VETime: Vision Enhanced Zero-Shot Time Series Anomaly Detection

Yingyuan Yang[1] **Tian Lan**[1] **Yifei Gao**[1] **Yimeng Lu**[1] **Xuming An**[1]
Wenjun He[2] **Meng Wang**[2] **Chenghao Liu**[3†] **Chen Zhang**[1†]

[1]Department of Industrial Engineering, Tsinghua University, Beijing, China
[2]2012 Lab, Huawei Technologies Ltd, Shenzhen, China
[3]Datadog AI Research
yangyingyuan22@mails.ucas.ac.cn lant23@mails.tsinghua.edu.cn
gao-yf@mail.tsinghua.edu.cn {luym25,axm24}@mails.tsinghua.edu.cn
{hewenjun8, wangmeng71}@huawei.com
twinsken@gmail.com zhangchen01@tsinghua.edu.cn

Abstract

Time-series anomaly detection (TSAD) requires identifying both immediate point anomalies and long-range context anomalies. However, existing zero-shot foundation models face a fundamental trade-off: 1D temporal models provide fine-grained pointwise localization but lack a global contextual perspective, while 2D vision-based models capture global patterns but suffer from weak temporal alignment and coarse-grained localization. To resolve this dilemma, we propose VETime, the first zero-shot TSAD framework that unifies temporal and visual modalities through fine-grained visual-temporal alignment and dynamic fusion. VETime introduces a Traceable Image Conversion and a Patch-level Temporal Alignment module to establish a shared visual-temporal timeline, preserving discriminative details while maintaining temporal sensitivity. Furthermore, we design an Anomaly Window Contrastive Learning mechanism and a Task-adaptive Multi-modal Fusion to adaptively integrate the complementary perceptual strengths of both modalities. Extensive experiments demonstrate that VETime significantly outperforms state-of-the-art models in zero-shot scenarios, achieving superior localization precision with lower computational overhead than current vision-based approaches.

1 Introduction

Time-series anomaly detection (TSAD) is a fundamental yet challenging problem, which requires identifying rare deviations while providing precise temporal localization. In practice, domain heterogeneity and resource-constrained deployment render dataset-specific training impractical, driving the development of Time-Series Foundation Models (TSFMs) for zero-shot anomaly detection. In this setting, models [Shentu et al., 2024, Lan et al., 2025] are evaluated directly on unseen target domains without any target-specific training or fine-tuning.

A general zero-shot anomaly detector must handle at least two complementary anomaly patterns, as shown in Figure 1(a). **Point anomalies** appear as abrupt, short-lived numerical deviations [Shentu et al., 2024], requiring fine-grained local sensitivity. **Context anomalies** correspond to longer-range violations or structural context [He et al., 2026], requiring a macroscopic view of the sequence. Models that overfit to either local continuity or global structure may generalize poorly across unseen datasets where anomaly duration and morphology vary drastically.

†Corresponding authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

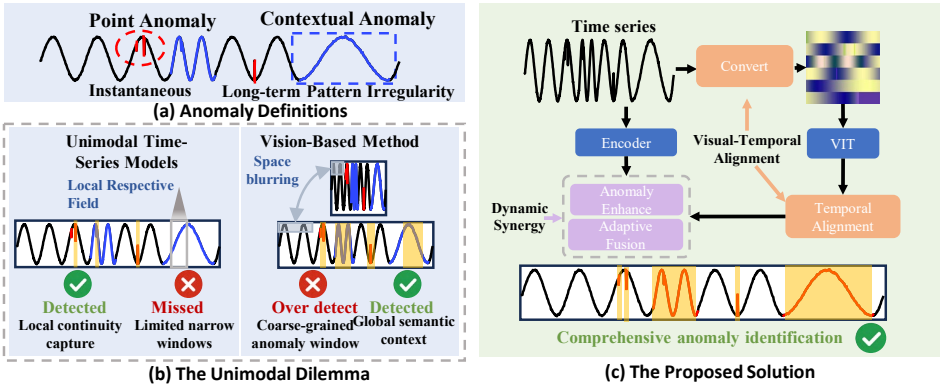


Figure 1: Comparison of the previous TSAD methods and the proposed VETime. While existing methods often rely on single-modality or coarse visual mapping, VETime introduces fine-grained temporal alignment and dynamic fusion to bridge the gap between visual and temporal perceptions.

Existing zero-shot TSAD methods mainly follow two directions. First, TSFMs [Ansari et al., 2024, Shi et al., 2024, Ekambaram et al., 2025] operate on 1D numerical sequences and preserve local temporal resolution, making them suitable for pointwise localization. However, their windowed numerical modeling often provides a local receptive field and lacks macroscopic context for long-range anomalies. Conversely, vision- or VLM-based methods [Zhou and Yu, 2024, Xu et al., 2025, He et al., 2026] render time series as images and rely on prompt-based reasoning, exploiting pretrained visual perception to capture global patterns. Furthermore, while existing methods [He et al., 2026, Zhou and Yu, 2024] demonstrate the utility of visual information for TSAD, they suffer from coarse windows and a lack of temporal alignment, leading to poor pointwise anomaly localization [Zhuang et al., 2024]. These single-perspective foundation models face a critical competency gap in effectively handling both anomaly types simultaneously, as shown in Figure 1(b).

A natural progression to resolve this dilemma is integrating temporal and visual modalities to leverage their complementary strengths. Although pioneers like Time-VLM [Zhong et al., 2025] and Aurora [Wu et al., 2026] explore multi-modal time-series modeling, they are tailored for forecasting. This causes critical task misalignment: forecasting prioritizes macro sequence trends over dense timestamp-level localization, while prediction-residual paradigms struggle to generalize across diverse anomaly patterns under zero-shot settings. Consequently, developing a native zero-shot multimodal TSAD framework requires tackling two key obstacles: **Visual-Temporal Ordered Alignment:** restoring explicit temporal indexing onto a shared timeline required for robust anomaly discrimination during 1D-to-2D visual conversion and multi-modal alignment; and **Dynamic Synergy for Diverse Anomaly Patterns:** adaptively fusing 1D local correlations and 2D macroscopic patterns to provide comprehensive context information for tackling complex unseen anomaly scenarios.

To bridge these gaps, we propose VETime, the first framework that unifies temporal and visual features via effective alignment and fusion, enabling robust anomaly detection and precise localization, as shown in Figure 1(c). To overcome the **alignment barrier**, we first develop a novel *Traceable Image Conversion* method which embeds the 1D sequence into a 2D topological manifold, preserving information-dense anomaly details without losing explicit temporal indices. Complementing this, we propose the *Patch-level Temporal Alignment* module, which enhances 2D visual representations from the pre-trained ViT with a 1D temporal ordering, establishing the structural basis for fine-grained cross-modal interaction. Subsequently, to facilitate **dynamic synergy**, we introduce two key mechanisms. Considering the distinct perceptual characteristics of visual and temporal modalities regarding anomalies, we propose a specialized *Anomaly Window Contrastive Learning*. This incorporates intra- and inter-window contrast between multi-modal features to achieve comprehensive anomaly identification. Finally, a *Task-adaptive Multi-modal Fusion* module equipped with a reconstruction head is applied to facilitate robust multi-modal fusion. This mechanism performs dynamic, adaptive weighted fusion of the aligned features, ensuring precise performance in downstream anomaly detection. Our main contributions are listed as follows:

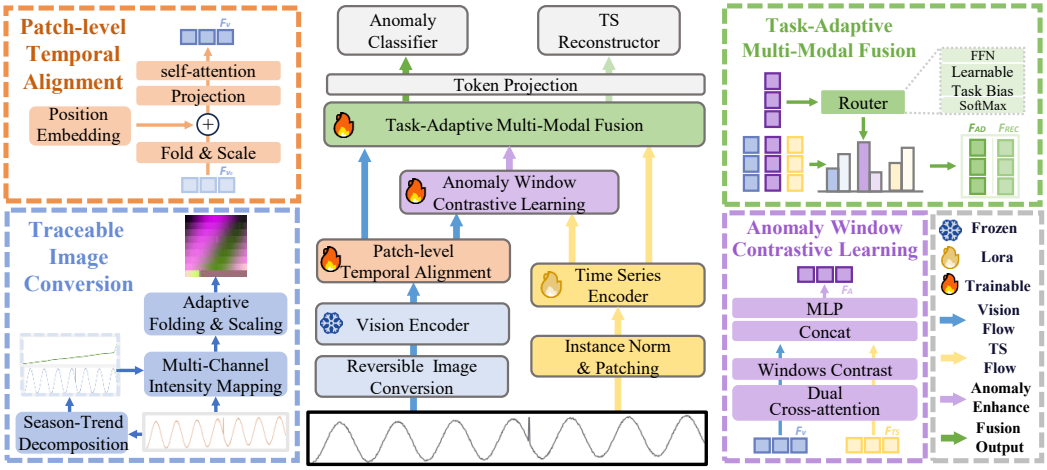


Figure 2: Overview of the proposed framework. The time series is first processed by a time-series encoder to extract temporal features F_{TS} while simultaneously undergoing Traceable Image Conversion to generate visual input. Then visual features F_{V_0} extracted from a frozen pre-trained image encoder are subsequently transformed into F_V through Patch-level Temporal Alignment to reinforce their temporal positional association. Then, F_{TS} and F_V are input into the Anomaly Window Contrastive Learning to derive an anomaly-enhanced representation F_A. Finally, a Task-adaptive Multi-modal Fusion module integrates all features (F_A, F_{TS} , F_V). Final outputs F_{AD} and F_{Rec} are mapped to the original sequence length via token projection for the respective anomaly classification and reconstruction heads.

- We propose VETime, the first zero-shot TSAD framework that integrates visual and temporal features, effectively exploiting the distinct perceptual advantages of each modality.
- We introduce the Traceable Image Conversion and Patch-level Temporal Alignment, which jointly capture informative visual contexts while preserving critical temporal sensitivities.
- We design Anomaly Window Contrastive Learning and Task-adaptive Multi-modal Fusion, harnessing multi-modal perceptual strengths across diverse anomaly types.
- Extensive experiments demonstrate that VETime significantly outperforms state-of-the-art zero-shot TSFMs. Compared to VLM-based baselines, VETime achieves superior precision with much lower computational overhead.

2 Related work

TSAD and time-series foundation models. Traditional TSAD relies on domain-specific reconstruction or prediction models [Hundman et al., 2018, Su et al., 2019, Audibert et al., 2020, Tuli et al., 2022, Xu et al., 2022, Yang et al., 2023]. Recently, time-series foundation models (TSFMs) have advanced cross-domain zero-shot generalization via large-scale pretraining [Goswami et al., 2024, Das et al., 2024, Ansari et al., 2024, Shi et al., 2024, Ekambaram et al., 2025, Gao et al., 2024]. Anomaly-oriented models such as DADA and TimeRCD [Shentu et al., 2024, Lan et al., 2025] further introduce anomaly-specific objectives or synthetic anomaly supervision for zero-shot TSAD. Despite preserving fine-grained local temporal precision, these 1D numerical approaches inherently lack the macroscopic perspective required for identifying long-range contextual anomalies. VETime addresses this gap by complementing zero-shot temporal modeling with traceable 2D visual context and visual-temporal fusion.

Time-series-to-image representations and vision-enhanced time-series modeling. Transforming 1D time series into 2D visual representations allows models to leverage mature Vision Transformers (ViTs) and capture multi-scale periodic patterns [He et al., 2026]. Traditional methods often rely on fixed spatial-temporal transforms like Gramian Angular Fields (GAF) or Markov Transition Fields (MTF) [Wang and Oates, 2015]. However, these transformations are non-invertible, destroying temporal phase details essential for point-wise localization. Recent vision-enhanced methods [Chen

et al., 2025, Wang et al., 2024, Dong et al., 2025] dilute the mapping between image patches and original timestamps, inevitably introducing severe information bottlenecks. In contrast, our work introduces an adaptive folding mechanism with exact spatial-temporal mapping, maintaining full reversibility and fine-grained temporal fidelity.

Vision- and VLM-based zero-shot TSAD. Recent studies [Zhuang et al., 2024, He et al., 2026, Xu et al., 2025] also leverage pretrained vision or multimodal large language models (MLLMs) for anomaly detection. VLM4TS [He et al., 2026] localizes candidate anomaly intervals via the ViT4TS module and applies a VLM to verify and refine them using global visual context. While validating the efficacy of visual perception for global patterns, their reliance on rendered line plots or prompt-based reasoning incurs high inference costs and weak timestamp-to-patch mapping. In contrast, VETime is an efficient detector that realigns visual patches to 1D temporal tokens, enabling dense pointwise anomaly scoring without computationally expensive VLM prompting.

3 Methodology

To harness the differentiated yet complementary advantages of temporal and visual modalities, we propose a unified framework capable of fine-grained temporal alignment and dynamic interaction for anomaly detection. As illustrated in Figure 2, the framework comprises four core components:

- **Traceable Image Conversion (TIC):** Transforms 1D time series into traceable visual representations, encapsulating global and local correlations into semantically rich visual formats.
- **Patch-level Temporal Alignment (PTA):** Realigns visual features to the temporal axis to ensure fine-grained semantic correspondence.
- **Anomaly Window Contrastive Learning (AWCL):** Leverages the complementary nature of visual and temporal modalities to enforce multi-scale discriminability.
- **Task-adaptive Multi-modal Fusion (TMF):** Dynamically integrates multimodal features using sequence reconstruction as an auxiliary constraint.

