.2/53.9/57.6/91.2	84.9
4 × 3	3 × 2	98.0/99.2/97.2	97.6/55.0/58.5/91.3	85.3
4 × 3	3 × 4	98.1/99.2/97.2	97.7/55.4/58.7/91.2	85.3
4 × 3	2 −3− 4	98.1/99.1/97.1	97.7/55.3/58.7/91.5	85.4
4 × 3	4 −3− 2	98.1/99.2/97.2	97.6/55.1/58.4/91.0	85.2
6 × 2	5 × 2	98.1/99.2/97.1	97.6/55.5/58.7/91.5	85.4
3 × 4	2 × 4	97.8/99.1/97.3	97.9/55.7/59.0/91.4	85.4
4 × 3	3 × 3	98.3/99.4/97.3	97.7/55.3/58.7/91.4	85.4

row), as the image-level and pixel-level mAU-ROC values are higher than 0.5, and image-level mAP and mF1 reach 79.5 and 84.7, respectively Furthermore, we use ImageNet-1K pretrained weights to initialize the encoder and open it for training (second row). However, this is much worse than the frozen encoder (*c.f.*, the third row). This is because the finetuned weight on the small AD dataset will reduce the original feature distribution, resulting in a weak expression capability of encoder features. While AD approaches highly depend on rich expression as previous works (Deng and Li, 2022; You et al., 2022a; Liu et al., 2023b) acknowledged. (2) The AD performance depends heavily on the adopted pretrained ViTs as shown in Rows 3 to 10 of Section 4.1. For example, the mAD of DINO is +3.0 ↑ higher than the supervised training model of ImageNet-1K, and +1.8 ↑ higher than the self-supervised training MAE. The results based on CLIP are significantly worse than those of other methods, in which the training objective is to align images and texts that disregard information for detailed structures. DINOv2 (Oquab et al., 2024) does not perform as well as DINO on the AD task, demonstrating once again the significance of the pretrained weights. Moreover, the performance of the pretrained model in AD does not correlate with its corresponding classification accuracy. For instance, the classification accuracy of DINO-Small, MoCo v3-Small, and MAE-Small on ImageNet-1K are 82.8, 83.2, and 83.6 (He et al., 2022), respectively, but on the contrary, DINO-Small significantly better than other evaluated approaches Table 10. (3) Based on DINO pretrained weights, we further explore the effect of smaller patch size and higher image resolution on the results (*c.f.*, rows 11 and row 12). Both manners would increase computation costs. Nevertheless, they slightly reduce the image-level indicators but significantly increase the pixel-level results, *e.g.*, mAU-ROC, mAP, mF_1-max, and mAU-PRO increase by up to +0.6 ↑, +11.4 ↑, +7.0 ↑, and +2.0 ↑, and the averaged mAD increases from 85.4 to 88.1. Considering both performance and computation, we utilize DINO (Caron et al., 2021) as the pre-training model for the encoder. (4) Furthermore, we investigate the impact of pyramidal backbone architecture and pre-trained weights on comparative methods. As shown in Table 11, using the same reconstruction-based RD as an example, the model accuracy decreases when the pre-trained weights are switched to DINO. This indicates that the method is highly sensitive to pre-trained weights. In contrast, our method significantly benefits from DINO weights. Additionally, when the backbone is replaced with a pyramidal Swin Transformer (Liu et al., 2021), the performance also declines. This demonstrates that RD cannot leverage the ViT structure as effectively as our ViTAD, further proving the superiority of our architecture.

4.3.2. Local structural designs for ViTAD

Based on the improved global designs, we further investigate four subtle structural designs to enhance the performance, *i.e.*, (i) whether the output for the encoder goes through the final batch normalization. (ii) whether the feature fuser uses linear feature transformation. (iii) whether the class token is inherited. (iv) whether the position embeddings are used in the decoder. As shown in Table 12, each local design slightly improves the metric results, and the model achieves the best performance when using all components together, *e.g.*, mAD reaches 85.4 that obtains a 0.4 improvement compared to the baseline model. Our final ViTAD model takes the configuration of the last row.

4.3.3. Model scale analysis

We analyze the ViTAD scale on anomaly detection based on two pretrained settings: supervised learning with ImageNet-1K (Deng et al., 2009) and unsupervised learning with DINO features (Caron et al., 2021). As illustrated in Table 13, larger-scale models do not continually improve AD performance, contrary to recent findings in the classification, detection, and segmentation tasks. For instance, the image-level mAU-ROC achieved by the DINO-B model is 0.8 lower than that by the DINO-S model. In addition, the mAD achieved by the DINO-B model is 0.4 lower than that by the DINO-S model. Thus, we choose ViT-S with DINO-S weights as the detailed structure.

