MODULATION-ONLY FEATURE RECONSTRUCTION FOR UNIFIED ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

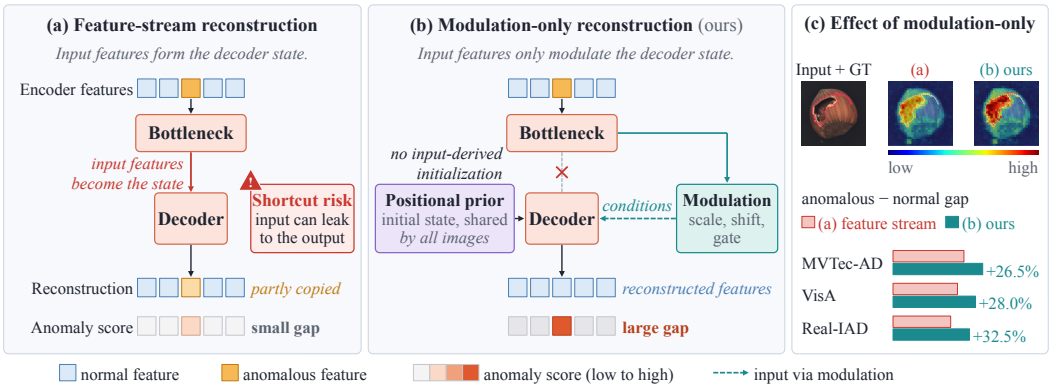

Figure 1: Reconstruction architectures by the pathway of input information. (a) Feature-stream reconstruction initializes the decoder state from a representation obtained from the input features. (b) MOFR initializes the state with positional embeddings shared across images and updates it with modulation coefficients generated from the input features. The two differ in whether input information is contained in the initial state or delivered through conditional updates. (c) Mean error gap between anomalous and normal regions of the two architectures (Sec. 4.4).

ABSTRACT

Feature reconstruction for anomaly detection uses the difference between the input features and a reconstruction learned from normal data. We propose Modulation-Only Feature Reconstruction (MOFR), which pairs an image-independent initial state with per-token feature conditioning. The decoder of MOFR starts from learnable positional embeddings shared by all images, and the input features generate the modulation coefficients of the residual updates and of the output head instead of forming the initial state. Using the per-token conditions and a global condition obtained from their mean, the decoder reconstructs features without token-to-token attention. On MVTec-AD, VisA, and Real-IAD, MOFR reaches image-level AUROCs of 99.8%, 99.0%, and 91.1% and AUPROs of 96.1%, 95.8%, and 96.3%. Crossing the initialization pathway with per-token conditioning shows that the contribution of per-token conditioning depends strongly on whether the initial state contains the input features. The differences from feature-stream reconstruction, which uses an input-derived state, are small and vary with the dataset and the metric. These results show that pairing a shared initial state with per-token conditioning is a reconstruction scheme that replaces input-derived initialization with conditional updates while retaining strong anomaly detection performance.

1 INTRODUCTION

Industrial anomaly detection learns from normal images only and must detect and localize defects that were not observed during training. Unified anomaly detection, which handles multiple product categories with one model, reduces the burden of maintaining a separate model per category, but the model must distinguish local defects from diverse normal appearances at the same time (You et al., 2022).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Methods that reconstruct pretrained features use the difference between the input features and the reconstructed features as the anomaly score. What matters is not a low reconstruction error overall but an error that separates normal from anomalous features. A reconstructor that preserves the input features well can restore defects as well as normal patterns. Prior work has mitigated this with feature perturbation, attention constraints, noisy bottlenecks, and reference representations (You et al., 2022; He et al., 2024c; Guo et al., 2025b; Luo et al., 2025).

This work focuses on the pathway through which input information reaches the decoder (Fig. 1). In feature-stream reconstruction, which uses an input-derived representation as the initial state, the decoder refines a state that already contains image-specific content. If the decoder instead starts from an image-independent state, the position-wise content of the input must be formed through conditional updates. This difference is a design choice for the reconstruction architecture that is separate from the dimensionality or the precision of the bottleneck.

To realize this, we propose Modulation-Only Feature Reconstruction (MOFR). The decoder starts from learnable positional embeddings shared by all images. The features of a frozen encoder pass through a shared bottleneck and become per-token condition codes, and each code together with a global condition obtained from the mean of the codes generates the shift, scale, and gate coefficients of a residual MLP. The global condition is also applied to the output head. The decoder has no token-to-token attention, and spatial context is provided through the encoder features and the global condition. In this architecture, modulation-only means that the input features are not injected directly into the initial decoder state but are delivered through the conditioning pathway.

On three industrial anomaly detection datasets, MOFR shows high detection and localization performance. In a cross comparison of the initial state and per-token conditioning, adding per-token conditioning to an input-derived initial state has a small effect, whereas per-token conditioning makes a large difference with the shared initial state. The pairing of a shared initial state and per-token conditioning can therefore be interpreted as an alternative to input-derived initialization. Since the improvements over feature-stream reconstruction are small and vary with the dataset and the metric, we do not claim a general superiority of the shared initial state.

