No-MambAAD: Revitalizing Conv-Only Networks for Unsupervised Anomaly Detection

No-MambAAD: Revitalizing Conv-Only Networks for Unsupervised Anomaly Detection

Masud An-Nur Islam Fahim
University of Vaasa
Vaasa, Finland
masud.fahim@uwasa.fi

Jani Boutellier
University of Vaasa
Vaasa, Finland
jani.boutellier@uwasa.fi

Abstract

Most of the current state-of-the-art visual unsupervised anomaly detection (UAD) methods leverage complex neural architecture modules: Transformer-based methods provide high-quality anomaly detection performance due to their global feature extraction capability, similar to the recent Mamba based methods that combine the strengths of CNNs and Transformers. Some of the simpler reconstruction-based UAD methods are purely CNN-based, which offers linear complexity, but is performance-restricted by feature extraction locality. Hence, the architecture variants have inherent design trade-offs: CNNs lacks long-range feature interaction, Transformers struggle with quadratic complexity, and Mamba based solutions suffer in high parameter count and scalability. In this work we propose to revisit CNN-based approaches by introducing novel strip-modulation and gated-mixer mechanisms, and propose ***No-MambAAD****, a novel visual UAD method absent of* ***Mamba*** *and* ***Attention*** *blocks. The proposed method offers similar or better anomaly detection performance than the current state-of-the-art approaches and outperforms the current state-of-the-art across multiple benchmarks with* ***38%*** *smaller parameter count.*

1. Introduction

Unsupervised Anomaly Detection (UAD) comprises of techniques used to identify unusual or unexpected patterns in test images, which differ from the typical characteristics of normal training images — all achieved without relying on labeled training data. Besides evident and popular applications in industrial visual quality inspection, UAD has been used in the design of many supporting systems for medical and military purposes [5], where paired data acquisition is expensive.

UAD approaches can be organized into three categories, as outlined in prior works [6]: Embedding-based [2–4, 12],

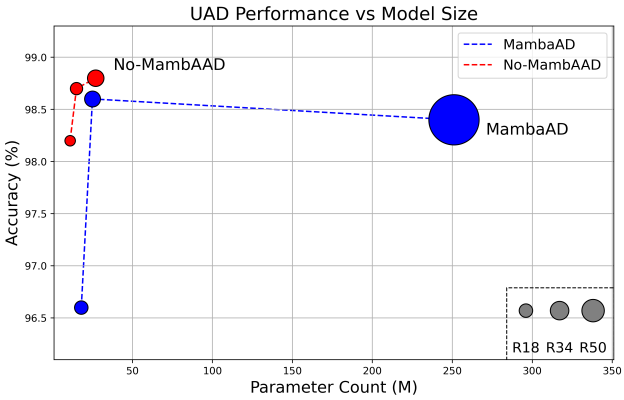

Figure 1. Parameter count vs. UAD performance between the proposed No-MambAAD and the current SotA study MambaAD [6] with identical encoders as and the MVTecAD [1] dataset for a fair comparison. No-MambAAD shows steady improvement in accuracy with larger encoders, whereas MambaAD [6] saturates. Additionally, No-MambAAD provides significantly lower decoder parameter count than MambaAD [6].

Synthesis-based [8, 9, 20, 22], and Reconstruction-based [5, 6, 15, 18]works. Reconstruction-based approaches typically maximize the feature similarity within multiple scales. For instance, the conv-only method RD4AD [15] or the Mamba variant MambaAD [6] use a pre-trained teacher-student model to detect anomalies by comparing features across multiple scales, or UniAD [6] that employs a pre-trained encoder paired with a transformer decoder, matching features within a single scale.

In this study, we aim to revitalize the convolution based approach for UAD without resorting to recent popular architectural modules such as **Mamba** or **A**ttention variants, and propose, **No-MambAAD**. The latter offer global feature interaction but at the cost of quadratic complexity. One the other hand, Mamba merges global and local feature interaction with linear complexity, but requires complicated deployment and customized scanning for implementing global

feature interaction. In contrast, regular convolution based solutions provide only local feature interaction, however with linear complexity.

Hybrid Mamba-convolution or attention-convolution architectures provide local-global feature extraction, but bring along the inherent challenges of Mamba/Attention blocks: issues with scalability and parameter count. An example of such a hybrid method is MambaAD, a Conv-Mamba hybrid that shows state-of-the-art (SotA) performance in UAD tasks by relying on the Mamba scanning scheme.

In contrast, MambaOut [19] is a recent pure convolutional network, which shows that Mamba blocks can be avoided in visual recognition and similar tasks by introducing gated convolutional layers, still achieving comparable or better performance than Mamba variants.