4.3.4. Resolution effect

As existing anomaly detection methods are designed to handle images of 224 × 224 (Deng and Li, 2022; Liu et al., 2023b) or 256 × 256 (Zavrtanik et al., 2021a; You et al., 2022a; Zhang et al., 2023b) pixels, there is no analysis of the effect of frame resolution on model performance. Considering the practical application requirements for different resolutions, we conduct experiments at intervals of 32 pixels within the range from 64 × 64 to 512 × 512 and additionally

Table 10
Empirical study on different pretrained weights of ViT-S. IN1K: pretrained on ImageNet-1K. IN22K: first pretrained on ImageNet-22K, then finetuned on ImageNet-1K. *: with patch size equaling 8. †: with further 384 resolution. ^{O}: Open encoder training. Note that ViT-B is used for MAE and CLIP due to the absence of the ViT-S model.

Model	Image-level			Pixel-level				mAD
	mAU-ROC	mAP	mF_1-max	mAU-ROC	mAP	mF_1-max	mAU-PRO	
Rand	59.5	79.5	84.7	74.7	15.3	20.4	45.7	54.2
IN1K^{O}	68.5	84.0	86.1	76.0	15.6	21.4	49.8	57.3
IN1K	94.4	97.5	95.3	96.6	51.6	54.5	87.2	82.4
IN22K	95.6	97.7	95.5	97.1	51.6	55.3	87.7	82.9
DeiT	95.8	98.1	96.1	97.1	53.9	56.8	87.8	83.7
CLIP	71.2	84.5	85.7	81.6	19.4	25.1	56.8	60.6
MoCo	95.3	97.7	95.2	97.4	53.0	56.2	90.6	83.6
MAE	95.3	97.7	95.2	97.4	53.0	56.2	90.6	83.6
DINOv2	93.2	96.8	94.9	96.1	47.4	52.4	86.8	81.1
DINO	98.3	99.4	97.3	97.7	55.3	58.7	91.4	85.4
DINO*	97.0	98.8	96.6	98.2	63.2	63.6	93.2	87.2
DINO†	97.3	98.9	96.3	98.3	66.7	65.7	93.4	88.1

Table 11
Research on pyramidal backbone architecture and pre-trained weights for reconstruction-based RD. The use of DINO weights in RD results in performance degradation, which demonstrates the inherent effectiveness of the ViTAD architecture itself, aside from the significant improvement brought by the introduction of DINO. .

Method	Pretrain	Image-level	Pixel-level	mAD
RD (Deng and Li, 2022)	IN1K	94.6/96.5/95.2	96.1/48.6/53.8/91.2	82.3
RD (Deng and Li, 2022)	DINO	89.0/94.7/92.7	94.8/47.4/51.3/89.2	79.9
RD-Swin Liu et al. (2021)	IN1K	90.2/94.9/92.9	95.1/47.6/52/88.6	80.2
ViTAD	DINO	98.3/99.4/97.3	97.7/55.3/58.7/91.4	85.4

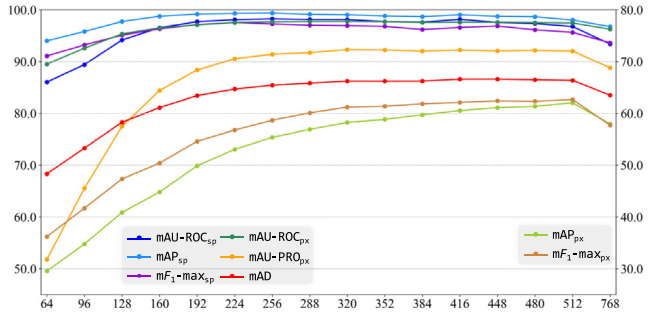


Fig. 8. Model performance of ViTAD with varying resolutions. Pixel-level mAP$_{px}$ and mF_1-max$_{px}$ use the right vertical axis, while the remaining metrics share the left vertical axis.

test the results under images of 768 × 768 pixels. As shown Fig. 8, the performance of ViTAD increases with larger resolution up to 256 × 256 pixels. We note that ViTAD still achieves satisfactory results under a low resolution (*e.g.*, 64 × 64), but it does not perform well when the resolution is high (*e.g.*, 768 × 768). As the typical resolution for AD is 256 × 256 pixels, we leave this for future work.