Ultimately, the fused representations capitalize on the synergy between temporal and visual domains, facilitating accurate localization across diverse anomaly types. It is noted that our framework supports univariate and multivariate time series settings. The main text focuses on univariate cases, with the multivariate extension detailed in Appendix A.

3.1 Traceable image conversion

To accentuate anomalous signals, we transform univariate time series into high-density visual representations via a three-stage pipeline as shown in Figure 3(a).

Multi-channel intensity mapping: Unlike prior single-channel mapping approaches [Chen et al., 2025], we employ a multi-channel encoding strategy, which capitalizes on the standard RGB architecture to incorporate denser information and better expose latent anomalous patterns. Following DLinear [Zeng et al., 2023], the raw series $X \in \mathbb{R}^L$ is decomposed into a trend component X_{trend} and a remainder component X_{rem}, where L is the length of the series. The triplet $\{X, X_{trend}, X_{rem}\}$ is independently Min-Max normalized to $[0, 255]$ and mapped to the R, G, and B channels, yielding a $1 \times L \times 3$ tensor that explicitly encompasses both global trends and high-frequency residuals.

Adaptive folding: To accommodate diverse temporal scales, we adopt a periodicity-based folding strategy [Wu et al., 2023] to transform a 1D sequence into a 2D grid with dimensions $(T_{fold}, \lceil L/T_{fold} \rceil, 3)$. The folding period T_{fold} is dynamically estimated via the autocorrelation function (defaulting to $\sqrt{L}$ for non-periodic samples), and adjusted to be a multiple of the ViT patch size to prevent temporal information discontinuity. For indivisible lengths, we use the mean of the final period to pad trailing segments, minimizing distribution shifts compared to zero-padding.

Dimension-aware scaling: To standardize input resolution from $(T_{fold}, \lceil L/T_{fold} \rceil, 3)$ to $(224, 224, 3)$ without fidelity loss, we apply a decoupled scaling strategy. We use linear interpolation

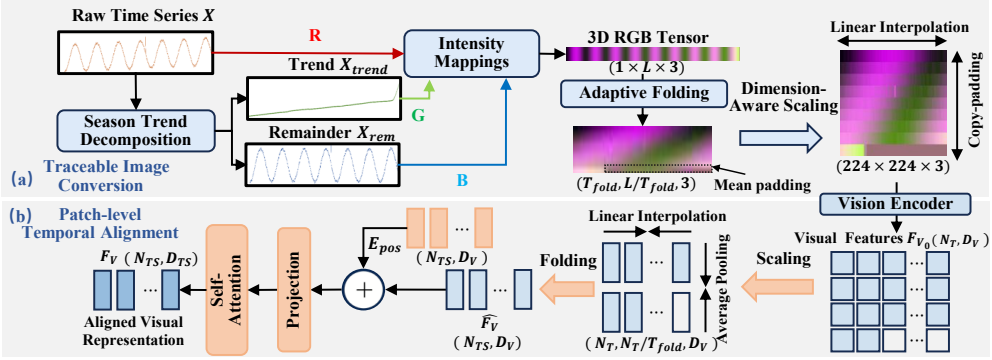


Figure 3: The framework of Traceable Image Conversion and Patch-level Temporal Alignment.

along the time axis (horizontal) to preserve waveform continuity, while employing a copy-padding strategy along the period axis (vertical) to prevent semantic distortion or pseudo-patterns that could arise from interpolation across distinct periods.

3.2 Patch-level temporal alignment

The generated visual representations are fed into a frozen visual encoder to extract initial visual features $F_{V_0} \in \mathbb{R}^{N_V \times D_V}$. In parallel, the time series undergoes instance normalization and patching, followed by a temporal encoder to yield $F_{TS} \in \mathbb{R}^{N_{TS} \times D_{TS}}$. Here, N and D denote the patch count and feature dimension for the respective modalities. To bridge the structural discrepancy between modalities, we introduce a Patch-level Temporal Alignment module for F_{V_0} (Figure 3(b)).

Specifically, visual features are mapped back to the 1D temporal domain by inverting the folding logic. F_{V_0} is reshaped to the initial 2D grid, linearly interpolated along the temporal axis to match the temporal patch count, and average-pooled along the periodicity axis to aggregate redundant repetitions. The resulting features are flattened to yield the concise aligned image features $\hat{F}_V \in \mathbb{R}^{N_{TS} \times D_V}$.

Finally, to recover temporal context lost during visual encoding, we add learnable positional encoding $E_{POS} \in \mathbb{R}^{N_{TS} \times D_V}$ followed by projection and self-attention layers to model intra- and inter-patch dependencies. Ultimately, visual representations F_V maintain temporal correspondence with F_{TS}.

3.3 Anomaly window contrastive learning

Time-series (F_{TS}) and visual F_V features are first input into a dual cross-attention. Here, features from each modality serve as queries to integrate complementary contexts, followed by a residual FFN for refinement, yielding updated representations Z_{TS} and Z_V (detailed in Appendix B.5). Given the inherent disparities between time-series and image features, specifically concerning their receptive fields and sensitivity to anomalies, we employ a hybrid strategy comprising Intra-Window and Inter-Window contrastive learning to model discriminative features across scales, as shown in Figure 4.

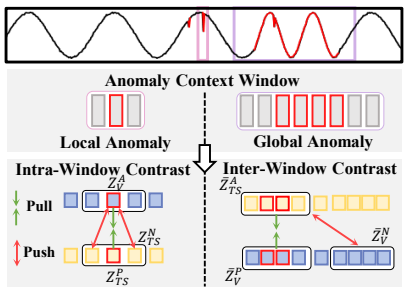


Figure 4: An illustration of Anomaly window contrastive learning

Anomaly context windows: During source-domain pre-training, we dynamically construct adaptive windows around anomalies. Point-level labels are converted to patch-level labels (set to 1 if any timestamp within the patch is anomalous). For a continuous anomaly segment spanning L_a patches, we define an Anomaly Context Window by symmetrically extending the anomaly segment bidirectionally into the normal context, with a maximum length of $L_w = 2L_a$. Each window contains one continuous anomaly instance surrounded by its local context.

Intra-window contrastive learning: Targeting short-duration point anomalies ($L_a \leq 1$ patches, e.g., outliers), this component enforces fine-grained anomaly alignment for the visual feature Z_V. Within a given window, the anomalous visual feature serves as the anchor Z_V^A, paired with the corresponding temporal feature as the positive sample Z_{TS}^A. Normal temporal features within the same window act as negatives Z_{TS}^N. The intra-window contrastive loss $\mathcal{L}_{intra}$ is defined as:

$$\mathcal{L}_{intra} = -\log \frac{\exp(Z_V^A \cdot Z_{TS}^A/\tau)}{\exp(Z_V^A \cdot Z_{TS}^A/\tau) + \sum_{k\in\mathcal{N}_{neg}} \exp(Z_V^A \cdot Z_{TS}^k/\tau)}, \tag{1}$$

where $\cdot$ denotes dot product similarity, τ is the temperature parameter, and $\mathcal{N}_{neg}$ represents the set of normal features within the window.

Inter-window contrastive learning: For long-duration context anomalies ($L_a \geq 2$ patches, e.g., seasonal anomaly), we enhance global discriminability for temporal feature Z_{TS}. Features are average-pooled along the temporal dimension within each window. The pooled time-series feature of an anomaly window $\bar{Z}_{TS}^A$ serves as the anchor, paired with its visual counterpart $\bar{Z}_V^A$, while pooled features from normal windows $\bar{Z}_V^N$ serve as negatives. The inter-window contrastive loss $\mathcal{L}_{inter}$ is:

$$\mathcal{L}_{inter} = -\log \frac{\exp(\bar{Z}_{TS}^A \cdot \bar{Z}_V^A/\tau)}{\exp(\bar{Z}_{TS}^A \cdot \bar{Z}_V^A/\tau) + \sum_{j\in\mathcal{W}_{neg}} \exp(\bar{Z}_{TS}^A \cdot \bar{Z}_V^j/\tau)}, \tag{2}$$

where $\mathcal{W}_{neg}$ denotes the set of normal window features. The total contrastive loss $\mathcal{L}_{aw}$ is computed independently for each sample to prevent interference from the diverse normal and anomalous patterns across instances:

$$\mathcal{L}_{aw} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{W_s}\sum_{s=1}^{W_s}\mathcal{L}_{intra}^{(i,s)} + \frac{1}{W_l}\sum_{l=1}^{W_l}\mathcal{L}_{inter}^{(i,l)}\right), \tag{3}$$

where N is the number of samples, W_s and W_l are the numbers of short and long anomalies in each instance. The optimized representations are concatenated and refined via an FFN to produce the anomaly-enhanced representation $F_A = \text{FFN}([Z_{TS}; Z_V])$, where $[\cdot;\cdot]$ denotes feature concatenation.

3.4 Task-adaptive multi-modal fusion

To realize dynamic semantic integration and efficient end-to-end training, we propose a Task-adaptive Multi-modal Fusion Module. This module acts as a dynamic routing mechanism, treating time-series (F_{TS}), vision (F_V), and anomaly-enhanced (F_A) features as experts in a candidate space.

Dynamic weight routing: To assign importance to the three experts $\mathcal{M} = \{TS, V, A\}$ for anomaly detection (AD) and reconstruction (Rec), the routing network generates patch-level weights $W^t \in \mathbb{R}^{N_{TS}\times\mathcal{M}}$ from the anomaly-enhanced feature $F_A \in \mathbb{R}^{N_{TS}\times D}$:

$$W^t = \text{Softmax}(\text{MLP}(F_A) + \mathbf{b}^t), \tag{4}$$

where $\mathbf{b}^t \in \mathbb{R}^{\mathcal{M}}$ is a learnable task-specific bias, and the Softmax function normalizes the probabilities across the expert dimension.

To prevent premature routing collapse and encourage diverse expert utilization, we introduce an auxiliary load-balancing loss. We compute the temporally averaged expert utilization $\hat{w}_m^t = \frac{1}{N_{TS}}\sum_{i=1}^{N_{TS}} w_{i,m}^t$, where $w_{i,m}^t$ is the assigned probability of expert m at patch i for task t, and minimize its negative entropy to maximize global routing diversity: $\mathcal{L}_e = \sum_{t\in\{AD,Rec\}}\sum_{m\in\mathcal{M}} \hat{w}_m^t \log \hat{w}_m^t$. This balances global modality allocation while permitting local dynamic gating based on contextual demands.

Task-adaptive fusion: Based on the routing weights, the fused feature F_t for each task is computed as a weighted sum of the experts $F_m \in \mathbb{R}^{N_{TS}\times D}$:

$$F_t = \sum_{m\in\mathcal{M}} W_m^t \odot F_m, \quad t \in \{AD, Rec\}, \tag{5}$$

where $W_m^t \in \mathbb{R}^{N_{TS}\times 1}$ is the weight vector for expert m, and $\odot$ denotes element-wise multiplication broadcasted along the feature dimension.

The resulting F_{AD} and F_{Rec} are subsequently fed into their respective heads. Crucially, sequence reconstruction serves as an auxiliary task; by forcing the network to reconstruct the raw input, it fosters deep cross-modal interactions and mitigates overfitting to sparse anomaly labels.

3.5 Optimization

The features F_{AD} and F_{Rec} are projected to the original sequence length and fed into separate specific heads. The model is pre-trained on source domains via a multi-task objective:

$$\mathcal{L}_{total} = \mathcal{L}_{BCE} + \mathcal{L}_{MSE} + \lambda_{aw}\mathcal{L}_{aw} + \lambda_e\mathcal{L}_e \tag{6}$$

where $\mathcal{L}_{BCE}$ and $\mathcal{L}_{MSE}$ target anomaly classification and sequence reconstruction; the detailed computation is provided in Appendix B.5. λ denotes the weights balancing the different losses.

4 Experiments

Datasets and metrics. Our evaluation is conducted on a comprehensive suite of 26 public datasets from the TSB-AD benchmark [Liu and Paparrizos, 2024], comprising 11 univariate (excluding versions derived from multivariate sources to prevent evaluation bias [Wu and Keogh, 2022]) and 15 multivariate datasets. We evaluate performance using four standard metrics: Affiliation-F1, F1-T, Standard-F1, and VUS-PR. Comprehensive details regarding dataset statistics, formal metric definitions, and baseline configurations are provided in Appendix B.

4.1 Comparative experiment

To guarantee a strict zero-shot evaluation and prevent data leakage, VETime is trained on a 0.5-billion-point synthetic corpus [Lan et al., 2025] generated from mathematical equations, without

Table 1: Performance of VETime against zero-shot baselines on 26 public datasets (11 univariate and 15 multivariate). Asterisked (*) results are excluded from ranking due to data leakage of models marked with (†). Red: best, Blue: second best.