These two studies have inspired the proposed reconstruction-based UAD work that revolves around a novel and efficient global feature extraction mechanism; the conv-only module *No-MambA* acts as a mixed feature aggregator within the decoder at each scale and as a feature refiner before feeding into the loss layer. To ensure global feature extraction, No-MambA utilizes a novel strip modulation operation that includes convolution, normalization, addition, and multiplication, maintaining linear complexity scaling. In principle, our deployed strip modulation weighs the local **strips** from the given feature tensors through a learnable set of weights, which stem from the global information of the feature tensor. Indeed, like the self-attention mechanism, this process does not consider the complete query-key-value setup. However, the proposed strip modulation can be applied for horizontal and vertical directions and weigh the full feature map through carefully designed weighted aggregation. We place this modulation operation in between the iterative convolution operations within the decoder, resulting in efficient global feature propagation throughout the reconstruction phase. This conv-only design choice shows surprising improvement over alternative global receptive field propagators like Attention or Mamba by achieving SotA performance across multiple datasets. In summary, we propose No-MambAAD, a reconstruction-based conv-only approach for UAD tasks with the following key contributions:

- We propose No-MambAAD, a convolution-only unsupervised anomaly detection method that achieves SotA performance in UAD for multiple datasets.
- No-MambAAD uses strip modulation, an effective global feature extractor, and integrates it with standard local feature extractors to overcome the typical limitations of convolutional approaches.
- Considering recent SotA Mamba-based [6] and Transformer-based [21] studies in UAD, No-MambAAD requires 38% and 59% fewer parameters, respectively, to outperform them in anomaly detection

2. Related Work

As mentioned before, AD works can be divided to three major categories: Embedding-based [2–4, 12], Synthesizing-based [8, 9, 20, 22], and Reconstruction-based [5, 6, 10, 15, 18]. Relying on a pre-trained backbone model, embedding-based approaches project RGB images into a multi-channel feature space. For example, [14] takes the latent multi-channel features and maps them into a multivariate Gaussian distribution for the UAD task, or [13] utilizes a memory bank to extract nominal patch features, which helps to compute the Mahalanobis distance for evaluation. Synthesizing UAD works synthesize the pseudo-anomaly images by applying various kinds of perturbations upon related images. For instance, [20] uses Perlin noise and textured images for anomaly synthesis, or [16] adopts Poisson image blending for anomalous sample production.

Finally, reconstruction-based UAD approaches rely upon matching feature similarities within different scales, and the proposed study falls into this category. Prior to our work, RD4AD [15] deployed a conv-only encoder-decoder setup for the UAD task in a multi-scale fashion, where the encoder remains frozen all over the training. Following [15], MambaAD [6] state space modeling within the RD4AD [15] framework achieved SotA performance. However, MambaAD [6] requires customized deployment like all other Mamba based works, and high classification performance necessitates extensive model size. Similarly, VitAD [21] uses a plain vision transformer, DinoMaly [5] adopts DINOv2-Register ViT [11] with unfocused linear attention, UniAD uses a masked attention module, and DiAD explores the diffusion model for consistent reconstruction guidance during the UAD task.

Table 1 presents a brief comparison between the current SotA UAD studies.

3. No-MambAAD

This study introduces the proposed No-MambAAD approach for UAD. Our framework includes a pre-trained image encoder, a bottleneck, and a decoder that utilizes the superior feature extraction capabilities of our novel No-MambA module. As No-MambA falls within the reconstruction-based anomaly detection category, feature matching is adopted between the encoder and decoder at multiple scales. During training, normal images are fed into the encoder, and from here, features go into the bottleneck, and the output of the bottleneck enters the decoder. Both the bottleneck and decoder are trainable entities, and we update them by maximizing the cosine similarity between feature tensors from the encoder and the decoder. In Figure 2, we present a flowchart of No-MambAAD.

Method	Pyramidal Encoder	Heavy Fuser	Pyramidal Decoder	Multi-Resolution Features	Image/Feature Augmentation	Category		
						Syn.	Emb.	Rec.
DRAEM [20]	✔	✖	○	✔	✔	✔	✚	✚
RD4AD [15]	✔	✔	✔	✔	✖	✖	✖	✔
UniAD [18]	✔	✖	✖	✔	✔	✔	✖	✔
DeSTSeg [22]	✔	✖	○	✔	✔	✔	✔	✖
SimpleNet [9]	✔	✖	○	✔	✔	✔	✔	✖
ViTAD [21]	✖	✖	✖	✖	✖	✖	✖	✔
MambaAD	✔	✔	✔	✔	✖	✖	✖	✔
No-MambAAD	✔	✔	✔	✔	✖	✖	✖	✔

Table 1. A methodical comparison between SotA UAD studies. ✔: Satisfied; ✖: Unsatisfied; ✚: Partially satisfied; ○: Inapplicable.

Strip Modulation (SM). Self-attention weighs the value representation of the input via attention weights from the query and key, but introduces quadratic computation cost. In contrast, the compute demand of convolution is linear but lacks the global feature interaction of self-attention. In the proposed No-MambAAD method, Strip modulation introduces global feature interaction without incurring the quadratic computational cost. We are not claiming that Strip modulation is equivalent to self-attention, however, it follows global weighing similar to self-attention to some extent, returning significant performance improvement over transformer variants in UAD tasks.

Strip modulation starts with an adaptive-average pooling operation to extract condensed global information from the input feature, followed by convolution, normalization, and activation operations. These operations are applied to the pooling layer's output to formulate it as a set of adaptive modulation weights to be applied to the raw input feature of the strip modulation block. The modulation weights update the channels through pointwise modulation, leaving a single weight for each channel. In strip modulation, the weights and feature channels are first grouped based on a predefined parameterization; then, based on the group count, strip, and kernel size, we reshape the adaptive weights from pooling layers and the grouped features.