4.3.5. Restraint anomaly map

Based on the standard 4-stage division, Table 14 displays performance under different constrained anomaly maps during the training phase, with all corresponding constrained features computing the anomaly maps. The model performs worst when only the last A_3 is used, while it achieves the best results when the features of the last three anomaly maps are constrained. This indicates that the shallow anomaly map (A_0) would interfere with deep features ($A_i, i > 0$) that reduces the performance. Thus, our ViTAD employs A_1, A_2, and A_3 for the training constraint.

4.3.6. Robustness evaluation

We evaluate the robustness of ViTAD in several aspects.

Loss Function. Following Occam's Razor principle, the reconstruction-based ViTAD only uses a pixel-wise loss function for model training. Specifically, we use four types of loss functions, *i.e.*, L1, Mean Square Error (MSE), pixel-wise Cosine Similarity (Cos$_p$), and flattened Cosine Similarity (Cos$_f$), for quantitative comparison with the SoTA RD (Deng and Li, 2022) and UniAD (You et al., 2022a). As shown in Table 15, our method is robust to different loss functions, with mAD showing no significant fluctuations (bottom part). RD has noticeable gaps in multiple metrics (middle part), and UniAD has significant gaps (top part) where a significant decrease in model performance when switching to any Cos$_p$ and Cos$_f$.

Train Epoch. The convergence of the model significantly impacts its application value. The top part of Table 16 shows the performance under different epochs. The proposed method achieves stable results at 30 epochs and optimal results at 100 epochs. Considering the balance between training resources and performance, we set the default train epoch to 100, and this number still has a significant advantage compared to the comparison methods (see Section 4.1).

Train Scheduler. The Cosine scheduler has been shown to have a positive effect in the fields of classification, detection, and segmentation; it has not been exploited for anomaly detection.

Table 16 shows the robustness of our approach to different train schedulers, and the effect of the Cosine scheduler slightly decreases compared to the step scheduler.

Train Augmentation. Table 16 shows the effect of 5 types of data augmentation on model training. We find that proven effective data augmentation methods in other fields may have a negative effect in the AD field. This is because the aim of AD is to fit the domain of the training set as closely as possible, and these augmentations could broaden the domain range, resulting in poor test outcomes. As the scale of the AD dataset undergoes a substantial transformation, *i.e.*, a much larger AD dataset is proposed, these augmentations might potentially enhance the robustness of the model. We leave this interesting finding for future work.

Metric Stability During Training. When replicating comparison methods, we find that some methods have significant fluctuations in metric evaluation during training. Therefore, we analyze the fluctuations in metrics during training in several mainstream methods. As shown in Fig. 9, DRAEM (Zavrtanik et al., 2021a) and SimpleNet (Liu et al., 2023b) have noticeable jitters during training, while UniAD (You et al., 2022a) and our method are very stable, but our method has significant advantages in metric results and convergence speed.

4.3.7. Advantage explanation of ViT for AD

As shown in Fig. 6, ViTAD exhibits a more compact overlap segmentation with the ground truth than the competing methods. We attempt to explain the advantage of ViT for the AD task from a

Table 12
Empirical study on local designs. Before Norm: F_4 is obtained before normalization. Add Linear: Add the linear layer in Fuser. Remove CLS Token: Removing the class token throughout the procedure. Use Pos. Embed.: Keeping the position embedding in the decoder. ViTAD achieves the best performance when using all components together.

Before Norm	Add Linear	Remove CLS Token	Use Pos. Embed.	Image-level	Pixel-level	mAD
✘	✘	✘	✘	97.6/99.0/96.8	97.5/55.0/58.2/91.0	85.0
✔	✘	✘	✘	97.9/99.1/97.0	97.5/54.8/58.1/91.1	85.1
✘	✔	✘	✘	97.8/99.1/96.8	97.6/55.1/58.3/91.2	85.1
✘	✘	✔	✘	97.7/99.1/96.8	97.6/55.0/58.3/91.2	85.1
✘	✘	✘	✔	97.6/99.1/96.8	97.6/55.2/58.2/91.2	85.1
✘	✔	✔	✔	98.1/99.2/97.0	97.7/55.2/58.5/91.0	85.2
✔	✘	✔	✔	98.1/99.2/97.0	97.6/55.0/58.3/91.1	85.2
✔	✔	✘	✔	98.0/99.1/97.2	97.6/55.1/58.3/91.3	85.2
✔	✔	✔	✘	98.0/99.1/97.2	97.7/55.3/58.5/91.3	85.3
✔	✔	✔	✔	98.3/99.4/97.3	97.7/55.3/58.7/91.4	85.4

Table 13
Ablation study on model scaling. Besides DINO-Small/Base models, results of ViT trained on ImageNet-1K at different scales are also presented. Pretrained models are from tripartite TIMM of version v0.8.15dev0.