Our contributions are as follows.

- We propose a feature reconstruction architecture that pairs an image-independent initial state with per-token feature conditioning and replaces input-derived initialization with conditional updates.
- We implement an attention-free decoder in which per-token and global conditions regulate the residual MLPs and the output head.
- We examine the performance and the scope of the proposed architecture through comparisons on three datasets, a cross analysis of the initialization pathway and conditioning, and a reconstruction error analysis.

2 Related Work

2.1 Feature-Based Unified Anomaly Detection

Anomaly detection methods that use pretrained features either model the distribution of normal features or use feature prediction and reconstruction errors. PatchCore uses a memory bank of normal patch features (Roth et al., 2022), and Reverse Distillation uses the difference between encoder features and restored features (Deng & Li, 2022). In unified anomaly detection, one model learns the normal distributions of multiple categories. UniAD addressed the identity shortcut in this setting, in which both normal and anomalous inputs are restored (You et al., 2022), and Dinomaly combined a pretrained Transformer with a noisy bottleneck, linear attention, and group-wise reconstruction (Guo et al., 2025b). Dinomaly2 extended this line of work with CLS-based context conditioning (Guo et al., 2025a). MOFR addresses the pathway through which the input features enter the decoder state in this feature reconstruction setting.

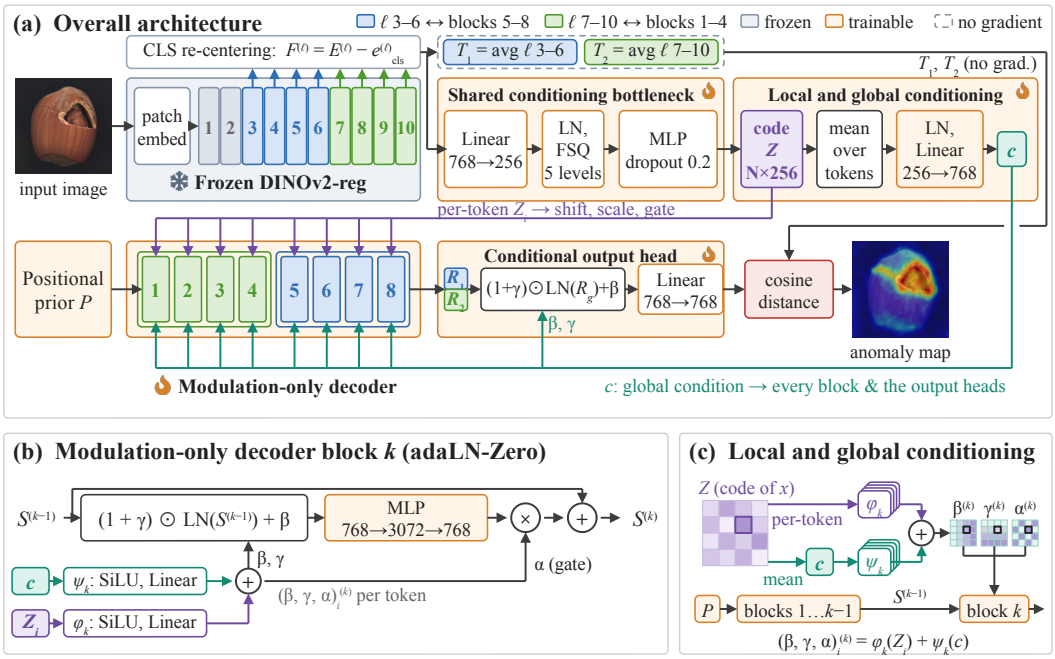

Figure 2: Overall architecture and conditional updates of MOFR. (a) The features of the frozen encoder give the reconstruction targets T_1, T_2 and the condition codes. The decoder starts from the shared positional embedding P and is updated with the per-token conditions Z_i and the global condition c. (b) Each block modulates the normalized state, gates the MLP output, and adds it to the residual state. (c) The modulation coefficients generated from the per-token and global conditions are summed. The mean of blocks 5–8 reconstructs T_1 and the mean of blocks 1–4 reconstructs T_2.

2.2 Information Transfer in Reconstruction

UniAD limits the direct transfer of the input with learnable queries, neighbor-masked attention, and feature jittering (You et al., 2022). RLR reconstructs features through learnable reference representations (He et al., 2024c), and INP-Former provides the decoder with normal prototypes extracted from the input image (Luo et al., 2025). MOFR updates a shared initial state with conditions generated from the input. Image-specific information is delivered through the modulation coefficients, and the residual connections of the decoder accumulate the state formed by the conditional updates.

2.3 Conditional Modulation and Quantization

FiLM generates feature-wise scale and shift coefficients from a condition (Perez et al., 2018), and the adaLN-Zero of DiT combines the modulation of normalized features with residual gating (Peebles & Xie, 2023). MOFR applies this conditioning scheme to token-wise residual MLPs. Finite scalar quantization (FSQ) quantizes each coordinate to a finite set of values without a learnable vector codebook (Mentzer et al., 2024). We apply FSQ to the conditioning bottleneck and analyze the effect of quantization separately from the effect of the input pathway.