At this stage, strips from reshaped features undergo a modulation that is based on grouped weights, and are summed accordingly. This process continues in a sliding window fashion. This particular modulation provides two benefits that are absent in regular pointwise modulation. First, the adaptive kernels, derived from global average pooling and convolution, encapsulate condensed global information from the input, enabling each group's strips to be modulated with weights tailored to the overall spatial and channel context, resulting in a weighted representation of the strip patches across grouped channels, enhancing adaptability to input-specific features. Second, due to the sliding window nature of strip modulation, just like the convolution, each strip's information passes on to the next strip and efficiently captures spatial relationships within each strip without requiring global pairwise interactions, reducing computational complexity compared to self-attention.

Given the scanning direction, our strip modulation layer can access the strip horizontally or vertically from the tensors. After modulation, we weigh it with trainable coefficients and deduct global-average pooling weighted input features, returning the modulated low-frequency component. In parallel, we also weigh the input tensor with another set of trainable weights to obtain the high-frequency components. Finally, aggregating both results returns the desired representation of the input features.

Internal Feature Rectifier (InFR). The No-MambAAD InFR module includes both regular and gated convolution, followed by strip modulation, and ends with a regular addition operation. At first, incoming features go through standard convolution, where the projected output shares the same number of channels as the input. Then, both the input and the convolved features go through a gated convolution. The output from this stage then continues to a separate strip modulation operation; strips are taken from different directions. Since straightforward strip modulation does not provide equally comprehensive global feature interaction as self-attention, the patches or strips are modulated individually and combined to achieve improved global feature interaction. To propagate the local features from the convolutional layers along the modulated features, we add the original input of the InFR module to the final aggregation operation before module output.

Mixed Feature Propagator (MFP). Mixed Feature Propagator (MFP) is a crucial component of the No-MambAAD decoder, as it iteratively updates features at different resolutions. The MFP module consists of several layers, including a convolutional layer, a normalization layer, and an InFR module. Additionally, it includes an upsampling layer, which is invoked only under specific conditions. In our standard convolutional layer, we utilize a kernel size of (3, 3) and a stride of 1. For the upsampling layer, the kernel size is (2, 2) wit the same stride. Normalization is applied in batches. The InFR module aims to extract global features from the convolutional layer and combine them with the local input features. The module name of Mixed Feature Propagator originates from this integration of local and global features.

To summarize, the input to the MFP module first goes

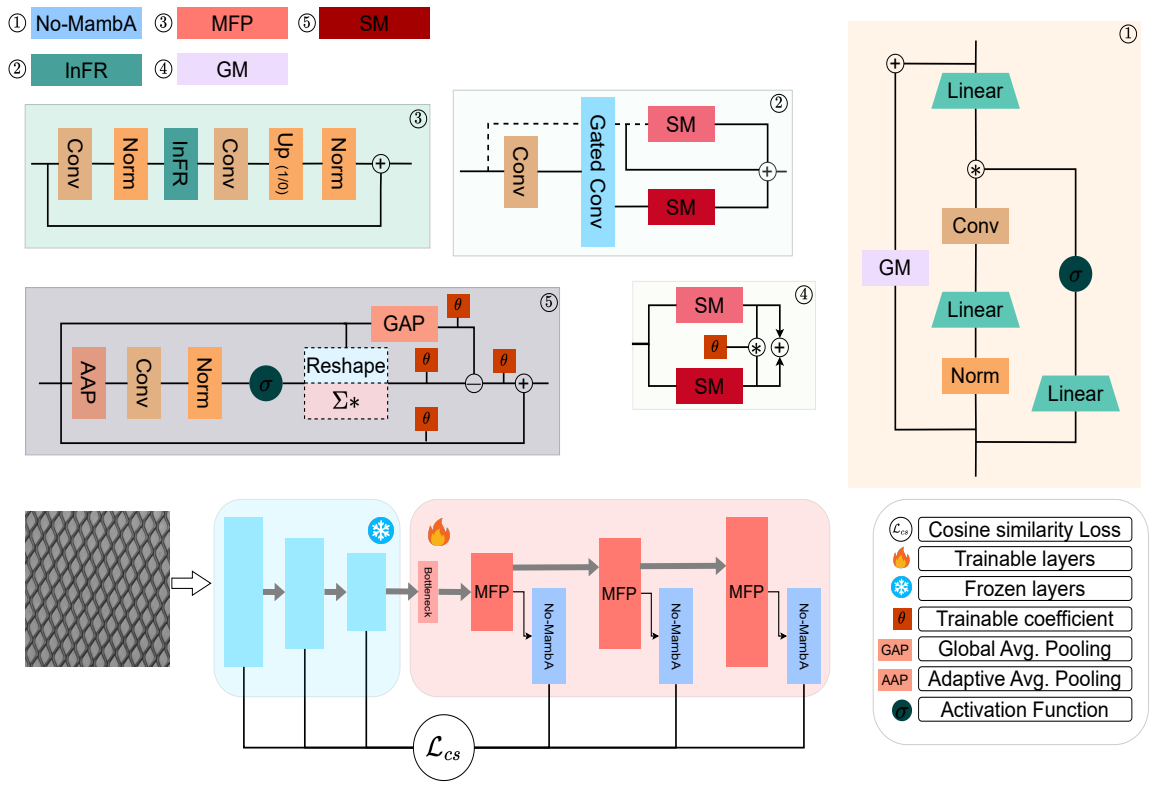

Figure 2. No-MambAAD flow diagram. Module names corresponding to numbers explained on the top of the figure.

through a standard convolution layer followed by a normalization layer. Next, it proceeds to the step within the InFR block. The output from the InFR block connects into another round of convolution and normalization before entering the final residual operation. If necessary, an upsampling operation is performed in between these steps.