	Backbone Scale	Image-level			Pixel-level				mAD
		mAU-ROC	mAP	mF_1-max	mAU-ROC	mAP	mF_1-max	mAU-PRO	
ImageNet-1K	T	89.7	95.2	93.0	94.3	43.8	49.7	82.7	78.3
	S	94.4	97.5	95.3	96.6	51.6	54.5	87.2	82.4
	B	96.1	98.1	96.5	96.0	50.7	54.6	86.8	82.7
	L	94.9	97.6	94.3	94.6	45.2	49.6	85.7	80.3
	H	90.9	95.6	92.2	93.8	45.0	48.3	79.8	77.9
DINO	S	98.3	99.4	97.3	97.7	55.3	58.7	91.4	85.4
	B	97.5	99.0	96.7	97.9	55.4	58.6	90.1	85.0

Table 14
Empirical study on restraint stages. Numbers represent the constrained stages.

Restraint Stage	Image-level			Pixel-level				mAD
	mAU-ROC	mAP	mF_1-max	mAU-ROC	mAP	mF_1-max	mAU-PRO	
A_3	95.2	98.0	94.6	96.9	50.0	54.7	87.1	82.4
A_2, A_3	97.0	98.8	96.3	97.4	52.3	57.1	89.2	84.0
A_1, A_2, A_3	98.3	99.4	97.3	97.7	55.3	58.7	91.4	85.4
A_0, A_1, A_2, A_3	95.5	97.6	95.2	97.4	53.5	57.4	90.5	83.9

Table 15
Ablation study on different pixel-wise loss functions of different approaches. †: Default loss function in the paper. Our ViTAD is robust to different loss functions, with mAD showing no significant fluctuations.

	Item	Image-level			Pixel-level				mAD
		mAU-ROC	mAP	mF_1-max	mAU-ROC	mAP	mF_1-max	mAU-PRO	
RD	L1	93.4	97.2	95.7	95.8	46.8	52.2	90.2	81.6
	MSE	97.7	99.0	96.7	96.4	48.4	53.3	91.1	83.2
	Cos_p	95.3	97.6	96.1	96.2	50.5	55.0	91.5	83.2
	Cos_f †	94.6	96.5	95.2	96.1	48.6	53.8	91.2	82.3
UniAD	L1	96.6	98.7	96.5	96.8	44.4	49.7	90.2	81.8
	MSE †	97.5	99.1	97.3	97.0	45.1	50.4	90.7	82.4
	Cos_p	74.6	88.1	87.9	84.1	19.1	24.6	64.2	63.2
	Cos_f	76.5	89.1	88.5	82.1	18.4	24.2	62.1	63.0
ViTAD	L1	97.6	99.0	96.8	97.5	54.9	58.4	91.2	85.1
	MSE	97.7	99.1	97.0	97.6	55.0	58.6	91.5	85.2
	Cos_p	98.1	99.3	97.2	97.7	55.4	58.9	91.8	85.5
	Cos_f †	98.3	99.4	97.3	97.7	55.3	58.7	91.4	85.4

frequency perspective. Given the substantial distribution difference between abnormal and normal regions, high-frequency information can serve as a representation of the abnormal area. Thus, we show the high- and low-frequency components of the original image (second and third columns) in Fig. 10, with red rectangular boxes indicating the high-frequency abnormal representation regions. ViTAD outperforms comparison methods in segmenting more accurate and compact abnormal regions. This is attributed to the global multi-head self-attention module of ViT, which can capture long-range context information and high-frequency details as previously demonstrated (Zhang et al., 2022a; Si et al., 2022), while methods that employ a pyramidal structure with low-receptive fields lack this ability.

5. Conclusion

This paper addresses the multi-class unsupervised anomaly detection task using a plain vision transformer. Specifically, we abstract a Meta-AD framework based on the current reconstruction methods, and by Occam's Razor principle, we propose a powerful yet efficient ViTAD baseline. We propose a comprehensive and fair evaluation benchmark on eight metrics for this increasingly popular task. ViTAD achieves impressive results on MVTec AD, VisA, and Uni-Medical datasets without inventing or introducing additional modules, datasets, or training techniques. In addition, we conduct thorough experiments to demonstrate the effectiveness and robustness of our method.