3 Method

MOFR is a feature reconstruction model trained on the normal images of multiple categories. It obtains the reconstruction targets and the condition codes from a frozen encoder and restores features by conditionally updating an initial state shared across images (Fig. 2). At inference, the pixel anomaly map and the image anomaly score are computed from the difference between the input features and the reconstructed features.

3.1 Reconstruction Targets

We use DINOv2 ViT-B/14 with four register tokens as the encoder and freeze its weights (Oquab et al., 2024; Darcet et al., 2024). For an input image x of size 392×392, the number of patches is $N = 28 \times 28 = 784$ and the feature dimension is $D = 768$. In the equations below, i denotes a patch position. The CLS and register tokens are excluded from the mean pooling of the global condition, the reconstruction loss, and the aggregation of the anomaly score.

We use the features of encoder blocks 3 to 10. Let $E_i^{(\ell)}$ and $e_{\mathrm{cls}}^{(\ell)}$ be the patch feature and the CLS feature of block ℓ. The reconstruction targets are defined by CLS centering and group-wise averaging (Guo et al., 2025a;b):

$$F_i^{(\ell)} = E_i^{(\ell)} - e_{\mathrm{cls}}^{(\ell)}, \qquad T_{g,i} = \frac{1}{4} \sum_{\ell \in \mathcal{L}_g} F_i^{(\ell)}. \tag{1}$$

Here $\mathcal{L}_1 = \{3, 4, 5, 6\}$ and $\mathcal{L}_2 = \{7, 8, 9, 10\}$, and $T_1, T_2 \in \mathbb{R}^{N \times D}$. The conditioning bottleneck receives the mean of the eight layers, $\bar{F} = (T_1 + T_2)/2$.

3.2 Shared Conditioning Bottleneck

Each patch feature is projected to $d = 256$ dimensions and normalized, and FSQ with five levels is applied to each coordinate (Mentzer et al., 2024):

$$U_i = \mathrm{LN}_q\big(W_{\mathrm{down}} \bar{F}_i + b_{\mathrm{down}}\big), \qquad Q_i = \tfrac{1}{2} \operatorname{round}(2 \tanh U_i), \qquad Z_i = \mathrm{MLP}_{\mathrm{bn}}(Q_i). \tag{2}$$

Each coordinate of Q_i takes one of $\{-1, -\frac{1}{2}, 0, \frac{1}{2}, 1\}$. A straight-through estimator is applied to the rounding operation during training (Bengio et al., 2013), and quantization is kept at inference. The bottleneck MLP has the structure $256 \to 1024 \to 256$ with GELU. Dropout with probability 0.2 is applied to its input, its GELU output, and its final output. The condition code Z_i produced by the MLP is not restricted to five levels.

The per-token condition is Z_i, and the global condition $c \in \mathbb{R}^D$ is generated from the mean of the patch codes:

$$c = W_c \, \mathrm{LN}_c\left(\frac{1}{N} \sum_{i=1}^{N} Z_i\right) + b_c. \tag{3}$$

LN_q and LN_c have learnable affine parameters, and W_c projects d dimensions to D dimensions. The per-token conditions provide position-wise content, and the global condition provides the context of the whole image.

3.3 Modulation-Only Decoder

The initial state of the decoder is a learnable positional embedding $P \in \mathbb{R}^{N \times D}$ shared by all images. P may differ across positions but does not depend on the input image. Starting from $S^{(0)} = P$, $L = 8$ blocks are applied. Each block generates a shift $\beta_i^{(k)}$, a scale $\gamma_i^{(k)}$, and a residual gate $\alpha_i^{(k)}$ from the per-token and global conditions:

$$\left[\beta_i^{(k)}, \gamma_i^{(k)}, \alpha_i^{(k)}\right] = \phi_k(Z_i) + \psi_k(c). \tag{4}$$

ϕ_k and ψ_k consist of SiLU and a linear layer and map $256 \to 2304$ and $768 \to 2304$. $\psi_k(c)$ is shared by all patch positions. Every block uses the same Z and c but has its own parameters. The state is updated as

$$V_i^{(k)} = \big(1 + \gamma_i^{(k)}\big) \odot \mathrm{LN}\Big(S_i^{(k-1)}\Big) + \beta_i^{(k)}, \qquad S_i^{(k)} = S_i^{(k-1)} + \alpha_i^{(k)} \odot \mathrm{MLP}_k\Big(V_i^{(k)}\Big). \tag{5}$$

Each MLP has the structure $768 \to 3072 \to 768$ with GELU and operates on each token independently. The state normalization has no affine parameters, and the range of the gate is not restricted. The modulation generators are initialized to zero so that the residual update is zero at the start of training (Peebles & Xie, 2023).