Metric	Model	Univariate Datasets											Total 1st	Total 2nd	Avg Rank
		IOPS	MGAB	NAB	NEK	Power	SED	Stock	TODS	UCR	WSD	YAHOO			
Affiliation-F1	VETime	**90.53**	**68.03**	**88.57**	79.56	**78.78**	**97.31**	69.60	85.85	**85.06**	**94.31**	**97.15**	**04**	**04**	**2.55**
	TimeRCD	83.28	**70.69**	82.48	79.73	**85.51**	**96.87**	71.84	86.37	**84.63**	90.33	**96.65**	**02**	**03**	**3.27**
	DADA †	89.37*	67.66*	86.56	**95.40**	69.79	65.18	**98.77**	76.89	72.21	93.92	92.20*	**02**	00	4.00
	TS-Pulse	68.76	67.33	70.80	73.05	69.94	67.44	67.93	67.90	67.70	68.22	70.05	00	00	6.64
	MOMENT†	87.54*	66.76*	**90.45***	92.26	75.97	59.13	45.26	59.76	75.77	**95.39**	79.99*	**02**	00	4.73
	TimesFM	81.88	66.95	79.73	90.49	69.88	67.14	**97.53**	**89.08**	70.03	78.97	91.28	00	02	5.27
	Chronos	**90.12**	67.89	86.66	**93.63**	69.72	67.89	96.85	**91.96**	74.35	90.98	96.34	01	02	**3.27**
	Time MOE	76.34	67.23	80.51	80.50	71.19	60.98	63.28	54.68	73.56	80.25	69.70	00	00	6.27
F1_T	VETime	**46.15**	**2.00**	**44.31**	**60.86**	**20.16**	**74.35**	22.41	**68.33**	**36.25**	**50.42**	**91.54**	**06**	**04**	**1.91**
	TimeRCD	28.44	**1.81**	**38.85**	35.87	**28.47**	**69.43**	31.73	**65.89**	34.30	35.04	**85.86**	**01**	**05**	**3.27**
	DADA †	42.50*	00.91*	37.24	47.98	19.80	09.56	**95.49**	35.18	07.22	**48.46**	79.52*	**01**	01	4.45
	TS-Pulse	04.10	00.81	34.61	27.07	19.90	09.71	15.98	13.45	05.12	4.57	05.50	00	00	7.00
	MOMENT †	33.15*	00.80*	52.27*	**63.66**	19.91	09.54	18.04	17.47	13.02	41.98	11.69*	**01**	00	5.00
	TimesFM	**48.95**	00.93	36.74	36.63	19.80	09.58	88.94	51.13	**41.38**	10.78	83.46	**02**	00	4.18
	Chronos	45.45	01.10	36.10	33.16	19.90	13.18	**89.30**	53.90	10.88	39.82	79.00	00	**01**	4.27
	Time MOE	25.95	00.63	38.70	15.78	19.85	17.73	34.13	20.91	08.29	22.6	37.11	00	00	5.91
Standard-F1	VETime	**34.56**	**1.89**	**37.87**	**53.95**	**20.12**	**75.02**	23.83	**69.13**	**30.61**	**46.55**	**88.93**	**08**	**02**	**1.73**
	TimeRCD	24.22	**01.62**	**27.70**	33.05	**28.59**	**69.88**	32.61	**67.02**	**28.13**	31.96	**87.02**	**01**	**06**	**3.27**
	DADA †	32.76*	00.80*	26.91	48.24	15.99	02.69	**95.59**	28.18	03.36	**45.06**	79.30*	**01**	01	4.36
	TS-Pulse	03.54	00.73	21.61	23.96	18.27	08.84	15.46	12.45	02.05	2.17	04.00	00	00	6.82
	MOMENT †	30.69*	00.67*	44.75*	**63.85**	16.39	03.36	19.38	14.64	09.00	41.42	10.54*	**01**	00	4.73
	TimesFM	**34.28**	00.83	26.46	38.15	16.73	02.96	89.13	40.08	07.86	38.5	84.44	00	**01**	4.27
	Chronos	32.69	00.99	26.22	33.54	17.47	08.74	**89.41**	40.52	08.21	34.58	78.89	00	**01**	4.55
	Time MOE	26.52	00.45	26.20	11.47	12.16	17.73	34.32	16.38	04.09	20.09	27.50	00	00	6.55
VUS-PR	VETime	**30.79**	**0.70**	**32.98**	**57.54**	**11.78**	**89.05**	76.08	**94.20**	**24.99**	**33.85**	**88.61**	**06**	**04**	**2.00**
	TimeRCD	**20.23**	**01.05**	24.32	27.88	**21.25**	**80.75**	77.28	**93.46**	**23.09**	21.77	84.41	**02**	**04**	**3.09**
	DADA †	24.97*	00.57*	**24.73**	46.85	10.61	06.42	**99.51**	64.83	02.94	33.42	70.74*	01	01	4.00
	TS-Pulse	04.64	00.56	16.40	19.39	11.72	09.11	70.95	45.86	01.20	1.83	09.93	00	00	6.82
	MOMENT †	37.35*	00.56*	45.38*	**67.74**	10.50	04.31	76.97	56.45	06.17	**55.26**	30.81*	**02**	00	3.73
	TimesFM	19.56	00.58	24.01	35.02	10.44	06.13	**98.39**	72.89	06.03	21.57	**86.78**	00	**02**	4.45
	Chronos	19.00	00.60	23.76	31.80	10.95	08.65	97.49	70.66	06.56	18.81	83.54	00	00	4.91
	Time MOE	16.63	00.52	22.62	19.76	09.34	10.87	74.78	48.78	02.10	10.93	20.90	00	00	6.82
VETime Grand Total (Univariate)													**24**	**14**	**2.05**

Metric	Model	Multivariate Datasets															Total 1st	Total 2nd	Avg Rank
		MSL	PSM	SMAP	SMD	SWaT	GHL	Daph.	Exath.	Gen.	GECCO	MIT	SVDB	LTDB	CATS	TAO			
Affiliation-F1	VETime	**87.02**	77.27	**87.06**	**88.94**	**75.45**	**74.93**	**84.61**	**99.90**	**75.41**	**87.63**	**75.16**	**73.03**	**78.28**	72.08	**75.41**	05	08	**1.80**
	TimeRCD	81.16	**81.61**	**87.73**	**92.58**	71.55	**74.96**	81.90	68.02	73.19	70.18	**75.18**	70.05	75.28	68.02	73.74	05	00	2.60
	DADA †	76.57	**81.27**	76.92	83.74	**76.18**	71.84	**92.40**	93.72	**87.84**	**79.40**	71.73	**72.20**	**76.99**	68.59	**99.66**	04	04	**2.40**
	MOMENT †	74.55	65.79	77.42	74.00	70.17	40.90	67.74	**93.89**	68.63	18.28	70.95	71.96	64.86	**74.32**	20.31	01	01	3.73
	LOF	**84.35**	61.98	63.32	64.13	56.34	61.00	64.00	61.00	51.00	56.00	66.00	53.00	40.00	**73.00**	10.00	00	02	4.47
F1_T	VETime	**43.52**	**39.06**	**45.52**	**53.36**	**39.94**	**24.12**	**55.42**	**97.45**	3.06	20.25	**17.80**	**16.17**	**32.10**	**21.12**	28.32	10	02	**1.63**
	TimeRCD	**42.47**	**37.98**	**33.74**	**53.91**	30.28	**24.10**	**50.36**	28.17	3.06	4.90	14.17	13.29	**29.91**	12.50	19.86	01	05	**3.03**
	DADA †	34.58	31.84	30.42	40.80	**35.13**	7.24	48.81	74.65	**20.19**	18.06	9.14	14.16	29.85	7.92	**99.04**	02	01	3.07
	MOMENT †	25.97	27.77	17.93	28.68	28.76	3.92	22.74	**96.00**	6.19	**41.59**	**15.74**	**15.83**	30.77	**24.06**	**37.58**	02	05	3.07
	LOF	38.97	25.58	21.81	10.13	30.62	3.00	20.00	19.00	**13.00**	**26.00**	2.00	5.00	22.00	10.00	19.00	00	02	4.20
Standard-F1	VETime	**35.24**	**27.05**	**32.16**	**51.41**	**55.87**	**11.72**	**41.06**	**97.43**	2.98	**29.31**	**16.74**	**15.63**	**32.02**	**19.88**	**30.43**	10	04	**1.47**
	TimeRCD	**30.66**	**26.00**	**30.48**	**44.89**	28.73	**11.78**	**41.06**	27.56	2.98	10.00	**8.19**	12.82	**29.57**	11.59	21.57	01	06	2.73
	DADA †	22.13	24.07	26.75	34.94	**34.78**	4.68	**43.67**	**73.11**	**27.12**	21.95	7.13	**13.81**	27.69	7.18	**99.03**	03	03	**2.60**
	MOMENT †	14.43	23.83	12.92	29.78	21.30	0.00	1.77	27.65	**4.19**	**35.48**	7.25	8.86	6.23	**19.76**	13.84	01	02	3.63
	LOF	30.65	18.80	18.70	8.41	29.08	0.00	1.00	1.00	2.00	19.00	0.00	0.00	0.00	7.00	9.00	00	00	4.57
VUS-PR	VETime	**30.51**	**23.86**	23.70	**48.21**	42.93	**4.82**	**35.09**	**99.77**	**3.08**	15.37	**14.19**	**11.87**	**27.11**	12.00	**92.15**	09	06	**1.40**
	TimeRCD	20.45	**18.69**	**22.68**	**37.03**	17.58	**4.85**	35.01	9.61	3.07	2.64	5.53	8.61	**23.75**	5.56	86.53	01	04	**2.93**
	DADA †	12.74	17.17	20.02	25.98	**21.13**	1.75	**35.69**	73.94	2.84	9.81	3.79	7.94	19.91	3.86	**99.42**	02	01	3.20
	MOMENT †	09.32	16.48	08.97	15.96	14.90	1.31	14.27	**96.19**	1.63	**17.43**	**6.12**	**10.60**	22.28	**16.97**	26.04	02	03	3.33
	LOF	**24.67**	13.58	10.59	4.40	14.50	1.00	11.00	16.00	**6.00**	10.00	3.00	5.00	16.00	6.00	15.00	01	01	4.13
VETime Grand Total (Multivariate)																	**34**	**20**	**1.58**

using target-domain data or statistics. Model selection, early stopping, and hyperparameter tuning are performed exclusively on a held-out synthetic validation set. It is then evaluated directly on unseen target domains without any target-specific fine-tuning (detailed in Appendix B.3). Further evaluations against full-shot models, extended analyses regarding scalability, and diverse anomaly types are detailed in Appendix C.

Comparison with time-series models: As shown in Table 1, VETime establishes a new state-of-the-art in zero-shot TSAD, demonstrating superior generalization across both univariate and multivariate settings. On univariate datasets, VETime secures 24 out of 44 first-place rankings, consistently achieving the lowest average rank of 2.05. On multivariate datasets, it dominates by securing the top rank in 34 comparison instances. Notably, VETime achieves the best overall rank and the largest number of first-place results in critical metrics like VUS-PR, highlighting its stability independent of threshold selection. Remarkably, it even outperforms full-shot models (detailed in Appendix C.1), validating its ability to capture universal discriminative anomaly patterns without domain-specific training.

Table 2: Performance of VETime against vision-based baselines (SMAP and MSL are univariate settings following VLM4TS[He et al., 2026]).

Metric	Method	NAB	YAHOO	SMAP	MSL
Affiliation -F1	VETime	**88.57**	**97.15**	**98.83**	**92.78**
	VIT4TS	61.12	60.66	91.55	65.80
	VLM4TS	**70.37**	**64.25**	**94.76**	66.15
	VisualTimeAnomaly	62.46	61.03	91.65	**86.20**
	AnomLLM	58.53	36.43	42.52	46.08
F1_T	VETime	**44.31**	**91.54**	**70.21**	**50.38**
	VIT4TS	28.29	7.61	50.97	23.48
	VLM4TS	30.75	**9.10**	**57.13**	**37.50**
	VisualTimeAnomaly	**41.64**	3.02	14.80	13.84
	AnomLLM	17.76	1.67	4.17	3.32
Standard -F1	VETime	**37.87**	**88.93**	**64.18**	**52.18**
	VIT4TS	33.81	7.74	51.65	27.05
	VLM4TS	**34.78**	**9.16**	**57.56**	**42.21**
	VisualTimeAnomaly	31.72	3.63	16.19	16.96
	AnomLLM	23.70	2.36	6.65	10.70
VUS-PR	VETime	**32.98**	**88.61**	**60.67**	**42.81**
	VIT4TS	25.78	31.63	46.44	26.52
	VLM4TS	**27.89**	**33.09**	**52.28**	**33.57**
	VisualTimeAnomaly	26.87	13.13	11.94	17.91
	AnomLLM	19.50	10.44	4.92	9.77
Per-series runtime(s)	VETime	**0.04**	**0.02**	**0.04**	**0.02**
	VIT4TS	**1.72**	5.86	**2.23**	**2.75**
	VLM4TS	6.78	9.27	7.00	7.39
	VisualTimeAnomaly	2.52	**2.35**	2.34	2.85
	AnomLLM	3.59	3.91	3.40	3.95

Comparison with vision-based models: Due to the high computational cost and token consumption of vision-based models, we evaluate VETime against them on four univariate public datasets, following the protocol of VLM4TS [He et al., 2026]. As presented in Table 2, VETime demonstrates a decisive advantage in both detection accuracy and computational efficiency. It consistently achieves superior performance, surpassing competing methods across diverse datasets. Furthermore, it achieves an inference speed approximately 100 times faster than its vision-based counterparts, establishing it as a highly practical framework for real-time TSAD.