Table 16
Ablation study on training epoch, scheduler (Sche.), and augmentation. CC: Center Crop. CJ: Color Jitter. RHF: Random Horizontal Flip. RR: Random Rotation. RRC: Random Resized Crop.

	Item	Image-level			Pixel-level				mAD
		mAU-ROC	mAP	mF_1-max	mAU-ROC	mAP	mF_1-max	mAU-PRO	
Epoch	30	97.4	99.0	96.6	97.5	55.2	58.7	91.1	85.1
	50	97.9	99.1	97.1	97.7	55.5	58.8	91.3	85.3
	100	98.3	99.4	97.3	97.7	55.3	58.7	91.4	85.4
	200	98.1	99.1	96.7	97.6	55.6	58.4	91.3	85.3
	300	98.1	99.1	96.9	97.6	55.4	58.6	91.2	85.3
Sche.	Cosine	98.0	99.0	97.1	97.7	55.5	58.7	91.2	85.3
	Step	98.3	99.4	97.3	97.7	55.3	58.7	91.4	85.4
Augmentation	CC+CJ	73.1	89.8	86.3	66.8	8.5	15.0	20.9	51.5
	CC+RHF	75.6	90.9	86.7	65.8	8.6	14.8	19.1	51.6
	CC+RR	72.1	88.9	85.7	67.0	8.6	15.1	21.0	51.2
	RRC	92.5	96.5	92.1	84.1	18.6	27.9	48.6	65.8
	All	88.3	95.0	90.3	75.9	13.2	21.7	35.9	60.0
	None	98.3	99.4	97.3	97.7	55.3	58.7	91.4	85.4

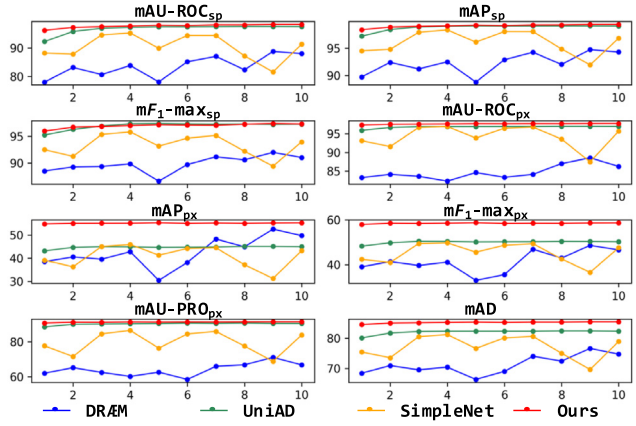


Fig. 9. Stability comparison of all metrics during the training process for different methods. Each model is tested ten times at linear intervals during the training process.

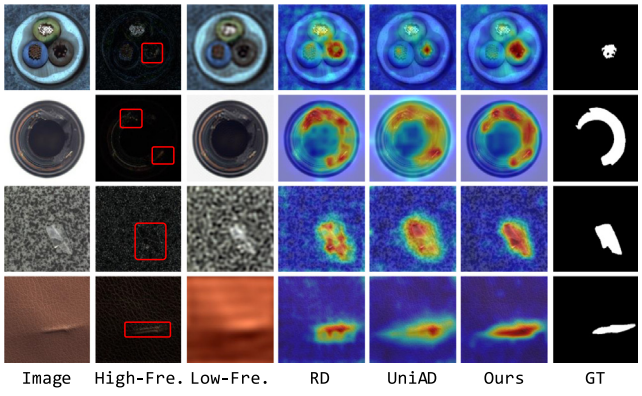


Fig. 10. Comparative analysis from a frequency domain perspective. The second and third columns represent the high- and low-frequency decomposition of the input image in the first column, while the last column represents the ground truth of the anomaly segmentation. The other columns display the anomaly segmentation results of different methods.

Broad Impact. This work thoroughly benchmarks mainstream and latest AD methods on three datasets under the challenging MUAD setting. Simultaneously, a novel ViTAD is proposed to explore the potential of plain ViT in AD tasks, filling the research gap and stimulating subsequent research works. In addition, the proposed ViTAD achieves state-of-the-art results on multiple datasets, with significantly smaller training costs, faster inference speed, and more memory-friendly, demonstrating its greater application values.