The outputs are aggregated into two groups by decoder depth. Matching encoder and decoder depths in reverse order, $R_1 = \frac{1}{4}\sum_{k=5}^{8} S^{(k)}$ reconstructs T_1 and $R_2 = \frac{1}{4}\sum_{k=1}^{4} S^{(k)}$ reconstructs T_2 (Guo et al., 2025b). The output head of each group modulates the normalized state with the global condition:

$$\left[\beta_g^{\text{out}}, \gamma_g^{\text{out}}\right] = \eta_g(c), \qquad \widehat{T}_{g,i} = W_g^{\text{out}}\left[\left(1+\gamma_g^{\text{out}}\right) \odot \text{LN}(R_{g,i}) + \beta_g^{\text{out}}\right] + b_g^{\text{out}}. \tag{6}$$

η_g consists of SiLU and a $768 \to 1536$ linear layer. The output projection is $768 \to 768$, with its weight initialized to the identity matrix and its bias to zero. The output modulation generator is also initialized to zero.

The decoder has neither self-attention nor cross-attention. Image-dependent information is delivered through the modulation coefficients of the blocks and the output head. Since the per-token conditions contain the context formed in the encoder and the global condition is shared by all positions, the token-wise operation of the decoder does not mean an absence of image context. With the feature dimension and the depth fixed, the decoder cost grows linearly with the number of patches.

3.4 Training Objective

With batch size B, we minimize the cosine reconstruction loss over all patches of normal images:

$$\delta_{bgi} = 1 - \cos\left(T_{g,b,i}, \widehat{T}_{g,b,i}\right), \qquad \mathcal{L}_{\text{rec}} = \frac{1}{2BN}\sum_{b=1}^{B}\sum_{g=1}^{2}\sum_{i=1}^{N}\delta_{bgi}. \tag{7}$$

The vector norms in the cosine similarity are bounded below by 10^{-8}. No gradient is passed to the encoder or the reconstruction targets, and the bottleneck, the positional embedding, the decoder, and the output heads are trained.

To give relatively larger gradients to normal patches that are hard to reconstruct, we apply gradient modulation by error rank (Guo et al., 2025b). In each feature group, the output gradient of the patches with small errors is scaled by 0.1 and the other patches keep their gradients. The proportion of modulated patches increases as training proceeds, and after 62.5% of training roughly 10% of the patches with the largest errors keep their original gradients. All patches remain in the loss, and this operation does not change the forward loss value.

3.5 Anomaly Scoring

At inference, dropout is disabled and quantization is kept. The patch anomaly score of image x is the mean cosine distance of the two feature groups:

$$a_i(x) = \frac{1}{2}\sum_{g=1}^{2}\left[1 - \cos\left(T_{g,i}(x), \widehat{T}_{g,i}(x)\right)\right]. \tag{8}$$

Bilinear interpolation of the patch score grid, followed by Gaussian smoothing with a 5×5 kernel of standard deviation 4, gives the pixel anomaly map $A(x)$. The image score $s(x)$ is the mean of the top 1% of the values in $A(x)$. Larger values indicate higher anomaly.

4 Experiments

4.1 Experimental Setup

We evaluate on the 15 categories of MVTec-AD (Bergmann et al., 2019), the 12 categories of VisA (Zou et al., 2022), and the 30 categories of Real-IAD (Wang et al., 2024). For each dataset, one model is trained on the normal images of all categories together, without anomalous images or category conditions. On Real-IAD, the image of each camera view is an individual evaluation unit.

Input images are resized to 448×448 and the central 392×392 region is used. The batch size is 16, and the model is updated 14,000 times on MVTec-AD and VisA and 48,000 times on Real-IAD. We use StableAdamW (Wortsman et al., 2023), and the learning rate decays with a cosine schedule

Table 1: Unified anomaly detection performance (%). Image-level entries are I-AUROC / I-AP / I-F1max and pixel-level entries are P-AUROC / P-AP / P-F1max / AUPRO. The highest value of each metric is in bold, and values equal at the displayed precision share the mark.

	MVTec-AD		VisA		Real-IAD	
Method	Image-level	Pixel-level	Image-level	Pixel-level	Image-level	Pixel-level
RD4AD	94.6/96.5/95.2	96.1/48.6/53.8/91.1	92.4/92.4/89.6	98.1/38.0/42.6/91.8	82.4/79.0/73.9	97.3/25.0/32.7/89.6
UniAD	96.5/98.8/96.2	96.8/43.4/49.5/90.7	88.8/90.8/85.8	98.3/33.7/39.0/85.5	83.0/80.9/74.3	97.3/21.1/29.2/86.7
SimpleNet	95.3/98.4/95.8	96.9/45.9/49.7/86.5	87.2/87.0/81.8	96.8/34.7/37.8/81.4	57.2/53.4/61.5	75.7/2.8/6.5/39.0
DeSTSeg	89.2/95.5/91.6	93.1/54.3/50.9/64.8	88.9/89.0/85.2	96.1/39.6/43.4/67.4	82.3/79.2/73.2	94.6/37.9/41.7/40.6
DiAD	97.2/99.0/96.5	96.8/52.6/55.5/90.7	86.8/88.3/85.1	96.0/26.1/33.0/75.2	75.6/66.4/69.9	88.0/2.9/7.1/58.1
MambaAD	98.6/99.6/97.8	97.7/56.3/59.2/93.1	94.3/94.5/89.4	98.5/39.4/44.0/91.0	86.3/84.6/77.0	98.5/33.0/38.7/90.5
Dinomaly	99.6/99.8/99.0	98.4/69.3/69.2/94.8	98.7/98.9/96.2	98.7/**53.2**/**55.7**/94.5	89.3/86.8/80.2	98.8/42.8/47.1/93.9
INP-Former	99.7/**99.9**/99.2	98.5/**71.0**/**69.7**/94.9	98.9/99.0/**96.6**	98.9/51.2/54.7/94.4	90.5/88.1/81.5	99.0/47.5/50.3/95.0
MOFR	**99.8**/**99.9**/**99.3**	**98.6**/70.1/69.3/**96.1**	**99.0**/**99.1**/**96.6**	**99.2**/52.9/**55.7**/**95.8**	**91.1**/**88.9**/**81.7**	**99.2**/**48.2**/**51.0**/**96.3**