Global Mixer (GM). The No-MambAAD global mixer module is a straightforward block, which merges strip modulation blocks with some trainable coefficients. As mentioned before, strip modulation applies scanning from different directions to ensure linear compute cost in feature extraction — in the GM module strip modulation is applied horizontally and vertically to capture the global response. Each channel is weighed with a trainable coefficient that prevents direct aggregation between the channels; the channels have a specific scanning direction in the modulation that results in refined frequency representation.

No-MambA. The proposed No-MambA module has two parts: local feature processing and global feature processing. The local feature processing part directly mimics the modeling trait of the standard Mamba block by taking inspiration from the mamba-out [19] block. In the local feature processing part, the input feature is treated as follows: a copy of the features is first normalized, then goes into a linear layer and ends with a convolutional operation. Similarly, another copy goes through dense convolution and activation layers consecutively. Then, the outcome of the convolution and activation layers forms a product, followed by a final dense convolution before aggregating the global feature processing path.

For global feature extraction, the proposed global mixer module is used by adopting a copy of the original feature into it. Finally, the local and global features are aggregated before forwarding them to the loss layer. The No-MambA module works as a feature refiner for the decoder, as the output of each MFP module works in two ways: one to the next MFP layer as the initial raw representation for the next-scale features and another to this No-MambA module, and the output of it is then compared with the features from encoder of the same scale. The placement of each module has been determined empirically.

Decoder. In No-MambAAD, we train the bottleneck and the decoder by maximizing the multi-scale feature cosine similarity. The decoder consists of two modules: the MFP module and the No-MambA module. The MFP module creates initial feature representations at each scale and passes them to the next MFP module and the current No-MambA module. This No-MambA module refines the incoming fea-

tures from the MFP module and passes them to the loss layer without upsampling.

Loss function. The default loss function in No-MambAAD training is cosine similarity, where the similarity is between the encoder and decoder for multi-stage feature resolutions. If E_f and D_f are the stacked feature list of the encoder E and the decoder D, then the Cosine loss is:

$$\mathcal{L}_{sim} = \mathcal{D}_{cos}(\mathcal{F}(E_f), \mathcal{F}(D_f)), \tag{1}$$

Here, $\mathcal{F}(\cdot)$ denotes the flattening of the features, and the equation for the cosine distance $\mathcal{D}_{cos}$ is

$$\mathcal{D}_{cos}(a, b) = 1 - \frac{a^T \cdot b}{\|a\| \ \|b\|}, \tag{2}$$

4. Experiments

The datasets, metrics and baselines of the experimental evaluation of No-MambAAD are presented below.

Datasets: For evaluation, we have used three well-known visual anomaly detection datasets, MVTec-AD [1], VisA [23], and Real-IAD [17]. The MVTec-AD dataset [1] contains 15 categories, including five texture and 10 object classes. It has 3,629 normal images for training and 1,725 for testing, with 467 normal and 1,258 anomalous images. The VisA dataset [23] includes 12 objects, with a training set of 8,659 normal images and a test set of 2,162 images, 962 normal and 1,200 anomalous. The Real-IAD dataset [17] features 30 unique objects and provides 36,465 normal images for training. Its test set consists of 114,585 images, of which 63,256 are normal and 51,329 are anomalous.

Metrics:. For evaluating the performance of our model, we followed [5, 6], and adopted the following metrics: Area Under the Receiver Operating Characteristic Curve (AU-ROC), Average Precision (AP), and F1-score-max (F_1-max). Area Under the Per-Region-Overlap (AU-PRO) metric is used to evaluate the segmentation performance of our model.

Models: To benchmark No-MambAAD, we have considered several SotA studies [6, 7, 9, 15, 18, 21, 22], some of which manifest anomaly detection through reconstruction [6, 15, 18, 21], and the rest use the embedding approach [9, 22].

Implementation details: For implementation, we have taken inspiration from previous SotA studies [5, 6]. No-MambAAD uses 256x256 resolution for all of the experiments. For the encoder and decoder, we choose ResNet34 as our default model. For optimization, the Adam optimizer with WarmCosineScheduler [5] with the base and final learning rate values of 2e-3 and 9e-5, respectively, are used. We set the epoch count as 100 during training.