CRediT authorship contribution statement

Jiangning Zhang: Writing – review & editing, Writing – original draft, Software, Methodology, Data curation, Conceptualization. **Xuhai Chen:** Writing – review & editing, Writing – original draft, Visualization, Methodology. **Yabiao Wang:** Writing – review & editing, Investigation, Funding acquisition. **Chengjie Wang:** Writing – review & editing, Investigation, Funding acquisition. **Yong Liu:** Writing – review & editing, Supervision, Resources, Project administration, Methodology. **Xiangtai Li:** Writing – review & editing, Validation, Methodology. **Ming-Hsuan Yang:** Writing – review & editing, Resources, Methodology. **Dacheng Tao:** Writing – review & editing, Resources, Methodology.

Declaration of competing 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.

Acknowledgment

This work is supported by a Grant from The National Natural Science Foundation of China (No. 62103363).

Data availability

I have shared the link to my full code in the manuscript.

References

Akcay, S., Atapour-Abarghouei, A., Breckon, T.P., 2019. Ganomaly: Semi-supervised anomaly detection via adversarial training. In: ACCV.
Akçay, S., Atapour-Abarghouei, A., Breckon, T.P., 2019. Skip-ganomaly: Skip connected and adversarially trained encoder-decoder anomaly detection. In: IJCNN.
Bao, J., Sun, H., Deng, H., He, Y., Zhang, Z., Li, X., 2023. BMAD: Benchmarks for medical anomaly detection. arXiv preprint arXiv:2306.11876.
Batzner, K., Heckler, L., König, R., 2024. Efficientad: Accurate visual anomaly detection at millisecond-level latencies. In: CACV.
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. IJCV.
Bergmann, P., Fauser, M., Sattlegger, D., Steger, C., 2019a. MVTec AD–A comprehensive real-world dataset for unsupervised anomaly detection. In: CVPR.
Bergmann, P., Fauser, M., Sattlegger, D., Steger, C., 2020. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In: CVPR.
Bergmann, P., Löwe, S., Fauser, M., Sattlegger, D., Steger, C., 2019b. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. In: VISIGRAPP.
Berroukham, A., Housni, K., Lahraichi, M., Boulfrifi, I., 2023. Deep learning-based methods for anomaly detection in video surveillance: a review. BEEI.
Cao, Y., Wan, Q., Shen, W., Gao, L., 2022. Informative knowledge distillation for image anomaly segmentation. KBS.

Cao, Y., Xu, X., Sun, C., Cheng, Y., Du, Z., Gao, L., Shen, W., 2023. Segment any anomaly without training via hybrid prompt regularization. arXiv preprint arXiv:2305.10724.

Caron, M., Touvron, H., Misra, I., Jegou, H., Mairal, J., Bojanowski, P., Joulin, A., 2021. Emerging properties in self-supervised vision transformers. In: ICCV.

Chen, X., Han, Y., Zhang, J., 2023a. A zero-/few-shot anomaly classification and segmentation method for CVPR 2023 VAND workshop challenge tracks 1&2: 1st place on zero-shot AD and 4th place on few-shot AD. arXiv preprint arXiv:2305.17382.

Chen, L., You, Z., Zhang, N., Xi, J., Le, X., 2022. UTRAD: Anomaly detection and localization with U-transformer. Neural Netw..

Chen, X., Zhang, J., Tian, G., He, H., Zhang, W., Wang, Y., Wang, C., Wu, Y., Liu, Y., 2023b. CLIP-AD: A language-guided staged dual-path model for zero-shot anomaly detection. arXiv preprint arXiv:2311.00453.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A., 2014. Describing textures in the wild. In: CVPR.

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database. In: CVPR.

Deng, H., Li, X., 2022. Anomaly detection via reverse distillation from one-class embedding. In: CVPR.

Deng, S., Sun, Z., Zhuang, R., Gong, J., 2023. Noise-to-norm reconstruction for industrial anomaly detection and localization. arXiv preprint arXiv:2307.02836.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR.

Gu, A., Dao, T., 2023. Mamba: Linear-time sequence modeling with selective state spaces. arXiv preprint arXiv:2312.00752.

Gu, Z., Liu, L., Chen, X., Yi, R., Zhang, J., Wang, Y., Wang, C., Shu, A., Jiang, G., Ma, L., 2023. Remembering normality: Memory-guided knowledge distillation for unsupervised anomaly detection. In: ICCV.