from 2×10^{-3} to 2×10^{-4} after an initial warm-up. No normal reference images or memory bank are used at inference.

The image-level metrics are I-AUROC, I-AP, and I-F1max, and the pixel- and region-level metrics are P-AUROC, P-AP, P-F1max, and AUPRO (Bergmann et al., 2020). F1max is the maximum F1 over score thresholds, and the pixel map is evaluated at 256×256. Each metric is computed per category and averaged with equal weights. All results are reported in percent and differences in percentage points (pp).

4.2 Comparison with Existing Methods

Table 1 compares MOFR with RD4AD (Deng & Li, 2022), UniAD (You et al., 2022), SimpleNet (Liu et al., 2023), DeSTSeg (Zhang et al., 2023), DiAD (He et al., 2024b), MambaAD (He et al., 2024a), Dinomaly (Guo et al., 2025b), and INP-Former (Luo et al., 2025). The values of the compared methods are taken from the literature.

The I-AUROC of MOFR is 99.8%, 99.0%, and 91.1% on MVTec-AD, VisA, and Real-IAD, the highest among the compared methods in Table 1. AUPRO is also high at 96.1%, 95.8%, and 96.3%. The advantage in the localization metrics is not uniform, however. On MVTec-AD, P-AP and P-F1max are 0.9 and 0.4 pp below INP-Former, and on VisA, P-AP is 0.3 pp below Dinomaly. On Real-IAD, MOFR has the highest value in every metric of the table. These results show that the features of diverse categories can be reconstructed without an input-derived initial state while keeping high anomaly detection performance.

Fig. 3 shows localization results for representative defects. MOFR responds to local defects, although some responses remain in normal regions. The visualization is used to inspect the position and the shape of the detection responses, and the methods are compared by the quantitative metrics of Table 1.

4.3 Initial State and Per-Token Conditioning

Table 2: Cross comparison of the initial state and per-token conditioning (%). Entries are I-AUROC / P-AUROC / AUPRO. The last column is the equal-weight mean of the three datasets for each metric. Best values in bold.

Initial state	Per-token cond.	MVTec-AD	VisA	Real-IAD	Mean
$U(Z)$	—	99.78 / 98.46 / 95.58	99.00 / 99.03 / 95.85	91.05 / 99.19 / **96.35**	96.61 / 98.89 / 95.93
$U(Z)$	✓	99.77 / 98.45 / 95.53	98.97 / 99.04 / **95.88**	**91.13** / 99.20 / **96.35**	96.62 / 98.90 / 95.92
P	—	97.57 / 95.78 / 89.03	90.14 / 96.35 / 84.25	88.60 / 98.29 / 93.43	92.10 / 96.81 / 88.90
P (MOFR)	✓	**99.82** / **98.59** / **96.09**	**99.02** / **99.16** / 95.84	91.12 / **99.22** / 96.27	**96.65** / **98.99** / **96.07**

Table 2 crosses the initial state with per-token conditioning. $U(Z) = W_{\text{init}} Z$ is an input-derived initial state that projects the bottleneck code to D dimensions, and P is the positional embedding