4.1. Quantitative comparison

As shown in Table 2, the proposed No-MambAAD sets a new benchmark over the previous studies for all three datasets. For instance, on the MVTec dataset, No-MambaAAD achieves 0.2% improvement in the classification and 0.3% improvement in the segmentation performance over the recent SotA approach, MambaAD [6]. One of the key contributions of our study is that our pure conv-only architecture integrates global feature interaction via strip modulation in the spirit of self-attention. Without relying on the self-attention mechanism, our approach outperforms the recent SotA transformer-based ViTAD [21] by achieving 2.2% improvement in segmentation and 0.5% in classification. Compared to the MVTec, the VisA dataset poses significant complexity and challenges, yet our method continues to improve performance, showing its robustness and effectiveness. By analyzing the classification and segmentation performance of MambaAD [6], our study improves the benchmark by 1.3% and 0.22%, respectively. Similarly, our model achieves 5.5% and 7.2% improvement over ViTAD [21] for classification and segmentation tasks. The same trend is present with the Real-IAD dataset. Overall, these quantitative benchmarks show that the proposed No-MambAAD can perform reliably better than concurrent Mamba and transformer alternatives.

4.2. Qualitative demonstration

No-MambAAD visual anomaly localization results with ground truth for the MVTecAD and VisA datasets are shown as heatmaps in Figure 3.

4.3. Parameter efficiency comparison

In an efficiency comparison between state-of-the-art methods, we compared our study with six studies, highlighting classification performance (AU-ROC), segmentation performance (AU-PRO), and the model parameter count. Our findings reveal that the No-MambAAD model meets performance benchmarks set by existing literature and stands out with its efficiency. Specifically, it employs the least number of parameters compared to all other SotA studies documented in Table 1. In comparison to our study, DiAD [7] uses almost 88 times more parameters, and MambaAD [6] uses 1.5 times more, yet our study ranks top when it comes to metric performance. Hence, this table shows that No-MambAAD can deliver high performance for the given tasks while remaining lightweight and more compute-friendly than the concurrent SotA studies.

5. Ablation Studies

This section presents various complementary results that evaluate different aspects of the proposed No-MambAAD. In this section, we present the following: a) Impact of back-

Dataset	Method	Image-level			Pixel-level			
		AUROC	AP	F_1-max	AUROC	AP	F_1-max	AUPRO
MVTec-AD [1]	RD4AD [15]	94.6	96.5	95.2	96.1	48.6	53.8	91.1
	SimpleNet [9]	95.3	98.4	95.8	96.9	45.9	49.7	86.5
	DeSTSeg [22]	89.2	95.5	91.6	93.1	54.3	50.9	64.8
	UniAD [18]	96.5	98.8	96.2	96.8	43.4	49.5	90.7
	DiAD [7]	97.2	99.0	96.5	96.8	52.6	55.5	90.7
	ViTAD [21]	98.3	99.4	97.3	**97.7**	55.3	58.7	91.4
	MambaAD [6]	98.6	99.6	97.8	**97.7**	56.3	59.2	93.1
	No-MambAAD (Ours)	**98.8**	**99.7**	**98.1**	97.4	**59.3**	**60.2**	**93.4**
VisA [23]	RD4AD [15]	92.4	92.4	89.6	98.1	38.0	42.6	**91.8**
	SimpleNet [9]	87.2	87.0	81.8	96.8	34.7	37.8	81.4
	DeSTSeg [22]	88.9	89.0	85.2	96.1	39.6	43.4	67.4
	UniAD [18]	88.8	90.8	85.8	98.3	33.7	39.0	85.5
	DiAD [7]	86.8	88.3	85.1	96.0	26.1	33.0	75.2
	ViTAD [21]	90.5	91.7	86.3	98.2	36.6	41.1	85.1
	MambaAD [6]	94.3	94.5	89.4	**98.5**	39.4	44.0	91.0
	No-MambAAD (Ours)	**95.5**	**95.9**	**91.7**	98.4	**45.6**	**49.5**	91.2
Real-IAD [17]	RD4AD [15]	82.4	79.0	73.9	97.3	25.0	32.7	89.6
	SimpleNet [9]	57.2	53.4	61.5	75.7	2.8	6.5	39.0
	DeSTSeg [22]	82.3	79.2	73.2	94.6	37.9	41.7	40.6
	UniAD [18]	83.0	80.9	74.3	97.3	21.1	29.2	86.7
	DiAD [7]	75.6	66.4	69.9	88.0	2.9	7.1	58.1
	ViTAD [21]	82.3	79.4	73.4	96.9	26.7	34.9	84.9
	MambaAD [6]	86.3	84.6	77.0	**98.5**	33.0	38.7	90.5
	No-MambAAD (Ours)	**87.1**	**85.8**	**78.2**	98.1	**39.8**	**44.1**	**91.1**

Table 2. UAD performance comparison for current SotA models. The best results are boldfaced, second best underlined. From the results above, No-MambAAD achieves SotA classification performance for all three datasets. For the segmentation task, No-MambAAD leads the benchmark for MVTecAD [1] and Real-IAD [17] datasets, while placing second best for the VisA [23] dataset.