Gu, Z., Zhu, B., Zhu, G., Chen, Y., Tang, M., Wang, J., 2024. Anomalygpt: Detecting industrial anomalies using large vision-language models. In: AAAI.

He, H., Bai, Y., Zhang, J., He, Q., Chen, H., Gan, Z., Wang, C., Li, X., Tian, G., Xie, L., 2024a. Mambaad: Exploring state space models for multi-class unsupervised anomaly detection. In: NeurIPS.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R., 2022. Masked autoencoders are scalable vision learners. In: CVPR.

He, K., Fan, H., Wu, Y., Xie, S., Girshick, R., 2020. Momentum contrast for unsupervised visual representation learning. In: CVPR.

He, H., Zhang, J., Chen, H., Chen, X., Li, Z., Chen, X., Wang, Y., Wang, C., Xie, L., 2024b. DiAD: A diffusion-based framework for multi-class anomaly detection. In: AAAI.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: CVPR.

Hu, T., Zhang, J., Yi, R., Du, Y., Chen, X., Liu, L., Wang, Y., Wang, C., 2024. AnomalyDiffusion: Few-shot anomaly image generation with diffusion model. In: AAAI.

Jeong, J., Zou, Y., Kim, T., Zhang, D., Ravichandran, A., Dabeer, O., 2023. Winclip: Zero-/few-shot anomaly classification and segmentation. In: CVPR.

Jiang, X., Liu, J., Wang, J., Nie, Q., Wu, K., Liu, Y., Wang, C., Zheng, F., 2022. Softpatch: Unsupervised anomaly detection with noisy data. In: NeurIPS.

Kim, W., Son, B., Kim, I., 2021. Vilt: Vision-and-language transformer without convolution or region supervision. In: ICML.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollar, P., Girshick, R., 2023. Segment anything. In: ICCV.

Lei, J., Hu, X., Wang, Y., Liu, D., 2023. PyramidFlow: High-resolution defect contrastive localization using pyramid normalizing flow. In: CVPR.

Li, Y., Mao, H., Girshick, R., He, K., 2022. Exploring plain vision transformer backbones for object detection. In: ECCV.

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

Liang, Y., Zhang, J., Zhao, S., Wu, R., Liu, Y., Pan, S., 2023. Omni-frequency channel-selection representations for unsupervised anomaly detection. TIP.

Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L., 2014. Microsoft coco: Common objects in context. In: ECCV.

Liu, W., Chang, H., Ma, B., Shan, S., Chen, X., 2023a. Diversity-measurable anomaly detection. In: CVPR.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In: ICCV.

Liu, J., Xie, G., Wang, J., Li, S., Wang, C., Zheng, F., Jin, Y., 2024. Deep industrial image anomaly detection: A survey. In: MIR.

Liu, Z., Zhou, Y., Xu, Y., Wang, Z., 2023b. Simplenet: A simple network for image anomaly detection and localization. In: CVPR.

Loshchilov, I., Hutter, F., 2019. Decoupled weight decay regularization. In: ICLR.

Lu, R., Wu, Y., Tian, L., Wang, D., Chen, B., Liu, X., Hu, R., 2023. Hierarchical vector quantized transformer for multi-class unsupervised anomaly detection. In: NeurIPS.

Madan, N., Ristea, N.C., Ionescu, R.T., Nasrollahi, K., Khan, F.S., Moeslund, T.B., Shah, M., 2023. Self-supervised masked convolutional transformer block for anomaly detection. TPAMI.

Mathian, E., Liu, H., Fernandez-Cuesta, L., Samaras, D., Foll, M., Chen, L., 2022. Haloae: An halonet based local transformer auto-encoder for anomaly detection and localization. arXiv preprint arXiv:2208.03486.

Mei, S., Yang, H., Yin, Z., 2018. An unsupervised-learning-based approach for automated defect inspection on textured surfaces. TIM.

Mishra, P., Verk, R., Fornasier, D., Piciarelli, C., Foresti, G.L., 2021. VT-ADL: A vision transformer network for image anomaly detection and localization. In: ISIE.

Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P.Y., Li, S.W., Misra, I., Rabbat, M., Sharma, V., Synnaeve, G., Xu, H., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P., 2024. DINOv2: Learning robust visual features without supervision. TMLR.

Pirnay, J., Chai, K., 2022. Inpainting transformer for anomaly detection. In: ICIAP.

Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al., 2021. Learning transferable visual models from natural language supervision. In: ICML.