Comparison under different anomaly types. To demonstrate the effectiveness of our method on both local and contextual anomalies, we categorized and reorganized the NAB and YAHOO datasets into Point, Context, and Mixed anomaly types. Our approach is compared with VLM4TS and TimeRCD, and the results are presented in Table 3. By synergizing the fine-grained sensitivity of temporal features with the global semantic context of visual representations, VETime effectively mitigates the limitations of unimodal baselines and achieves consistent superiority across all anomaly categories.

Table 3: Performance comparison across anomaly types.

Metric	Model	Point	Context	Mixed anomaly
Affiliation-F1	VETime	**93.20**	**91.79**	**87.78**
	TimeRCD	**74.84**	**73.84**	**73.03**
	VLM4TS	67.11	61.94	60.79
F1_T	VETime	**66.17**	**63.32**	**52.61**
	TimeRCD	**31.51**	**54.06**	**42.12**
	VLM4TS	21.60	35.90	29.63
Standard-F1	VETime	**62.33**	**59.74**	**44.28**
	TimeRCD	**28.34**	32.18	26.20
	VLM4TS	23.15	**35.90**	**34.52**
VUS-PR	VETime	**59.33**	**57.59**	**44.61**
	TimeRCD	**61.85**	**44.93**	**42.90**
	VLM4TS	30.57	32.10	25.61

Table 4: Ablation study on NAB, WSD, and YAHOO (Affiliation-F1 / VUS-PR (%)). Base: TS Encoder + Classifier.

Variant	TIC	PTA	AWCL	TMF	NAB	WSD	YAHOO
Base					82.48 / 24.32	90.33 / 21.77	96.65 / 84.41
V1	✓				83.31 / 26.99	89.53 / 22.76	94.97 / 80.33
V2	✓	✓			82.93 / 27.75	90.56 / 24.70	95.19 / 83.82
V3	✓	✓	✓		83.99 / 29.52	90.11 / 26.30	96.86 / 85.31
V4	✓	✓		✓	**86.84** / **31.23**	**91.63** / **29.00**	**97.11** / **87.26**
V5	✓			✓	83.23 / 24.81	90.54 / 24.91	96.06 / 82.62
V6		✓		✓	82.89 / 25.64	88.63 / 22.32	94.84 / 81.95
V7			✓	✓	81.84 / 24.86	88.34 / 20.20	94.12 / 77.68
Full-Model	✓	✓	✓	✓	**88.57** / **32.98**	**94.31** / **33.85**	**97.15** / **88.61**

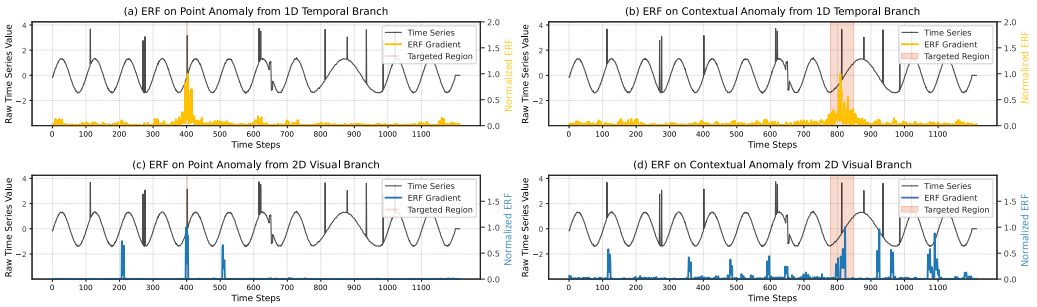


Figure 5: Effective Receptive Field (ERF) analysis comparing 1D temporal and 2D visual branches across different anomaly types. ERFs are computed via tracking the gradient flow from the final anomaly output back to the raw 1D or 2D inputs, with (b) and (d) averaged across the entire targeted region. 1D temporal branch (a, b) is strictly confined to local contexts, whereas 2D visual branch (c, d) exhibits period-aligned attention peaks that capture cross-period global awareness.

4.2 Ablation study

Table 4 ablates individual VETime components (see Appendix D for extended analyses on architectures, mapping strategies, and training objectives), revealing two key design imperatives:

Visual-temporal alignment (TIC + PTA). Effective multimodal modeling requires explicit feature alignment rather than naive visual conversion. TIC generates discriminative visual anomaly representations, while PTA remaps these features onto a shared 1D timeline. Removing this foundation (V6 and V7) causes multimodal interaction to fail, often degrading performance below the unimodal baseline.

Dynamic aggregation (AWCL + TMF). Merely generating and aligning visual representations (V1 and V2) yields marginal gains. Substantial improvements emerge when AWCL enforces multi-scale discriminability, successfully isolating short-term (local) and long-term (global) anomalies. Concurrently, TMF's task-adaptive routing dynamically allocates modality importance, significantly outperforming simple concatenation. This synergy (V3 and V4) enables the full model to resolve complex anomalies across challenging datasets like NAB and WSD.

4.3 Model analysis

This section explores the model's internal mechanisms through interpretability and sensitivity evaluations. Complementary analyses covering feature separability, computational overhead, and expanded visualizations are detailed in Appendices C and E.

Effective receptive field (ERF) analysis: Figure 5 visualizes the ERF of both branches, demonstrating that the visual branch attends to cross-periodic features, thereby overcoming the intrinsic locality of 1D receptive fields. For a marked region, the temporal branch is confined to adjacent segments, while the visual branch attends to long-range features across periods. This stems from aggregating multi-component interactions (via RGB mapping) and structurally converting long-range temporal distances into spatial adjacencies (via folding). This restructuring enhances information routing efficiency and yields semantically dense tokens, providing a global perspective absent in pure 1D models. Further structural analysis is detailed in Appendix C.4.

Hyperparameter sensitivity: We evaluate the impact of λ_{aw}, λ_e, and τ as shown in Figure 6. For λ_{aw} and τ in contrastive learning, deviations from empirical optima degrade performance, as excessive values over-smooth feature distributions. Additionally, λ_e is crucial for preventing expert collapse; insufficient regularization (< 0.2) impairs modality balancing and weakens discriminability.

Visualization of fusion weights. Figure 7 reveals a pronounced task divergence in routing weights: Anomaly-Enhanced features dominate detection to maximize discriminability, while Temporal

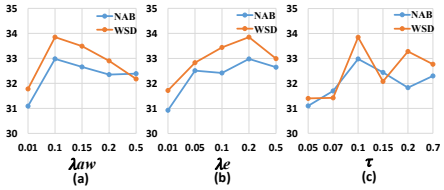


Figure 6: Hyperparameter analysis of λ_{aw}, λ_e, and τ under VUS-PR (%).

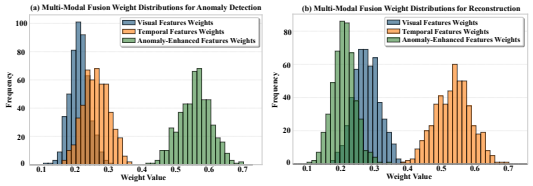


Figure 7: Visualization of fusion weights on the univariate datasets.

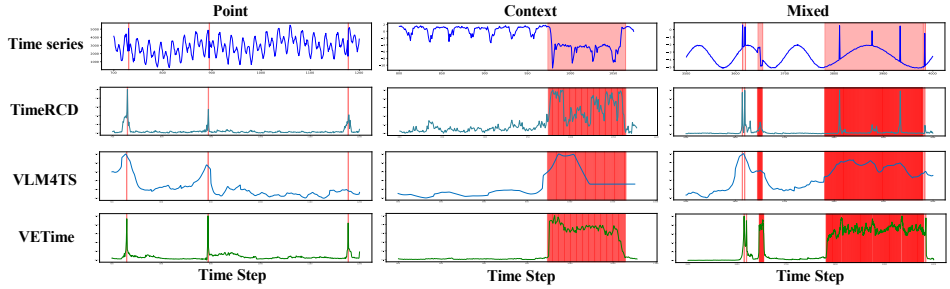


Figure 8: Qualitative anomaly detection results. The top row displays the time series with ground-truth anomalies (red shading), followed by anomaly scores from TimeRCD, VLM4TS, and VETime.

features govern reconstruction to retain numerical continuity. This strategy mitigates expert collapse by effectively separating high-level semantics from low-level precision.

Qualitative results: Figure 8 visualizes anomaly scores across Point, Contextual, and Mixed scenarios. VETime consistently outperforms unimodal baselines: it maintains precise point anomaly localization while leveraging global visual semantics to identify long-duration contextual deviations. In complex mixed scenarios, VETime significantly suppresses background noise compared to TimeRCD and rectifies VLM4TS's missed detections, demonstrating robust generalization.

5 Conclusion and limitations

We propose VETime, which is the first framework to fuse visual and temporal modalities for zero-shot time-series anomaly detection (TSAD). To bridge the gap between heterogeneous modalities, we introduce a Traceable Image Conversion method combined with Patch-level Temporal Alignment. Furthermore, by leveraging Anomaly Window Contrastive Learning and Task-adaptive Multi-modal Fusion, the model addresses the limitations of unimodal approaches in capturing diverse anomaly patterns. Extensive experiments validate its state-of-the-art zero-shot performance. Future work will incorporate textual modalities to provide semantic explanations for detected anomalies.

Limitations. (1) Sensitivity to high-frequency noise: Reliance on seasonal-trend decomposition makes VETime susceptible to irregular fluctuations where true structure and noise are entangled. **(2) High-dimensional constraints:** The channel-independent multivariate extension may miss complex inter-variable dependencies, and mapping ultra-high-dimensional data to a fixed visual canvas risks obscuring fine-grained signatures. **(3) Overhead and interpretability:** Incorporating a vision backbone inevitably increases computational complexity. Moreover, while our empirical evaluations are exhaustive and align with prior works, establishing rigorous theoretical justifications for multi-modal time-series representations remains an inherent, field-wide challenge.

Acknowledgements

Funding Statement This work was supported by the NSFC Grant 72271138, Tsinghua University Initiative Scientific Research Program 20243080039, and Tsinghua-Huawei Collaboration Project 20252001894. This work was partially supported by the computing resources from the Tsinghua High-Performance Computing Center and the KaiTuo 1000 supercomputer system.

Competing Interests The authors declare no competing interests relevant to this work during the 36 months preceding submission.

References

Qichao Shentu, Beibu Li, Kai Zhao, Yang Shu, Zhongwen Rao, Lujia Pan, Bin Yang, and Chenjuan Guo. Towards a general time series anomaly detector with adaptive bottlenecks and dual adversarial decoders. *arXiv preprint arXiv:2405.15273*, 2024.

Tian Lan, Hao Duong Le, Jinbo Li, Wenjun He, Meng Wang, Chenghao Liu, and Chen Zhang. Towards foundation models for zero-shot time series anomaly detection: Leveraging synthetic data and relative context discrepancy, 2025.

Zelin He, Sarah Alnegheimish, and Matthew Reimherr. Harnessing vision-language models for time series anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 21690–21698, 2026. doi: 10.1609/aaai.v40i26.39319.

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

Xiaoming Shi, Shiyu Wang, Yuqi Nie, Dianqi Li, Zhou Ye, Qingsong Wen, and Ming Jin. Time-moe: Billion-scale time series foundation models with mixture of experts. *arXiv preprint arXiv:2409.16040*, 2024.

Vijay Ekambaram, Subodh Kumar, Arindam Jati, Sumanta Mukherjee, Tomoya Sakai, Pankaj Dayama, Wesley M Gifford, and Jayant Kalagnanam. Tspulse: Dual space tiny pre-trained models for rapid time-series analysis. *arXiv preprint arXiv:2505.13033*, 2025.

Zihao Zhou and Rose Yu. Can LLMs understand time series anomalies? *arXiv preprint arXiv:2410.05440*, 2024.

Xiongxiao Xu, Haoran Wang, Yueqing Liang, Philip S. Yu, Yue Zhao, and Kai Shu. Can multimodal LLMs perform time series anomaly detection? *arXiv preprint arXiv:2502.17812*, 2025.

Jiaxin Zhuang, Leon Yan, Zhenwei Zhang, Ruiqi Wang, Jiawei Zhang, and Yuantao Gu. See it, think it, sorted: Large multimodal models are few-shot time series anomaly analyzers. *arXiv preprint arXiv:2411.02465*, 2024.

Siru Zhong, Weilin Ruan, Ming Jin, Huan Li, Qingsong Wen, and Yuxuan Liang. Time-VLM: Exploring multimodal vision-language models for augmented time series forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 78478–78497. PMLR, 2025.

Xingjian Wu, Jianxin Jin, Wanghui Qiu, Peng Chen, Yang Shu, Bin Yang, and Chenjuan Guo. Aurora: Towards universal generative multimodal time series forecasting. In *ICLR*, 2026.

Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom. Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 387–395, 2018.

Ya Su, Youjian Zhao, Chenhao Niu, Rong Liu, Wei Sun, and Dan Pei. Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2828–2837, 2019.