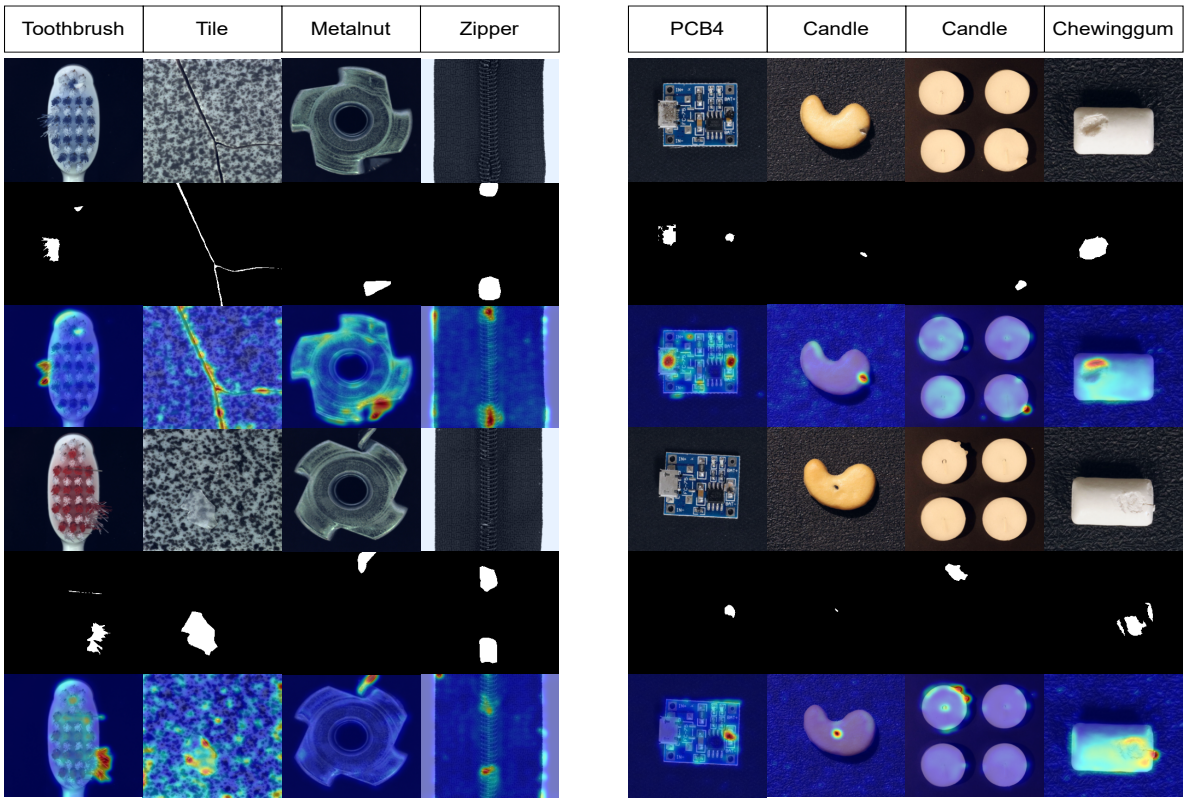

Figure 3. Anomalous sample heatmaps for No-MambAAD, demonstrating the visual anomaly localization performance against ground-truth masks.

Backbone	Decoder Depth	Image-level			Pixel-level				Params(M)	Method
		AU-ROC	AP	F1_max	AU-ROC	AP	F1_max	AU-PRO		
ResNet18	[3,4,6,3]	96.6	98.8	96.4	**96.8**	53.2	56.2	91.8	20.3	MambaAD [6]
		98.2	**99.3**	**97.4**	**96.8**	**58.8**	**60.3**	**92.3**	**9.3**	No-MambAAD
ResNet34	[3,4,6,3]	98.6	**99.6**	97.8	97.7	56.3	59.2	**93.1**	25.7	MambaAD [6]
		98.7	**99.6**	**98.1**	97.7	**63.2**	**63.3**	93.0	**15.1**	No-MambAAD
ResNet50	[3,4,6,3]	98.4	99.4	97.7	97.7	54.2	57.0	92.3	251.0	MambaAD [6]
		98.8	**99.6**	**98.2**	**97.8**	**63.3**	**63.3**	**94.1**	**34.5**	No-MambAAD
WideResNet50	[3,4,6,3]	98.6	99.5	**98.0**	**98.0**	57.9	60.3	93.8	268.0	MambaAD [6]
		98.9	**99.6**	97.8	97.8	**63.5**	**63.7**	**94.4**	**65.3**	No-MambAAD

Table 3. Comparison of backbone impact between MambaAD [6] and the proposed No-MambAAD method. For this experiment, identical decoder depth was kept for each backbone model together with three-stage feature matching.

Dataset	Inference	AU-ROC (I)	AU-PRO (P)
MVTecAD [1]	3 stage	98.7	93.0
	4 stage	**98.8**	**93.4**
VisA [23]	3 stage	95.2	91.0
	4 stage	**95.5**	**91.2**

Table 4. The impact of feature matcher stages for No-MambAAD on MVTecAD and VisA datasets. For both datasets, four stages of feature matching consistently improves over three-stages in both classification and segmentation tasks.

Method	#parameters (M)	AU-ROC (I)	AU-PRO (P)
UniAD [18]	24.5	96.5	90.7
RD4AD [15]	80.6	94.6	91.1
DeSTSeg [22]	35.2	89.2	64.8
SimpleNet [9]	72.8	95.3	86.5
DiAD [7]	1331.3	97.2	90.7
ViTAD [21]	38.6	983	91.4
MambaAD [6]	25.7	98.6	93.1
Ours	**15.8**	**98.8**	**93.4**

Table 5. Comparison of parameter efficiency between SotA studies. Here we have used four stages of feature matching between the encoder and decoder. As a result, the number of parameters for No-MambAAD is slightly higher than the reported parameter count in the Table 3.