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

Schlegl, T., Seeböck, P., Waldstein, S.M., Schmidt-Erfurth, U., Langs, G., 2017. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: IPMI.

Si, C., Yu, W., Zhou, P., Zhou, Y., Wang, X., Yan, S., 2022. Inception transformer. In: NeurIPS.

Tan, M., Le, Q., 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. In: ICML.

Tien, T.D., Nguyen, A.T., Tran, N.H., Huy, T.D., Duong, S., Nguyen, C.D.T., Truong, S.Q., 2023. Revisiting reverse distillation for anomaly detection. In: CVPR.

Wan, Q., Cao, Y., Gao, L., Shen, W., Li, X., 2022. Position encoding enhanced feature mapping for image anomaly detection. In: CASE.

Wang, X., Zhang, X., Cao, Y., Wang, W., Shen, C., Huang, T., 2023. Seggpt: Segmenting everything in context. arXiv preprint arXiv:2304.03284.

Xie, G., Wang, J., Liu, J., Jin, Y., Zheng, F., 2023. Pushing the limits of fewshot anomaly detection in industry vision: Graphcore. In: ICLR.

Xu, Y., Zhang, J., Zhang, Q., Tao, D., 2022. Vitpose: Simple vision transformer baselines for human pose estimation. In: NeurIPS.

Yao, H., Wang, X., 2022. Generalizable industrial visual anomaly detection with self-induction vision transformer. arXiv preprint arXiv:2211.12311.

You, Z., Cui, L., Shen, Y., Yang, K., Lu, X., Zheng, Y., Le, X., 2022a. A unified model for multi-class anomaly detection. In: NeurIPS.

You, Z., Yang, K., Luo, W., Cui, L., Zheng, Y., Le, X., 2022b. Adtr: Anomaly detection transformer with feature reconstruction. In: ICONIP.

Zavrtanik, V., Kristan, M., Skočaj, D., 2021a. Draem-a discriminatively trained reconstruction embedding for surface anomaly detection. In: ICCV.

Zavrtanik, V., Kristan, M., Skočaj, D., 2021b. Reconstruction by inpainting for visual anomaly detection. PR.

Zavrtanik, V., Kristan, M., Skočaj, D., 2022. Dsr–a dual subspace re-projection network for surface anomaly detection. In: ECCV.

Zhang, J., Chen, X., Xue, Z., Wang, Y., Wang, C., Liu, Y., 2023a. Exploring grounding potential of VQA-oriented GPT-4V for zero-shot anomaly detection. arXiv preprint arXiv:2311.02612.

Zhang, J., He, H., Gan, Z., He, Q., Cai, Y., Xue, Z., Wang, Y., Wang, C., Xie, L., Liu, Y., 2024a. ADer: A comprehensive benchmark for multi-class visual anomaly detection. arXiv preprint arXiv:2406.03262.

Zhang, X., Li, S., Li, X., Huang, P., Shan, J., Chen, T., 2023b. DeSTSeg: Segmentation guided denoising student-teacher for anomaly detection. In: CVPR.

Zhang, J., Li, X., Li, J., Liu, L., Xue, Z., Zhang, B., Jiang, Z., Huang, T., Wang, Y., Wang, C., 2023c. Rethinking mobile block for efficient attention-based models. In: ICCV.

Zhang, J., Li, X., Wang, Y., Wang, C., Yang, Y., Liu, Y., Tao, D., 2022a. Eatformer: Improving vision transformer inspired by evolutionary algorithm. arXiv preprint arXiv:2206.09325.

Zhang, B., Tian, Z., Tang, Q., Chu, X., Wei, X., Shen, C., et al., 2022b. Segvit: Semantic segmentation with plain vision transformers. In: NeurIPS.

Zhang, J., Wang, C., Li, X., Tian, G., Xue, Z., Liu, Y., Pang, G., Tao, D., 2024b. Learning feature inversion for multi-class anomaly detection under general-purpose COCO-AD benchmark. arXiv preprint arXiv:2404.10760.

Zhang, X., Xu, M., Zhou, X., 2024c. RealNet: A feature selection network with realistic synthetic anomaly for anomaly detection. In: CVPR.

Zhao, Y., 2023. OmniAL: A unified CNN framework for unsupervised anomaly localization. In: CVPR.

Zhou, C., Paffenroth, R.C., 2017. Anomaly detection with robust deep autoencoders. In: KDD.

Zou, Y., Jeong, J., Pemula, L., Zhang, D., Dabeer, O., 2022. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In: ECCV.