Julien Audibert, Pietro Michiardi, Frédéric Guyard, Sébastien Marti, and Maria A Zuluaga. Usad: Unsupervised anomaly detection on multivariate time series. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 3395–3404, 2020.

Shreshth Tuli, Giuliano Casale, and Nicholas R Jennings. Tranad: Deep transformer networks for anomaly detection in multivariate time series data. *Proceedings of the VLDB Endowment*, 15(6): 1201–1214, 2022.

Jiehui Xu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Anomaly transformer: Time series anomaly detection with association discrepancy. In *International Conference on Learning Representations (ICLR)*, 2022.

Yiyuan Yang, Chaoli Zhang, Tian Zhou, Qingsong Wen, and Liang Sun. DCdetector: Dual attention contrastive representation learning for time series anomaly detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. Moment: A family of open time-series foundation models. In *International Conference on Machine Learning (ICML)*, 2024.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *Forty-first International Conference on Machine Learning*, 2024.

Shanghua Gao, Teddy Koker, Owen Queen, Thomas Hartvigsen, Theodoros Tsiligkaridis, and Marinka Zitnik. UniTS: A unified multi-task time series model. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Zhiguang Wang and Tim Oates. Imaging time-series to improve classification and imputation. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3939–3945, 2015.

Mouxiang Chen, Lefei Shen, Zhuo Li, Xiaoyun Joy Wang, Jianling Sun, and Chenghao Liu. VisionTS: Visual masked autoencoders are free-lunch zero-shot time series forecasters. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 8979–9007, 2025.

Shiyu Wang, Jiawei Li, Xiaoming Shi, Zhou Ye, Baichuan Mo, Wenze Lin, Shengtong Ju, Zhixuan Chu, and Ming Jin. Timemixer++: A general time series pattern machine for universal predictive analysis. *arXiv preprint arXiv:2410.16032*, 2024.

Sixun Dong, Wei Fan, Teresa Wu, and Yanjie Fu. Teaching time series to see and speak: Forecasting with aligned visual and textual perspectives. *ArXiv*, abs/2506.24124, 2025.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023.

Qinghua Liu and John Paparrizos. The elephant in the room: Towards a reliable time-series anomaly detection benchmark. In *NeurIPS 2024*, 2024.

Renjie Wu and Eamonn J. Keogh. Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress (extended abstract). In *Proceedings of 2022 IEEE 38th International Conference on Data Engineering*, pages 1479–1480, 2022.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *Forty-first International Conference on Machine Learning*, 2024.

Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 93–104, 2000.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 eighth ieee international conference on data mining*, pages 413–422. IEEE, 2008.

Alban Siffer, Pierre-Alain Fouque, Alexandre Termier, and Christine Largouet. Anomaly detection in streams with extreme value theory. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 1067–1075, 2017.

M Saquib Sarfraz, Mei-Yen Chen, Lukas Layer, Kunyu Peng, and Marios Koulakis. Position: Quo vadis, unsupervised time series anomaly detection? In *International Conference on Machine Learning*, pages 43461–43476. PMLR, 2024.

Alexis Huet, Jose Manuel Navarro, and Dario Rossi. Local evaluation of time series anomaly detection algorithms. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 635–645, 2022.

John Paparrizos, Paul Boniol, Themis Palpanas, Ruey S Tsay, Aaron Elmore, and Michael J Franklin. Volume under the surface: a new accuracy evaluation measure for time-series anomaly detection. *Proceedings of the VLDB Endowment*, 15(11):2774–2787, 2022.

A Multivariate model adaptations

In the multivariate setting, we adapt the model as follows:

For multivariate time series input, inspired by the proven efficacy of Channel-Independent (CI) architectures in recent time-series foundation models, we extend the aforementioned univariate **Traceable Image Conversion** to construct a unified three-channel heatmap representation. By treating each variable independently, we avoid the optimization bottlenecks in modeling complex, noisy cross-channel correlations in zero-shot scenarios. Specifically, each variable is first decomposed via Seasonal-Trend decomposition, and subsequently transformed through a variable-specific Gamma correction to apply differentiated non-linear mappings. To strictly preserve the zero-shot constraint, this Gamma parameter is dynamically computed during inference using only the statistics of the current input instance, preventing any leakage of target-domain global priors. This strategy enhances inter-variable discriminability in the visual domain by adapting to their distinct statistical distributions. To prevent distorted periodicities and signal discontinuities at variable boundaries, the folding period is calculated prior to concatenation. Specifically, we estimate the period for each variable independently and select their greatest common divisor as the unified folding period. The resulting individual heatmaps are then concatenated along the temporal dimension to form an integrated multivariate heatmap of size $1 \times (N_v L) \times 3$, where N_v denotes the number of variables and L the temporal length per variable. This composite representation then undergoes the folding and scaling operations. Crucially, to address the high dimensionality where the vertical axis after folding may far exceed the target resolution, an adaptive pooling operation is applied to dynamically downsample the vertical dimension to exactly 224 if it is greater than 224. This resolves the limitations of copy-padding in multivariate scenarios and yields a consistent and unified visual input.

Regarding **Patch-level Temporal Alignment**, the folding and scaling operations applied to the image features remain unchanged in the multivariate setting; the output is $\hat{F}_V \in \mathbb{R}^{N_{TS} \times N_v \times D_V}$. However, we inject an identical positional encoding vector E_{POS} into the token sequence of each variable. Furthermore, we adapt the self-attention mechanism by leveraging the any-variate attention [Woo et al., 2024], which processes each variable's temporal dynamics correction independently. Formally, the output is formulated as $F_V \in \mathbb{R}^{(N_{TS} N_v) \times D_{TS}}$. This design enables the model to separately reorganize the temporal order of features for each variable.

The **Anomaly Window Contrastive Learning** objective is applied independently to the feature sequence of each variable. Specifically, for every variable, we construct its own anomaly context window and perform positive/negative sample selection within its own temporal trajectory.

Finally, the **Task-adaptive Multi-modal Fusion** module remains unmodified in the multivariate setting, preserving its original architecture and functionality.

B Experimental details

B.1 Benchmark datasets

Our evaluation uses a selection of datasets from one primary source, the TSB-AD benchmark by Liu and Paparrizos [2024], including Univariate Datasets and Multivariate Datasets. The specific characteristics of these datasets, including their domain, number of time series (TS), average length, and anomaly ratio, are summarized in Table 5 and Table 6.

B.2 Baselines

To ensure a fair and consistent evaluation, the window sizes and hyperparameters for all baseline models are strictly configured according to the default settings recommended in their respective original papers.

Zero-Shot Models: These models are pre-trained on large-scale datasets and can be applied directly to new time series without fine-tuning.

- **Time-RCD**: This model proposed by [Lan et al., 2025] is a pre-trained general anomaly detector. We set the window size to 5000 and patch size is 16.

Table 5: Univariate Datasets

Name	Domain	#TS	Avg Length	AR (%)
UCR	Misc.	228	67818.7	0.6
NAB	Web	28	5099.7	10.6
YAHOO	Web	259	1560.2	0.6
IOPS	Operations	17	72792.3	1.3
MGAB	Sensor	9	97777.8	0.2
SED	Energy	3	23332.3	4.1
TODS	Traffic	15	5000.0	6.3
NEK	Weather	9	1073.0	8.0
Power	Power Grid	1	35040.0	8.5
Stock	Stock	20	15000.0	9.4
WSD	Web	111	17444.5	0.6

Table 6: Multivariate Datasets

Name	Domain	#TS	Avg Length	AR (%)
MSL	Space	16	3119.4	5.1
PSM	Sensor	1	217624.0	11.2
SMAP	Space	27	7855.9	2.9
SMD	Server	22	25466.4	3.8
SWaT	ICS	2	207457.5	12.7
GHL	Reservoir	25	199001.0	1.1
Daphnet	Medical	1	38774.0	5.9
Exathlon	Cluster	27	60878.4	9.8
Genesis	Robotics	1	16220.0	0.3
GECCO	Water	1	138521.0	1.2
MITDB	Medical	13	336153.8	2.7
SVDB	Medical	31	207122.6	4.8
LTDB	Medical	5	100000.0	15.5
CATSv2	Simulated	6	240000.0	3.7
TAO	Ocean	13	10000.0	8.7

- **DADA**: This model proposed by [Shentu et al., 2024] is a pre-trained general anomaly detector. We set the window size to 100.
- **TS-Pulse**: An IBM foundation model that uses a TSMixer architecture [Ekambaram et al., 2025]. It is pre-trained on a massive corpus for multi-task applications, including anomaly detection. Following the suggestion from the paper, we set the window size with a length of 96.
- **MOMENT**: A foundation model that utilizes a patch-based pre-training strategy to learn rich representations from diverse time series data [Goswami et al., 2024]. We use a window size of 64.
- **TimesFM**: A decoder-only transformer model from Google trained on a large time series corpus using a patching strategy, enabling strong zero-shot generalization [Das et al., 2024]. We set the window size to 96.
- **Chronos**: A generative model that frames time series analysis as a language modeling task, using a transformer-based architecture to learn and predict time series values [Ansari et al., 2024]. Its window size is 100.
- **Time MOE**: A decoder-only transformer model with a sparse Mixture-of-Experts (MoE) architecture. It is pre-trained on a large time series corpus for forecasting and multi-task learning [Shi et al., 2024]. We set the window size to 96.

Full-Shot Models: These models require training on the target dataset.

- **TranAD**: A transformer-based model that uses a reconstructive approach to detect anomalies by comparing original and reconstructed time series [Tuli et al., 2022]. It is configured with a window size of 10.
- **USAD**: An autoencoder-based model that employs an adversarial training mechanism to enhance its reconstruction capability and anomaly detection [Audibert et al., 2020]. We set the window size to 100.
- **OmniAnomaly**: A deep learning model that uses a Variational Autoencoder (VAE) with a Gated Recurrent Unit (GRU) to learn normal patterns and detect deviations [Su et al., 2019]. The model is configured with a window size of 100.
- **LOF**: A traditional statistical method that measures the local deviation of a data point from its neighbors, identifying anomalies with lower local density [Breunig et al., 2000]. For univariate datasets, we set `n_neighbors` to 50, and for multivariate, we set `n_neighbors` to 50 with metric as euclidean.
- **IForest**: An ensemble of Isolation Trees that isolates anomalies based on the number of random partitions required to separate them from the rest of the data [Liu et al., 2008]. For univariate datasets, we set `n_estimators` to 200, and for multivariate, we set `n_estimators` to 25 and `max_features` to 0.8.

Vision-based Models: These models use powerful pre-trained models for temporal anomaly detection without the need for additional training. It is important to note that prior vision-based methods do not provide native support for multivariate processing; therefore, our comparative evaluation against these baselines is restricted solely to univariate scenarios.

- **VIT4TS**: A lightweight anomaly screening module that uses a pretrained Vision Transformer to encode time series rendered as line plots from overlapping sliding windows (window length 224, stride 56) and computes cross-window patch-level dissimilarity against a median normal reference to generate high-resolution anomaly heatmaps without fine-tuning [He et al., 2026].
- **VLM4TS**: A zero-shot time series anomaly detection framework that combines a pretrained vision-language model (GPT-4o) with a two-stage pipeline. It first uses ViT4TS for candidate localization and then prompting the full VLM with a global line plot and textual candidate list to perform semantic validation and refinement of anomalies under a unified multimodal reasoning process [He et al., 2026].
- **VisualTimeAnomaly**: It converts univariate and multivariate time series into line-plot images (TSIs) and uses off-the-shelf multimodal large language models (GPT-4o) on point-wise, range-wise, and variate-wise anomaly detection tasks. [Xu et al., 2025].
- **AnomLLM**: It involves plotting time series data as line charts and leverages GPT-4o-mini's visual multimodal capabilities to detect anomaly patterns by directly "viewing" the plot, rather than analyzing numerical text. [Zhou and Yu, 2024].

B.3 Evaluation setting and zero-shot protocol

To ensure a rigorous and transparent evaluation, we explicitly clarify our zero-shot setting as follows:

1. **Definition of Zero-Shot Setting:** In this work, "zero-shot" strictly denotes **zero target-domain training or fine-tuning**. Following TSFM pre-training protocols [Shentu et al., 2024, Lan et al., 2025], VETime is trained solely with anomaly supervision on a synthetically generated corpus (0.5B points) constructed from mathematical equations. This synthetic pre-training incorporates zero statistical priors or labels from target benchmark datasets.
2. **Strict Target-Domain Isolation:** All model selection, early stopping, and hyperparameter configurations ($\lambda_{aw}, \lambda_e, \tau$) are determined exclusively on a held-out validation set split from our synthetic corpus. Any dataset-specific hyperparameter tuning on the target evaluation domains is strictly prohibited.
3. **Unsupervised Anomaly Thresholding:** For metric calculation (e.g., F1 scores), we employ the unsupervised Peaks-Over-Threshold (POT) dynamic thresholding algorithm [Siffer et al., 2017]. Anomaly thresholds δ are adaptively calculated for each individual time series

based solely on test-time predicted score distributions, completely without access to target benchmark ground-truth labels.

4. **Baseline Evaluation Protocols:** Zero-shot baselines are evaluated using their official pre-trained weights, as retraining unsupervised reconstruction models on anomaly-inclusive synthetic data would invalidate their detection mechanisms.