Dataset	InFR	No-MambA	AU-ROC (I)	AU-PRO (P)
MVTecAD [1]	✔	✖	98.1	92.3
	✖	✔	98.3	92.8
	✔	✔	**98.8**	**93.4**

Table 6. Impact of deactivating the novel modules, InFR and No-MambA, within our model. Turning off the InFR module results in a more pronounced decline in performance than deactivating the No-MambA module, thereby confirming the design choice of including the No-MambA module as the feature refiner. Enabling both, the SotA result in the MVTecAD dataset is achieved.

bones, b) Trade-offs between feature matching stages, and c) Module effectiveness.

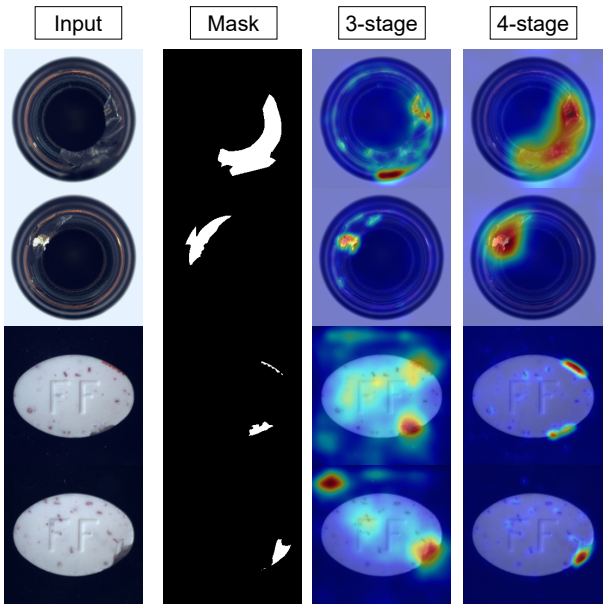

Figure 4. Visual effect of varying the number of feature-matching stages in the No-MambAAD model, when evaluated on the MVTecAD dataset. The visual analysis reveals that an increase in the number of feature-matching stages enhances the precision of anomaly localization, ultimately yielding segmentation outcomes that are more consistent than those achieved with the baseline three-stage configuration.

5.1. Effect of different backbones

Here, the No-MambAAD performance scaling as a function of pre-trained backbone model is measured using the MVTecAD dataset. For this ablation, various ResNet models were used as backbone alternatives, specifically ResNet18, ResNet34, ResNet50, and WideResNet50, as shown in Table 3. Our findings show that ResNet18 delivers the lowest performance among the tested backbones while having the fewest parameters and the lowest training time. However, when comparing our model's performance with the same variant of MambaAD [6], we achieve **1.76%**

higher performance.

The trend is also consistent across other models. For example, using the ResNet34 backbone, our approach has a **44%** lower parameter count than MambaAD [6] while still outperforming it. Additionally, in the case of WideResNet50, the parameter ratio between MambaAD [6] and No-MambAAD is **4:1**; however, No-MambAAD achieves a performance that is **0.3%** higher than that of MambaAD [6].

Likewise, the decoder depth configuration was altered for the mentioned backbones, and with all of the variants, MambaAD was outperformed with significantly fewer parameters. From ResNet18 to WideResNet50, a consistent improvement in performance was observed for No-MambAAD, indicating that our model can scale and generalize effectively with larger and better backbones. In contrast, MambaAD deviates from this trend, as it requires significantly more computational resources when targeting higher AD performance. After analyzing the results and parameter counts, ResNet34 was selected as the default backbone for all the results presented in this paper. However, to ensure a fair comparison with MambaAD, we employed three stages of feature matching during the ablation study.

5.2. Generalization and training stage tradeoffs

Figure 4 illustrates pixel-level anomaly segmentation on the MVTec dataset for No-MambAAD with different numbers of feature matching stages. To make a fair comparison, the training setup was kept identical in all cases and the same ResNet34 encoder was used. The results illustrate that increasing the number of feature-matching stages can increase the anomaly detection performance and improve the pixel-level segmentation performance. For additional results, a short comparison for the MVTecAD and VisA datasets is shown in Table 4.

Accordingly, as illustrated in Figure 4, the proposed No-MambAAD achieves a superior segmentation outcome by employing a four-stage architecture that maximizes cosine similarity between the encoder and decoder features, outperforming the conventional three-stage configuration. The improved performance comes with a computational cost, as the four-stage method requires more parameters and a longer training time than the three-stage setup. However, No-MambAAD consistently shows significantly lower computational demands across all configurations when compared to the current SotA methods.

5.3. Impact of different modules

In Table 6, we present the ablation on the impact of internal modules of the No-MambAAD model. Likewise, for this ablation, we have kept the original training setup and used the ResNet34 encoder with three stages of feature matching. The InFR and No-MambAAD modules were disabled one at a time and the performance impact on MVTEC AD was measured. Table 6 shows that both the InFR and No-MambA modules are required for achieving the best UAD classification performance.