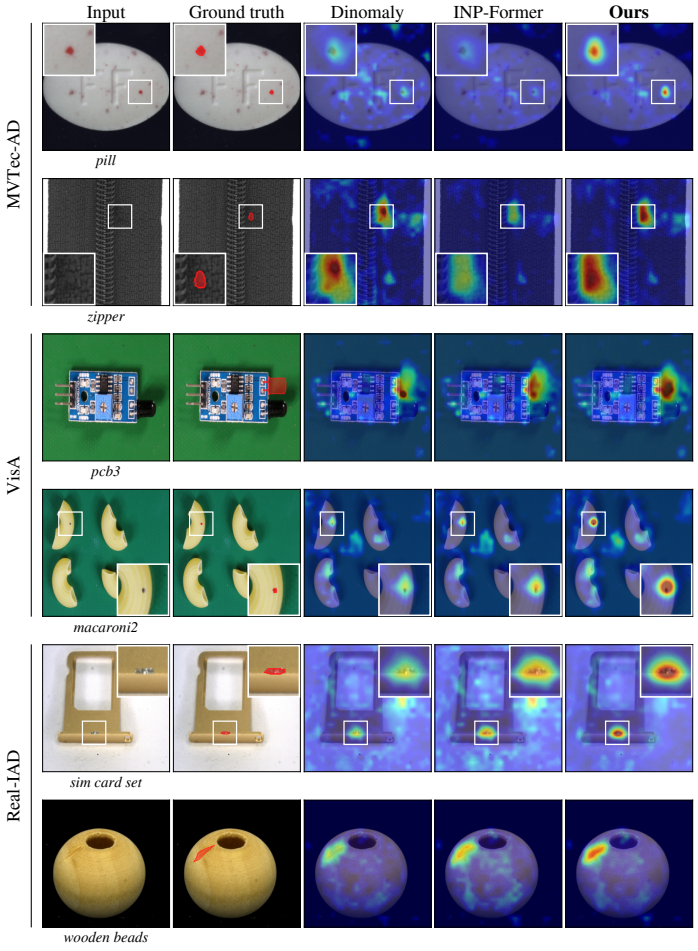

Figure 3: Final pixel anomaly maps for representative defects. Columns are the input image, the ground-truth mask, Dinomaly, INP-Former, and MOFR. Rows are *pill* and *zipper* of MVTec-AD, *pcb3* and *macaroni2* of VisA, and *sim card set* and *wooden beads* of Real-IAD. Boxes mark zoomed regions.

shared across images. All configurations use the shared bottleneck and the global condition. The configuration with P and without per-token conditioning therefore still receives image information through the global condition. We define the configuration with $U(Z)$ and without per-token conditioning as the feature-stream baseline.

Adding per-token conditioning to the input-derived state $U(Z)$ changes the three metrics by at most 0.08 pp on every dataset. With the shared state P, adding per-token conditioning raises I-AUROC by 2.25, 8.88, and 2.52 pp on MVTec-AD, VisA, and Real-IAD. The contribution of per-token conditioning thus differs greatly between the case in which the position-wise content of the input is already contained in the initial state and the case in which it must be delivered through the conditioning pathway.

Comparing MOFR with feature-stream, the I-AUROC differences are $+0.04$, $+0.02$, and $+0.07$ pp and the P-AUROC differences are $+0.13$, $+0.13$, and $+0.03$ pp. AUPRO is 0.51 pp higher on MVTec-AD but 0.01 and 0.08 pp lower on VisA and Real-IAD. In addition, the highest I-AUROC on Real-IAD is obtained by the configuration that adds per-token conditioning to $U(Z)$. This comparison therefore supports that a shared initial state with per-token conditioning can replace input-derived initialization, rather than a universal superiority of P.

4.4 Reconstruction Errors of Normal and Anomalous Regions

Fig. 4 analyzes the raw patch errors before interpolation and smoothing. Patches with no defect pixels are normal patches, and patches with a defect occupancy of at least 50% are anomalous

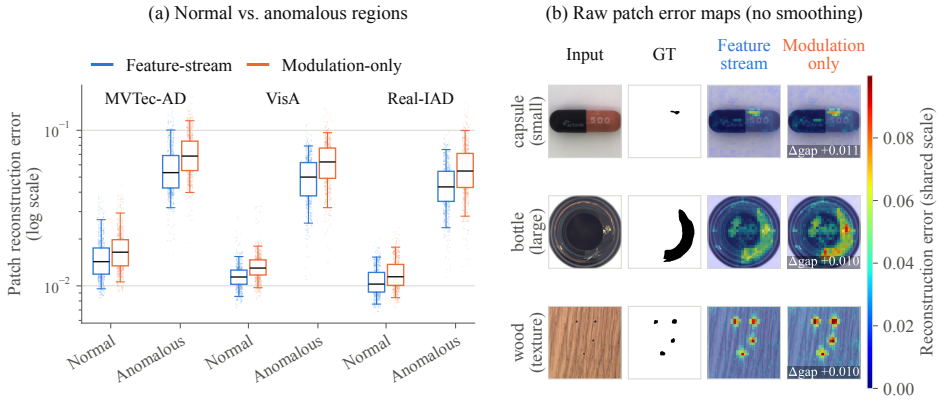

Figure 4: Raw reconstruction errors of normal and anomalous regions. (a) Per-image mean errors of normal (0% defect occupancy) and anomalous (≥50%) patches on a log scale. (b) Patch errors of MVTec-AD examples before interpolation and smoothing on a common color range. Δgap is the gap difference between MOFR and feature-stream, $(r_A - r_N)_{\text{MOFR}} - (r_A - r_N)_{\text{FS}}$.

patches. For each defect image that contains both kinds of patches, we compute the mean errors of the normal and anomalous patches, r_N and r_A, and their gap $g = r_A - r_N$. This analysis compares MOFR with the $U(Z)$ feature-stream baseline of Table 2.