B.4 Evaluation metric calculations

Our performance evaluation is conducted using four metrics: Affiliation-F1, Temporal-F1 ($F1_T$), Standard-F1 ($F1$), and Volume Under Surface - Precision/Recall (VUS-PR).

- **Standard-F1** is a widely used metric that provides a harmonic mean of precision and recall. It is calculated using point-wise True Positives (TP), False Positives (FP), and False Negatives (FN).
- **F1-T** as described by Sarfraz et al. [2024], is a range-based metric that evaluates anomaly detection performance by considering the temporal context of anomalies. It is a variant of the F1-score that addresses common issues in time series evaluation, such as overlapping predictions and temporal proximity.
- **Affiliation-F1** is a distance-based metric that measures the "affiliation" or proximity between the ground truth and predicted anomaly points. It is designed to be less sensitive to minor temporal shifts in the predicted anomalies. The score is calculated by finding the optimal one-to-one mapping between the ground truth and detected anomalies and then computing the F1-score based on these affiliations [Huet et al., 2022].
- **VUS-PR** is a threshold-independent, parameter-free metric for time series anomaly detection. Unlike point-wise metrics, VUS-PR is robust to time lags and measures the area under a 3D surface plot of precision, recall, and a buffer parameter [Paparrizos et al., 2022]. It addresses the limitations of standard F1 scores by creating a continuous buffer region around each anomaly, thus providing a more reliable and nuanced evaluation of model performance.

B.5 Implementation details

Implementation: The proposed model was implemented using the PyTorch framework on Linux with one H20 24GB GPU. The input time series is divided into patches of size 16. The visual and temporal backbones employ the encoders of the frozen MAE [Chen et al., 2025] and pre-trained transformer architecture [Lan et al., 2025], respectively, with feature dimensions of 768 and 512. These are projected to a shared dimension of 512 during fusion. Hyper-parameters λ_{aw}, λ_e, and τ are fixed at 0.1, 0.2, and 0.1. The model is trained on synthetic data [Lan et al., 2025] with a batch size of 32, using AdamW optimizer with a learning rate of 5e-4 and a weight decay of 1e-5, for up to 25 epochs, with early stopping if no improvement is observed for 4 consecutive epochs. For extremely long sequences, VETime follows the sliding-window design of the temporal backbone and processes the input sequentially with a fixed window size of 5000 rather than compressing the full series into one image, which improves scalability with respect to sequence length.

Training data: To ensure robust zero-shot generalization, we constructed a large-scale synthetic dataset comprising a total of 0.5 billion data points following Lan et al. [2025]. The synthetic generation process relies strictly on predefined mathematical equations and does not incorporate any target-domain statistical priors, ensuring absolutely zero exposure to any target-domain data. The dataset features sequences with variable lengths ranging from 500 to 10,000 time steps. It is designed to cover a diverse spectrum of anomaly patterns, encompassing both long-term context deviations (e.g., trend shifts) and fine-grained local anomalies (e.g., point outliers).

Encoders: Our framework leverages a dual-encoder architecture to process heterogeneous modalities.

- **Vision Encoder:** We adopt the encoder from a pre-trained Masked Autoencoder (MAE) as our visual backbone. Specifically, we utilize the lightweight ViT-Base architecture to extract robust semantic representations from the converted time-series images.

- **Time-Series Encoder:** We employ a classical Transformer architecture consisting of learnable positional encodings, a stack of Encoder layers with Multi-Head Self-Attention (MHSA), and Feed-Forward Networks (FFN), followed by Layer Normalization. This encoder features 8 transformer layers, each with 8 attention heads. We initialize our encoder using the pre-trained weights from Time-RCD [Lan et al., 2025].

To achieve parameter-efficient adaptation, we apply Low-Rank Adaptation (LoRA) to fine-tune the linear projection matrices within both the Attention mechanism and the FFN modules in the time-series encoder. Specifically, we set the LoRA rank $r = 8$ and the scaling hyperparameter $\alpha = 16$, resulting in a scaling factor of $\alpha/r = 2$. LoRA adapters are inserted into all trainable linear layers of the encoder, while the original pre-trained weights remain frozen during fine-tuning.

AWCL window size selection: The AWCL window mechanism is exclusively active to compute contrastive losses during the synthetic pre-training phase. While its base size is informed by source-domain labels ($L_w = 2L_a$), the synthetic corpus inherently contains highly diverse anomaly lengths (L_a ranging from a few steps to hundreds). Consequently, the absolute physical receptive field of L_w is constantly shifting. This implicitly acts as scale-jittering, forcing AWCL to learn scale-invariant, domain-agnostic representations of local (intra) and global (inter) variations rather than overfitting to a specific curriculum ratio. During inference, the fused feature is simply concatenated from the bimodal features ($F_A = \text{FFN}([Z_{TS}; Z_V])$) without constructing windows. Therefore, when deploying to new datasets, there is no need to adjust any window size, ensuring ease of use and full zero-shot reproducibility.

Dual cross-attention: To facilitate deep interaction and integrate complementary contexts from the heterogeneous modalities, the aligned time-series features F_{TS} and visual features F_V are fed into a dual cross-attention module. Within this module, features from each modality serve as mutual queries to extract relevant information from the other. Formally, the process is defined as:

$$\begin{aligned} Z_{TS}^0 &= F_{TS} + \text{CrossAttn}(F_{TS}, F_V), & Z_V^0 &= F_V + \text{CrossAttn}(F_V, F_{TS}), \\ Z_{TS} &= Z_{TS}^0 + \text{FFN}\big(Z_{TS}^0\big), & Z_V &= Z_V^0 + \text{FFN}\big(Z_V^0\big), \end{aligned} \tag{7}$$

where

$$\text{CrossAttn}(X, Y) = \text{Softmax}\left(\frac{XW_Q(YW_K)^\top}{\sqrt{d}}\right)(YW_V), \tag{8}$$

denotes the cross-attention mechanism with X serving as queries and Y providing keys and values; W_Q, W_K, and $W_V \in \mathbb{R}^{d\times d}$ are learnable projection matrices; d is the dimension of the feature embeddings; and $\text{FFN}(\cdot)$ represents a feed-forward network applied with residual connection.

Loss: The task-specific features F_{AD} and F_{Rec}, derived from the fusion module, are passed through dedicated prediction heads to generate the predicted anomaly probabilities $\hat{y}_t = \text{Sigmoid}(\text{MLP}_{AD}(F_{AD})) \in (0, 1)$ and reconstructed sequence $\hat{x}_t = \text{MLP}_{Rec}(F_{Rec})$, where MLP_{AD} and MLP_{Rec} act as the classifier and projector, respectively.

The model is trained end-to-end using a composite loss function balancing classification accuracy, reconstruction quality, and feature alignment:

$$\mathcal{L}_{total} = \mathcal{L}_{BCE} + \mathcal{L}_{MSE} + \lambda_{aw}\mathcal{L}_{aw} + \lambda_e\mathcal{L}_e \tag{9}$$

where $\mathcal{L}_{BCE} = -\frac{1}{L}\sum_{t=1}^{L}\left[y_t \log(\hat{y}_t) + (1 - y_t)\log(1 - \hat{y}_t)\right]$ is the Binary Cross-Entropy loss over sequence length L with ground-truth binary labels $y_t \in \{0, 1\}$, and $\mathcal{L}_{MSE} = \frac{1}{L}\sum_{t=1}^{L}\|x_t - \hat{x}_t\|_2^2$ is the Mean Squared Error loss supervising sequence reconstruction against the normalized input x_t.

Inference pipeline:

1. **Anomaly Score Computation:** Patch-level features F_{AD} are projected to point resolution and mapped via the MLP head to per-timestamp anomaly scores $S_t \in (0, 1)$, relying purely on classification without reconstruction error.
2. **Thresholding and Binary Prediction:** Following standard protocols [Shentu et al., 2024, Tuli et al., 2022], we employ the unsupervised POT algorithm [Siffer et al., 2017] to

dynamically calculate thresholds δ independently for each time series. Specifically, the risk quantile hyperparameter q defining the extreme value distribution is fixed at the standard default of 10^{-4} across all evaluations, consistent with the setup in DADA [Shentu et al., 2024]. This adaptive thresholding requires no ground-truth labels or dataset-specific tuning, strictly preventing target-domain label leakage under a fair zero-shot setting.

C Additional experimental results

C.1 Comparison with Full-Shot Models

To comprehensively evaluate VETime, we compare its zero-shot performance against state-of-the-art full-shot baselines. Full-shot models are trained directly on the training splits of the target datasets, giving them an inherent advantage in capturing dataset-specific distributions. Remarkably, VETime, operating strictly in a zero-shot manner, achieves the best aggregate rank while remaining competitive across both univariate (Table 7) and multivariate (Table 8) scenarios.

Table 7: Performance of VETime (zero-shot) against full-shot baselines on 11 public univariate datasets. Red: best, Blue: second best.

Metric	Model	Univariate Datasets											Total 1st	Total 2nd	Avg Rank
		IOPS	MGAB	NAB	NEK	Power	SED	Stock	TODS	UCR	WSD	YAHOO			
Full-Shot Models															
Affiliation-F1	VETime	**90.53**	68.03	88.57	79.56	**78.78**	**97.31**	69.60	**85.85**	**85.06**	**94.31**	**97.15**	**07**	00	**1.91**
	TranAD	**83.19**	67.28	90.28	**85.02**	71.56	61.03	57.94	52.76	73.31	**84.34**	**76.08**	00	**04**	3.45
	USAD	71.08	67.81	**91.54**	71.13	76.48	55.60	35.92	47.90	**76.00**	65.10	53.05	00	**02**	4.36
	OmniAnomaly	80.32	67.35	**92.35**	**86.30**	**78.16**	61.26	**75.24**	50.73	73.53	78.02	71.31	**03**	01	**3.00**
	LOF	81.06	**68.44**	75.75	84.74	66.76	**63.85**	**69.74**	**60.58**	73.53	81.29	75.63	00	**04**	3.09
	IForest	52.81	**68.82**	39.84	71.15	00.00	70.09	0.06	44.17	50.56	41.24	33.30	01	00	5.09
F1_T	VETime	**46.15**	**2.00**	**44.31**	60.86	20.16	**74.35**	22.41	**68.33**	**36.25**	**50.42**	**91.54**	**05**	**03**	**2.09**
	TranAD	22.63	01.65	37.28	69.97	22.36	09.57	16.73	13.51	07.75	20.94	08.41	00	00	4.18
	USAD	20.99	**04.07**	**61.46**	**70.64**	**28.23**	09.54	16.86	20.85	**14.63**	14.18	09.35	**03**	02	3.18
	OmniAnomaly	**51.17**	01.61	40.09	**82.20**	**23.48**	**09.68**	**36.22**	14.33	08.47	**34.79**	24.16	02	**04**	**3.09**
	LOF	27.97	01.15	35.76	63.57	19.80	09.60	**66.14**	**31.63**	08.31	24.38	**55.93**	01	02	3.45
	IForest	07.64	00.84	21.44	65.56	00.00	09.54	1.10	11.06	06.36	4.28	04.90	00	00	5.00
Standard-F1	VETime	34.56	**1.89**	**37.87**	53.95	20.12	**75.02**	23.83	**69.13**	**30.61**	**46.55**	**88.93**	**05**	**02**	**2.00**
	TranAD	**34.85**	01.46	27.33	60.36	22.36	02.63	16.23	11.94	04.40	20.23	05.70	00	01	3.91
	USAD	30.66	**03.89**	**56.15**	**62.91**	**28.24**	03.41	17.99	23.87	**10.74**	13.20	07.21	**03**	**02**	**2.73**
	OmniAnomaly	**47.05**	01.44	28.81	**74.03**	**23.50**	00.43	**38.59**	12.65	05.11	**29.57**	21.40	02	**03**	3.36
	LOF	30.28	01.05	24.04	56.92	12.18	**04.11**	**66.20**	**25.77**	04.70	22.62	**48.95**	01	**03**	3.55
	IForest	08.37	00.73	29.41	58.10	19.77	03.81	16.91	13.35	04.09	2.07	03.20	00	00	5.09
VUS-PR	VETime	**30.79**	**0.70**	**32.98**	57.54	11.78	**89.05**	76.08	**94.20**	**24.99**	**33.85**	**88.61**	**06**	**02**	**2.09**
	TranAD	21.61	00.64	24.82	**61.63**	13.04	05.75	78.08	47.33	02.25	12.20	25.78	00	01	3.73
	USAD	16.58	**00.75**	**55.03**	58.53	**18.68**	04.37	74.53	**56.36**	**08.85**	10.00	14.15	**03**	**02**	3.27
	OmniAnomaly	**25.35**	00.64	27.17	**74.51**	**14.32**	06.20	**91.29**	45.55	02.40	**16.37**	29.26	02	**03**	**2.64**
	LOF	19.43	00.57	21.18	58.52	09.31	06.81	**83.07**	49.14	02.39	12.85	**41.37**	00	**02**	4.18
	IForest	08.59	00.62	23.57	56.50	11.56	**07.71**	70.99	46.62	02.88	2.06	10.47	00	01	5.00
Grand Total (Full-Shot)													**23**	**07**	**2.02**

Table 8: Performance of VETime (zero-shot) against full-shot baselines on multivariate datasets. Red: best, Blue: second best.