6. Conclusion

We propose No-MambAAD, a convolution-only method for multiclass unsupervised anomaly detection that matches or outperforms SotA alternatives that are based on Transformers or Mamba. Our decoder utilizes a novel Mixed Feature Propagator (MFP) module for aggregating global features during standard local feature propagation, followed by the No-MambA module, which refines the aggregated mixed features. This simplified decoder requires fewer parameters than recent SotA studies while maintaining consistent performance across diverse datasets. In our future work, we plan to expand this research to include cross-domain and cross-modality anomaly detection.

Acknowledgement

This work has been supported by the ERDF project EURA 2021/403559/09 02 01 01/2023/EPL "TENTA".

References

[1] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9592–9600, 2019. 1, 5, 6, 7

[2] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4183–4192, 2020. 1, 2

[3] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: a patch distribution modeling framework for anomaly detection and localization. In *International conference on pattern recognition*, pages 475–489. Springer, 2021. 1, 2

[4] Hanqiu Deng and Xingyu Li. Anomaly detection via reverse distillation from one-class embedding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9737–9746, 2022. 1, 2

[5] Jia Guo, Shuai Lu, Weihang Zhang, Fang Chen, Hongen Liao, and Huiqi Li. Dinomaly: The less is more philosophy in multi-class unsupervised anomaly detection. *arXiv preprint arXiv:2405.14325*, 2024. 1, 2, 5

[6] Haoyang He, Yuhu Bai, Jiangning Zhang, Qingdong He, Hongxu Chen, Zhenye Gan, Chengjie Wang, Xiangtai Li, Guanzhong Tian, and Lei Xie. Mambaad: Exploring state space models for multi-class unsupervised anomaly detection. *arXiv preprint arXiv:2404.06564*, 2024. 1, 2, 5, 6, 7, 8

[7] Haoyang He, Jiangning Zhang, Hongxu Chen, Xuhai Chen, Zhishan Li, Xu Chen, Yabiao Wang, Chengjie Wang, and Lei

Xie. A diffusion-based framework for multi-class anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 8472–8480, 2024. 5, 6, 7

[8] Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9664–9674, 2021. 1, 2

[9] Zhikang Liu, Yiming Zhou, Yuansheng Xu, and Zilei Wang. Simplenet: A simple network for image anomaly detection and localization. *arXiv preprint arXiv:2303.15140*, 2023. 1, 2, 3, 5, 6, 7

[10] Ruiying Lu, YuJie Wu, Long Tian, Dongsheng Wang, Bo Chen, Xiyang Liu, and Ruimin Hu. Hierarchical vector quantized transformer for multi-class unsupervised anomaly detection. *Advances in Neural Information Processing Systems*, 36:8487–8500, 2023. 2

[11] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2

[12] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14318–14328, 2022. 1, 2

[13] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14318–14328, 2022. 2

[14] Marco Rudolph, Bastian Wandt, and Bodo Rosenhahn. Same same but differnet: Semi-supervised defect detection with normalizing flows. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1907–1916, 2021. 2

[15] Mohammadreza Salehi, Niousha Sadjadi, Soroosh Baselizadeh, Mohammad H Rohban, and Hamid R Rabiee. Multiresolution knowledge distillation for anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14902–14912, 2021. 1, 2, 3, 5, 6, 7

[16] Hannah M Schlüter, Jeremy Tan, Benjamin Hou, and Bernhard Kainz. Natural synthetic anomalies for self-supervised anomaly detection and localization. In *European Conference on Computer Vision*, pages 474–489. Springer, 2022. 2

[17] Chengjie Wang, Wenbing Zhu, Bin-Bin Gao, Zhenye Gan, Jianning Zhang, Zhihao Gu, Shuguang Qian, Mingang Chen, and Lizhuang Ma. Real-iad: A real-world multi-view dataset for benchmarking versatile industrial anomaly detection. *arXiv preprint arXiv:2403.12580*, 2024. 5, 6

[18] Zhiyuan You, Lei Cui, Yujun Shen, Kai Yang, Xin Lu, Yu Zheng, and Xinyi Le. A unified model for multi-class anomaly detection. *arXiv preprint arXiv:2206.03687*, 2022. 1, 2, 3, 5, 6, 7

[19] Weihao Yu and Xinchao Wang. Mambaout: Do we really need mamba for vision? *arXiv preprint arXiv:2405.07992*, 2024. 2, 4

[20] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Draema discriminatively trained reconstruction embedding for surface anomaly detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8330–8339, 2021. 1, 2, 3

[21] Jiangning Zhang, Xuhai Chen, Yabiao Wang, Chengjie Wang, Yong Liu, Xiangtai Li, Ming-Hsuan Yang, and Dacheng Tao. Exploring plain vit reconstruction for multi-class unsupervised anomaly detection. *arXiv preprint arXiv:2312.07495*, 2023. 2, 3, 5, 6, 7

[22] Xuan Zhang, Shiyu Li, Xi Li, Ping Huang, Jiulong Shan, and Ting Chen. Destseg: Segmentation guided denoising student-teacher for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3914–3923, 2023. 1, 2, 3, 5, 6, 7

[23] Yang Zou, Jongheon Jeong, Latha Pemula, Dongqing Zhang, and Onkar Dabeer. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In *European Conference on Computer Vision*, pages 392–408. Springer, 2022. 5, 6, 7