On the 1,212 analyzed images of MVTec-AD, the mean error of normal regions increases from 0.0156 to 0.0175 and that of anomalous regions from 0.0584 to 0.0717. The mean gap grows from 0.0428 to 0.0542. On the 729 images of VisA and the 28,697 images of Real-IAD, the mean gap grows from 0.0390 to 0.0498 and from 0.0349 to 0.0462. The fraction of images whose gap increases is 93.3%, 96.3%, and 98.7% on the three datasets.

This shows a tendency of the proposed architecture to increase the reconstruction error of anomalous regions more than that of normal regions. Since the mean gap describes neither the variance nor the ranking of the scores, an increase of the gap does not translate into an improvement of every detection and localization metric.

4.5 Component Analysis

Table 3: Component-wise comparison (%). Entries are I-AUROC / P-AUROC / AUPRO. The last column is the equal-weight mean of the three datasets for each metric. Best values in bold.

Configuration	MVTec-AD	VisA	Real-IAD	Mean
MOFR	**99.82** / 98.59 / **96.09**	**99.02** / **99.16** / 95.84	**91.12** / **99.22** / **96.27**	**96.65** / **98.99** / **96.07**
w/o global image condition	99.79 / 98.13 / 94.43	98.87 / 99.06 / 95.53	90.54 / 99.13 / 95.97	96.40 / 98.77 / 95.31
w/o CLS centering	99.78 / **98.62** / 95.96	98.60 / 98.97 / 94.96	89.83 / 98.97 / 94.65	96.07 / 98.85 / 95.19
Bottleneck width $d = 768$	99.79 / 98.56 / 95.81	98.90 / 99.10 / **95.93**	90.56 / 99.16 / 96.15	96.42 / 98.94 / 95.96
w/o bottleneck dropout	99.73 / 98.46 / 95.64	98.71 / 98.97 / 95.82	89.71 / 99.04 / 95.90	96.05 / 98.82 / 95.79
w/o FSQ rounding	99.81 / 98.56 / 95.98	98.96 / 99.14 / 95.79	90.90 / 99.19 / 96.15	96.56 / 98.96 / 95.97
w/o conditional output head	**99.82** / 98.57 / 96.04	98.97 / 99.14 / 95.77	90.99 / 99.20 / 96.20	96.59 / 98.97 / 96.00
w/o patch gradient modulation	99.81 / 98.54 / 96.00	98.83 / 99.09 / 95.77	90.54 / 99.16 / 96.12	96.39 / 98.93 / 95.96

Table 3 removes or changes one component of MOFR at a time. The effect of per-token conditioning was covered in Table 2, and here we compare the global condition, CLS centering, the bottleneck, the output head, and the gradient modulation during training.

Removing the global image condition lowers I-AUROC, P-AUROC, and AUPRO on all three datasets, with the largest drop in MVTec-AD AUPRO (1.66 pp). Removing CLS centering lowers I-AUROC and AUPRO on all three datasets, by 1.29 and 1.62 pp on Real-IAD, whereas MVTec-AD P-AUROC rises by 0.03 pp.

Widening the bottleneck to 768 or removing its dropout lowers I-AUROC on all three datasets, and only the wider bottleneck raises VisA AUPRO, by 0.09 pp. Removing FSQ rounding lowers I-AUROC by 0.01, 0.06, and 0.22 pp and AUPRO by 0.11, 0.05, and 0.12 pp, so its effect is small but consistent in direction.

Removing the conditional output head or the patch gradient modulation lowers all three metrics on all three datasets, except that MVTec-AD I-AUROC is unchanged without the output head, by at most 0.13 and 0.58 pp in Real-IAD I-AUROC, and the mean I-AUROC falls from 96.65% to 96.59% and 96.39%. The component analysis shows that each component contributes to image-level performance, while no single configuration is optimal for every localization metric, since MVTec-AD P-AUROC and VisA AUPRO are highest without CLS centering and with the wider bottleneck.

4.6 Model Size and Inference Throughput

Table 4: Model size (M parameters) and inference throughput (img/s). Throughput is measured on a single RTX 3090 with batch size 8 and BF16.

Method	Actual input resolution	Params (M) ↓	Throughput (img/s) ↑
RD4AD	256 × 256	150.6	421.5
UniAD	256 × 256	24.5	313.5
SimpleNet	256 × 256	72.8	166.6
DeSTSeg	256 × 256	35.2	366.8
MambaAD	256 × 256	25.7	119.5
DiAD	256 × 256	1331.3	3.9
Dinomaly	392 × 392	132.8	131.1
INP-Former	392 × 392	139.8	138.9
MOFR	392 × 392	133.1	148.4

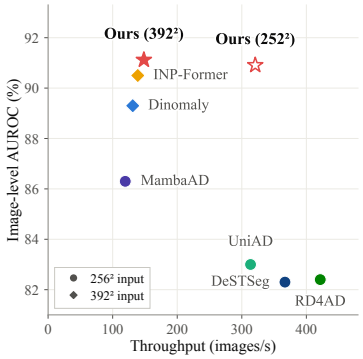

Figure 5: Accuracy versus throughput on Real-IAD. The I-AUROC of the compared methods is taken from the literature, and the throughput is measured by us. Circle: 256^2 input, diamond: 392^2, filled and open star: MOFR at 392^2 and 252^2.