Metric	Model	Multivariate Datasets					Additional Datasets										Total 1st	Total 2nd
		MSL	PSM	SMAP	SMD	SWaT	GHL	Daph.	Exath.	Gen.	GECCO	MIT	SVDB	LTDB	CATS	TAO		
Full-Shot Models																		
Affiliation-F1	VETime	**87.02**	**77.27**	87.06	**88.94**	**75.45**	**74.93**	**84.61**	**99.90**	**75.41**	**87.63**	**75.16**	**73.03**	**78.28**	72.08	**75.41**	12	01
	TranAD	79.91	**73.83**	**87.39**	**92.20**	**75.37**	39.00	**72.00**	**94.00**	**69.00**	**26.00**	**71.00**	71.00	64.00	**75.00**	**41.00**	02	09
	USAD	81.86	57.86	87.25	85.09	75.06	35.95	19.82	84.96	46.84	14.51	63.28	58.33	35.81	57.04	2.05	00	00
	OmniAnomaly	**83.15**	58.17	**91.38**	85.82	73.39	**40.90**	67.74	93.89	68.63	18.28	70.95	**71.96**	**64.86**	**74.32**	20.31	01	05
F1_T	VETime	43.52	**39.06**	**45.52**	**53.36**	39.94	**24.12**	**55.42**	**97.45**	3.06	20.25	**17.80**	**16.17**	**32.10**	21.12	**28.32**	05	05
	TranAD	39.42	25.49	29.12	37.98	**49.58**	4.00	17.00	95.00	**16.00**	40.00	16.00	15.00	30.00	24.00	18.00	01	01
	USAD	**48.71**	28.96	43.94	50.41	**50.41**	**4.06**	**23.29**	**97.08**	2.82	**44.79**	**28.27**	**51.27**	**51.31**	**35.27**	17.31	06	04
	OmniAnomaly	**49.36**	**30.42**	**46.63**	**51.84**	46.64	3.92	22.74	96.00	**6.19**	**41.59**	15.74	15.83	30.77	**24.06**	**37.58**	03	05
Standard-F1	VETime	35.24	27.05	32.16	51.41	55.87	**11.72**	**41.06**	**97.43**	2.98	29.31	**16.74**	**15.63**	**32.02**	**19.88**	**30.43**	05	03
	TranAD	29.60	25.63	25.11	43.99	**61.86**	**0.00**	1.00	37.00	**4.00**	33.00	7.00	7.00	5.00	14.00	5.00	00	03
	USAD	**38.71**	**28.41**	**38.66**	**53.06**	**62.82**	**0.00**	0.00	**46.87**	0.00	**42.82**	**20.87**	**33.84**	**11.84**	**25.87**	0.50	05	07
	OmniAnomaly	**39.10**	**30.43**	**40.50**	**57.06**	55.93	**0.00**	**1.77**	27.65	**4.19**	**35.48**	7.25	8.86	6.23	19.76	**13.84**	05	04
VUS-PR	VETime	**30.51**	**23.86**	23.70	**48.21**	42.93	**4.82**	**35.09**	**99.77**	**3.08**	15.37	**14.19**	**11.87**	**27.11**	12.00	**92.15**	08	03
	TranAD	14.78	16.49	13.37	28.34	**47.37**	1.00	14.00	95.00	**3.00**	**19.00**	6.00	9.00	21.00	**17.00**	9.00	02	02
	USAD	29.95	17.59	**26.37**	34.53	**44.73**	1.30	11.54	**96.43**	1.17	8.37	**12.18**	**43.75**	**48.84**	**20.93**	9.60	03	04
	OmniAnomaly	**31.57**	**18.58**	**28.07**	**37.44**	42.97	**1.31**	**14.27**	96.19	1.63	**17.43**	6.12	10.60	22.28	16.97	**26.04**	02	06
VETime Grand Total (Full-Shot)																	**30**	**12**

C.2 Statistical support

To provide robust statistical evidence for our evaluations, we conduct further analyses covering training stability across multiple random seeds and confidence interval estimations.

Training stability: To ensure a rigorous and controlled comparison, we retrained VETime alongside methods that support anomaly supervision (TimeRCD and DADA) on the same synthetic dataset across 5 independent random seeds. As shown in Table 9, VETime achieves top-tier overall performance with remarkably low standard deviations (generally $< 1\%$), confirming that its empirical gains stem directly from architectural advancements rather than data advantages.

Table 9: Performance comparison of supervised baselines re-trained on the same synthetic dataset (mean ± standard deviation over 5 runs). Best result is in red, second-best is in blue.

Metric	Method	IOPS	MGAB	NAB	NEK	Power	SED	Stock	TODS	UCR	WSD	YAHOO
Affiliation-F1	TimeRCD	**83.58±0.62**	**70.19±1.39**	82.17±1.99	**78.56±2.20**	**85.25±1.71**	**96.90±0.42**	**71.66±1.05**	**86.66±2.60**	**84.78±0.40**	**90.12±0.23**	**96.54±0.37**
	DADA	73.00±0.92	66.28±1.66	**84.51±0.63**	78.20±1.81	68.05±0.87	68.20±1.58	**96.55±0.49**	66.31±0.32	66.36±0.23	86.10±0.73	90.46±0.64
	VETime	**90.22±0.66**	**68.18±0.18**	**88.19±0.43**	**79.95±0.97**	**78.99±0.33**	**97.95±0.98**	69.45±0.61	**85.43±0.54**	**85.06±0.24**	**94.28±0.06**	**97.12±0.16**
VUS-PR	TimeRCD	21.82±3.79	**1.03±0.06**	24.42±1.35	27.47±2.65	**21.17±0.15**	**80.38±0.90**	**76.72±1.45**	**93.29±1.69**	**23.08±0.07**	21.94±2.71	**84.23±0.75**
	DADA	**24.77±1.85**	0.62±0.04	**24.46±0.79**	**64.82±2.41**	**12.42±0.64**	6.43±0.26	**99.79±0.02**	57.47±0.11	2.12±0.65	**30.38±1.05**	73.05±0.47
	VETime	**30.84±1.62**	**0.69±0.02**	**32.84±0.21**	**56.42±2.71**	11.85±0.18	**88.37±2.24**	76.21±0.78	**93.82±0.53**	**24.67±0.78**	**33.16±0.65**	**87.92±1.06**

Confidence intervals: We calculated the 95% confidence intervals (CIs) using the sequence-level Bootstrapping method, which resamples intact individual time series with replacement 1000 times per dataset. The detailed performance across zero-shot, full-shot, and vision-based baselines is summarized in Table 10 and Table 11. Across the 11 univariate datasets, we performed Friedman tests followed by two-sided Wilcoxon signed-rank tests with Holm correction. A significant method effect was found for full-shot Affiliation-F1 ($p = 0.0076$), where VETime significantly outperforms TranAD ($p_{\text{Holm}} = 0.0488$) and USAD ($p_{\text{Holm}} = 0.0146$). For other metrics and panels, VETime consistently achieves the best average rank while demonstrating sustained advantages across diverse benchmarks.

Table 10: Performance of VETime against zero-shot and full-shot time-series baselines across 11 datasets (Mean ± 95% CI). Best result is in red, second-best is in blue.

Metric	Method	IOPS	MGAB	NAB	NEK	Power	SED	Stock	TODS	UCR	WSD	YAHOO
Zero-Shot Models												
Affiliation-F1	VETime	**90.39±3.10**	**68.17±0.61**	**88.36±3.38**	74.96±17.28	**78.78±0.00**	**97.08±1.34**	69.63±0.54	85.84±5.26	**85.08±1.54**	93.25±1.48	**97.07±1.79**
	TimeRCD	83.37±3.67	**70.71±1.85**	83.07±9.54	82.39±12.56	**85.51±0.00**	**96.48±1.79**	71.77±1.49	**86.58±8.30**	**84.68±1.86**	90.36±2.70	**96.75±1.29**
	DADA	**89.61±1.51**	67.70±0.40	87.51±6.79	**95.79±0.13**	69.79±0.00	65.18±2.03	**98.83±0.17**	76.91±12.69	72.20±0.83	**93.81±2.72**	92.33±3.51
	MOMENT	87.60±2.02	67.08±3.42	**91.18±4.32**	91.53±5.67	75.97±0.00	58.19±4.52	45.28±0.91	59.76±13.08	75.61±2.55	**95.46±1.86**	79.81±3.79
	TimesFM	81.93±3.85	66.84±0.95	79.63±3.86	**95.17±1.76**	69.88±0.00	66.18±3.40	**97.51±0.42**	**88.91±4.93**	69.97±1.22	78.97±1.75	91.34±1.81
$F1_T$	VETime	**44.20±8.13**	**2.01±0.51**	**44.22±6.53**	**62.28±12.04**	**20.16±0.00**	**76.05±7.68**	22.45±1.35	**68.56±8.75**	**36.42±3.95**	**50.28±3.88**	**91.54±1.65**
	TimeRCD	27.96±6.46	1.73±0.38	36.15±5.99	34.58±13.81	**28.47±0.00**	**70.34±3.89**	31.17±2.37	**64.97±9.65**	32.60±3.37	35.80±3.94	**85.54±2.63**
	DADA	42.01±7.18	**1.82±0.34**	37.79±5.35	48.88±15.55	19.80±0.00	9.46±1.67	**94.63±1.85**	34.39±4.91	6.88±1.15	**48.28±3.89**	79.26±2.83
	MOMENT	31.35±9.57	0.77±0.10	**50.81±7.45**	**63.13±9.90**	19.91±0.00	8.62±1.03	19.73±2.91	16.06±2.55	13.75±2.67	41.21±3.70	10.88±1.50
	TimesFM	**45.48±8.65**	1.02±0.24	34.54±6.73	38.38±11.31	19.80±0.00	9.69±2.15	**88.98±1.51**	52.17±9.34	**41.94±3.21**	10.10±2.07	83.89±2.59
Standard-F1	VETime	**35.01±7.54**	**1.89±0.49**	**38.06±5.95**	**52.16±15.88**	**20.12±0.00**	**76.71±7.70**	23.87±1.63	**69.39±8.49**	**30.71±3.92**	**46.55±4.08**	**88.85±2.52**
	TimeRCD	22.61±4.14	**1.40±0.32**	27.20±4.71	32.03±11.17	**28.59±0.00**	**69.04±2.48**	32.51±0.86	**66.97±7.47**	27.78±3.28	29.48±3.32	**87.48±3.38**
	DADA	32.70±1.22	0.96±0.32	28.73±3.74	47.22±6.94	15.99±0.00	2.56±0.23	**95.39±1.50**	28.35±2.35	3.90±0.56	**45.21±4.16**	81.80±4.21
	MOMENT	29.90±7.38	0.66±0.05	**44.35±7.68**	**63.99±11.45**	16.39±0.00	3.34±0.15	19.88±0.49	14.18±3.37	9.59±2.29	41.05±3.34	10.52±0.76
	TimesFM	**35.95±6.34**	0.98±0.22	27.15±6.03	34.86±5.82	16.73±0.00	2.02±0.12	**89.47±1.77**	37.20±10.14	8.27±2.74	37.05±3.25	84.76±3.02
VUS-PR	VETime	**31.78±10.95**	0.71±0.15	**33.13±7.59**	**55.42±16.37**	**11.78±0.00**	**89.98±4.68**	75.32±10.69	**93.89±2.76**	**25.20±4.03**	33.81±4.68	**88.60±2.82**
	TimeRCD	19.85±6.20	**1.03±0.28**	24.15±7.91	27.81±12.61	**21.25±0.00**	**80.94±2.46**	**77.61±11.69**	**93.63±2.57**	23.02±3.83	21.52±3.82	84.46±3.26
	DADA	24.43±7.39	0.57±0.06	24.49±4.05	49.47±22.32	10.61±0.00	6.58±1.06	**99.52±0.14**	64.95±12.18	2.89±0.41	33.35±3.86	70.78±4.82
	MOMENT	**36.95±10.90**	0.55±0.05	**45.28±9.09**	**68.21±12.20**	10.50±0.00	4.27±0.46	77.25±10.59	57.00±11.36	6.12±1.72	**55.51±5.07**	30.72±3.30
	TimesFM	18.79±11.64	0.59±0.14	23.64±8.22	38.78±17.28	10.44±0.00	8.01±2.61	**98.41±0.43**	73.01±4.13	5.90±4.01	21.43±4.95	**86.77±3.37**
Full-Shot Models												
Affiliation-F1	VETime	**90.39±3.10**	**68.17±0.61**	88.36±3.38	74.96±17.28	**78.78±0.00**	**97.08±1.34**	**69.63±0.54**	**85.84±5.26**	**85.08±1.54**	**93.25±1.48**	**97.07±1.79**
	TranAD	**83.71±11.01**	67.35±0.44	90.52±6.41	**85.77±6.84**	71.56±0.00	61.03±6.15	58.05±4.84	**51.81±10.95**	73.29±1.32	**84.16±3.10**	**76.12±3.71**
	USAD	70.05±16.57	**67.79±1.45**	**92.32±3.36**	70.44±15.64	76.48±0.00	55.60±4.76	35.93±0.57	45.18±12.40	**75.99±3.17**	65.33±6.61	52.57±6.88
	OmniAnomaly	80.95±11.65	67.40±0.51	**92.44±6.13**	**87.10±8.79**	**78.16±0.00**	**61.26±3.16**	**75.28±3.83**	50.74±5.92	73.46±1.38	77.93±5.24	71.18±3.55
$F1_T$	VETime	**44.20±8.13**	**2.01±0.51**	**44.22±6.53**	62.28±12.04	20.16±0.00	**76.05±7.68**	**22.45±1.35**	**68.56±8.75**	**36.42±3.95**	**50.28±3.88**	**91.54±1.65**
	TranAD	26.77±11.70	1.47±0.30	35.04±3.76	63.45±6.34	22.36±0.00	8.13±1.03	17.66±3.22	12.68±1.47	7.03±0.87	21.66±3.79	8.79±1.43
	USAD	21.61±12.58	**4.28±0.70**	**61.90±7.73**	**72.68±16.15**	**28.23±0.00**	**9.12±1.03**	16.67±3.10	**23.89±6.86**	**13.47±2.85**	14.32±3.55	9.27±1.43
	OmniAnomaly	**48.57±12.92**	1.33±0.27	36.36±3.43	**81.01±9.07**	**23.48±0.00**	8.73±1.03	**37.52±1.54**	13.28±1.36	8.64±1.06	**32.90±6.02**	**19.65±3.53**
Standard-F1	VETime	**35.01±7.54**	**1.89±0.49**	**38.06±5.95**	52.16±15.88	20.12±0.00	**76.71±7.70**	**23.87±1.63**	**69.39±8.49**	**30.71±3.92**	**46.55±4.08**	**88.85±2.52**
	TranAD	**31.97±9.90**	1.43±0.27	27.85±3.36	60.02±7.47	22.36±0.00	2.34±0.30	16.12±0.94	10.52±1.33	4.43±0.52	19.93±3.19	5.42±0.76
	USAD	22.68±12.58	**2.27±1.18**	**52.12±8.68**	**64.55±12.33**	**28.24±0.00**	**3.44±0.52**	17.40±0.71	**24.39±5.09**	**9.77±2.42**	12.92±3.05	7.38±0.88
	OmniAnomaly	32.39±14.05	1.22±0.25	26.03±4.12	**72.42±5.26**	**23.50±0.00**	0.43±0.13	**38.03±2.10**	12.97±0.44	5.58±0.59	**28.22±5.16**	**21.52±2.62**
VUS-PR	VETime	**31.78±10.95**	**0.71±0.15**	**33.13±7.59**	55.42±16.37	11.78±0.00	**89.98±4.68**	75.32±10.69	**93.89±2.76**	**25.20±4.03**	**33.81±4.68**	**88.60±2.82**
	TranAD	21.44±6.34	0.64±0.05	24.67±4.08	**61.72±16.02**	13.04±0.00	5.72±0.29	**78.49±11.04**	**47.89±12.51**	2.24±0.35	12.04±2.99	**25.63±2.40**
	USAD	16.11±4.62	**0.72±0.14**	**54.65±9.38**	59.23±17.84	**18.68±0.00**	**4.30±0.46**	74.74±11.12	57.26±11.32	**8.71±2.43**	10.00±2.71	14.11±1.56
	OmniAnomaly	**24.64±7.11**	0.63±0.05	27.04±4.13	**74.63±9.01**	**14.32±0.00**	6.23±0.54	**91.52±4.90**	46.19±12.01	2.37±0.43	**16.19±3.70**	**29.30±4.10**