Table 4 reports the model size and the throughput measured on a single RTX 3090 with batch size 8 and BF16. At the same 392 × 392 input size, MOFR processes 148.4 images per second with 133.1M parameters, about 13.2% and 6.8% more than Dinomaly (131.1 img/s) and INP-Former (138.9 img/s). In Fig. 5, MOFR lies above and to the right of both. At a 252 × 252 input, the throughput rises to 320.8 img/s, about 2.2 times the default, while the Real-IAD I-AUROC decreases by 0.2 pp and the pixel-level metrics decrease more. Detailed accuracy and throughput for each input resolution are given in Appendix A.

5 Conclusion

We proposed MOFR, which pairs an image-independent initial state with per-token feature conditioning. Instead of initializing the decoder state with the input features, MOFR reconstructs features by conditionally updating a shared positional embedding. The residual MLP decoder with per-token and global conditions maintained high anomaly detection performance on three industrial datasets without token-to-token attention.

The cross comparison of the initialization pathway and conditioning showed that the contribution of per-token conditioning depends strongly on whether the initial state contains input information. The differences from feature-stream reconstruction were small and varied with the dataset and the metric, and the possibility of reconstructing anomalous features through modulation remains. Our results support the feasibility and the performance of a reconstruction scheme that replaces input-derived initialization with conditional updates, rather than a universal superiority of the shared initial state.

AI USE STATEMENT

In the interest of transparency and in compliance with the ICLR 2027 guidelines, we report that a large language model (LLM) was used to assist in the preparation of this paper.

Scope of use. The model's role was limited to that of a writing and figure-preparation assistant. Its contributions include:

- Correcting grammatical errors, spelling, and punctuation.
- Improving sentence structure and flow for clarity.
- Refining word choices for precision and conciseness.
- Polishing sentences.
- Creating and editing figures.

REFERENCES

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

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 (CVPR)*, pp. 9592–9600, 2019.

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 (CVPR)*, pp. 4183–4192, 2020.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International Conference on Learning Representations (ICLR)*, 2024.

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 (CVPR)*, pp. 9737–9746, 2022.

Jia Guo, Shuai Lu, Lei Fan, Zelin Li, Donglin Di, Yang Song, Weihang Zhang, Wenbing Zhu, Hong Yan, Fang Chen, Huiqi Li, and Hongen Liao. One Dinomaly2 detect them all: A unified framework for full-spectrum unsupervised anomaly detection. *arXiv preprint arXiv:2510.17611*, 2025a.

Jia Guo, Shuai Lu, Weihang Zhang, Fang Chen, Huiqi Li, and Hongen Liao. Dinomaly: The less is more philosophy in multi-class unsupervised anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20405–20415, 2025b.

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. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 71162–71187, 2024a.

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, pp. 8472–8480, 2024b.

Liren He, Zhengkai Jiang, Jinlong Peng, Wenbing Zhu, Liang Liu, Qiangang Du, Xiaobin Hu, Mingmin Chi, Yabiao Wang, and Chengjie Wang. Learning unified reference representation for unsupervised multi-class anomaly detection. In *European Conference on Computer Vision (ECCV)*, pp. 216–232, 2024c.

Zhikang Liu, Yiming Zhou, Yuansheng Xu, and Zilei Wang. SimpleNet: A simple network for image anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20402–20411, 2023.

Wei Luo, Yunkang Cao, Haiming Yao, Xiaotian Zhang, Jianan Lou, Yuqi Cheng, Weiming Shen, and Wenyong Yu. Exploring intrinsic normal prototypes within a single image for universal anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9974–9983, 2025.

Fabian Mentzer, David Minnen, Eirikur Agustsson, and Michael Tschannen. Finite scalar quantization: VQ-VAE made simple. In *International Conference on Learning Representations (ICLR)*, 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research (TMLR)*, 2024.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205, 2023.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

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 (CVPR)*, pp. 14318–14328, 2022.

Chengjie Wang, Wenbing Zhu, Bin-Bin Gao, Zhenye Gan, Jiangning Zhang, Zhihao Gu, Shuguang Qian, Mingang Chen, and Lizhuang Ma. Real-IAD: A real-world multi-view dataset for benchmarking versatile industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22883–22892, 2024.

Mitchell Wortsman, Tim Dettmers, Luke Zettlemoyer, Ari Morcos, Ali Farhadi, and Ludwig Schmidt. Stable and low-precision training for large-scale vision-language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 10271–10298, 2023.

Zhiyuan You, Lei Cui, Yujun Shen, Kai Yang, Xin Lu, Yu Zheng, and Xinyi Le. A unified model for multi-class anomaly detection. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 4571–4584, 2022.

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 (CVPR)*, pp. 3914–3923, 2023.

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 (ECCV)*, pp. 392–408, 2022.