C.3 Comparison under different settings

Comparison under different scalability. Figure 9 evaluates the scalability of VETime with respect to varying sequence lengths and variable dimensions. For extremely long sequences, rather than naively compressing the entire series into a single visual representation, VETime adopts a sliding-window mechanism (with a window size of 5,000) inherited from its temporal backbone. This design choice ensures stable performance and prevents resolution degradation as sequence length increases.

Furthermore, when handling high-dimensional multivariate data, VETime maintains a highly competitive average rank as the number of variables grows. However, mapping multivariate data onto a fixed-size visual canvas (224×224) over a constant temporal window (5000 steps) introduces a

Table 11: Performance of VETime against vision-based baselines across 4 datasets (Mean ± 95% CI). Best result is in red, second-best is in blue.

Metric	Method	NAB	YAHOO	SMAP	MSL
Affiliation-F1	VETime	**88.36±3.38**	**97.07±1.79**	**95.13±3.72**	**90.66±2.65**
	VIT4TS	**64.64±11.91**	**65.48±3.67**	93.55±4.20	**69.90±22.83**
	VLM4TS	60.50±13.28	64.27±3.66	**94.48±3.42**	64.63±25.15
	VisualTimeAnomaly	62.12±4.25	60.80±3.10	90.89±3.76	86.23±3.33
	AnomLLM	58.23±10.61	36.36±4.24	42.03±20.36	46.34±20.64
$F1_T$	VETime	**44.22±6.53**	**91.54±1.65**	**66.04±4.28**	**49.07±6.93**
	VIT4TS	30.07±3.03	8.05±1.77	51.13±2.58	23.48±8.77
	VLM4TS	28.24±2.62	**9.44±1.73**	**58.59±3.29**	**38.81±2.54**
	VisualTimeAnomaly	**40.34±5.73**	2.58±0.47	13.32±2.11	14.88±2.87
	AnomLLM	16.53±4.56	1.34±0.80	4.86±1.31	3.12±1.08
Standard-F1	VETime	**38.06±5.95**	**88.85±2.52**	**62.23±4.98**	**53.04±5.06**
	VIT4TS	**35.75±7.73**	7.20±2.11	52.56±2.33	27.54±5.97
	VLM4TS	33.81±7.20	**9.45±0.94**	**58.90±3.33**	**45.38±7.80**
	VisualTimeAnomaly	32.72±5.69	3.86±0.57	16.53±0.88	15.39±3.52
	AnomLLM	22.39±3.81	2.29±0.17	6.73±1.64	10.21±1.74
VUS-PR	VETime	**33.13±7.59**	**88.60±2.82**	**60.63±11.56**	**39.34±16.01**
	VIT4TS	**27.20±5.43**	**34.63±3.55**	51.83±9.51	27.88±14.78
	VLM4TS	25.71±5.33	33.30±3.44	**52.92±8.87**	**29.32±15.32**
	VisualTimeAnomaly	**27.20±9.34**	13.15±1.78	12.70±7.33	20.73±15.02
	AnomLLM	19.58±5.20	10.49±1.45	5.26±2.74	10.12±3.94
Per-series runtime (ms)	VETime	**36.54±6.89**	**24.64±0.03**	**47.06±3.19**	**74.11±48.72**
	VIT4TS	**1733.8±230.5**	2764.9±402.3	**2440.1±885.0**	6553.2±442.3
	VLM4TS	6730.6±596.5	7416.8±389.8	7148.8±996.6	9958.2±402.1
	VisualTimeAnomaly	1980.6±129.3	**2075.1±600.0**	3388.6±172.3	**1832.0±232.6**
	AnomLLM	3593.9±256.4	3957.5±140.1	3501.7±539.3	3928.4±311.5

theoretical risk of information bottlenecks. A practical physical upper limit of approximately 10 variables ($224 \times 224/5000 \approx 10$) ensures that each data point occupies at least one distinct pixel without severe spatial overlap (aliasing). Exceeding this limit severely compresses the per-timestamp pixel representation, leading to visual information sparsity and a loss of fine-grained point-level details. As shown in Figure 9, empirical results indicate that performance experiences some degradation in the mid-to-high dimensional range (25-38 variables) due to this bottleneck.

Interestingly, VETime remains highly competitive on the 59-variable SMAP dataset, demonstrating that our framework remains robust and effective even in ultra-high-dimensional settings. This indicates that the actual failure mode is tied to **information redundancy** rather than just variable count. Datasets with highly correlated variables (like SMAP) suffer less from the visual compression bottleneck.

C.4 Normal-anomaly feature separability

To further assess the discriminative power of the learned representations, we evaluated normal-anomaly feature separability of 1D temporal (F_{TS}), 2D visual (F_V), and fused (F_{AD}) outputs using the Calinski-Harabasz (CH) Index and Silhouette Coefficient. As shown in Figure 10, F_{TS} excels at capturing local spikes, while F_V trades local precision for stronger global awareness. Crucially, the fused representation F_{AD} successfully synergizes both, achieving the highest separability across all scenarios.

The unique advantages of the 2D visual branch stem from three key mechanisms:

- **Cross-Periodic Dependencies via folding:** Following the 2D temporal modeling principles established in TimesNet [Wu et al., 2023], the folding mechanism converts these long-range temporal distances into vertical spatial adjacencies. This enables the 2D backbone to efficiently gain cross-periodic patterns without requiring inefficiently large 1D receptive fields.
- **Cross-Component Interaction via Channel Mapping:** By mapping the raw signal, Trend, and Remainder components into distinct RGB channels, ViT integrates the complex interplay between trend and residual information into unified token embeddings, naturally achieving early cross-component fusion.

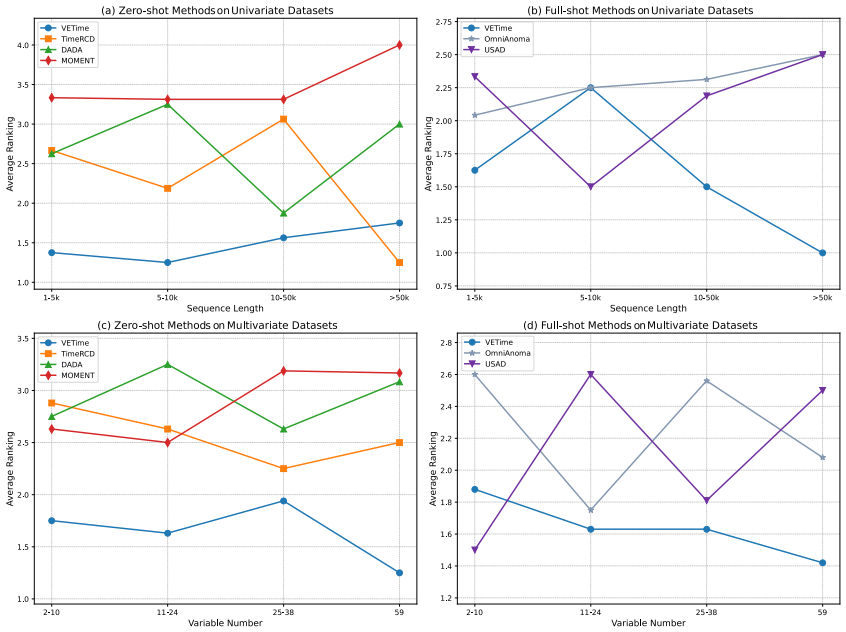


Figure 9: Performance of VETime (strictly zero-shot) against zero-shot and full-shot models. Metrics: Average Rank (lower is better) across 4 metrics on different datasets.

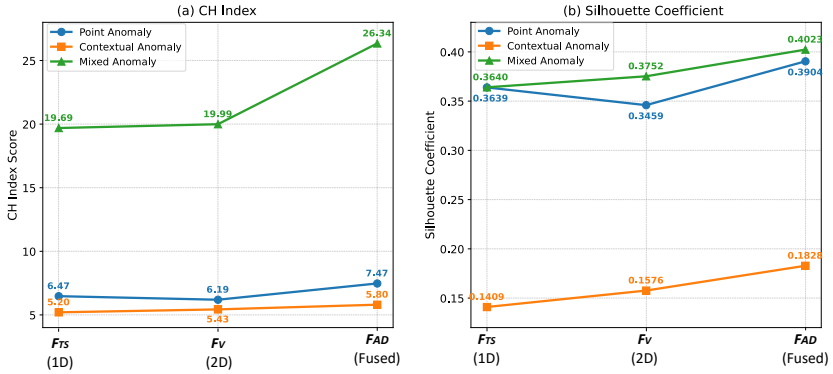


Figure 10: Quantitative validation in feature separability between normal and anomalous representations. The clustering metrics clearly illustrate that the 2D visual transformation introduces global awareness, while the final fusion F_{AD} maximizes the overall normal/anomalous separability.

- **Rich Contextual Patching:** For a length-5000 sequence, unlike 1D temporal patching yielding 313 localized, low-level numerical tokens (patch size 16), our 2D transformation constructs 196 structured tokens (patch size 16x16) derived from a 224×224 representation. Rather than diluting information, each visual patch intrinsically aggregates aforementioned cross-periodic dependencies and multi-component interactions, significantly amplifying the semantic density compared to 1D numerical tokens.

Moreover, pre-trained ViTs function as robust local edge detectors and global texture matchers [Chen et al., 2025]. By transforming cross-periodic dynamics into spatial adjacencies, anomalies naturally manifest as structural edge breaks or localized texture distortions, aligning perfectly with the inductive biases of natural image perception.

C.5 Efficiency & performance.

Table 12 compares the inference costs and parameters of VETime against a temporal-only baseline (TS Encoder + Classifier) and a size-matched variant (Dual-TS) that replaces the visual encoder with

Table 12: Efficiency and performance comparison. Metrics are reported as Affiliation-F1 / VUS-PR (%). Best result is in red, second-best is in blue.

Model	Params. (M)	Mem. (MB)	Avg. Time (ms)	NAB	WSD	YAHOO
Base	37.06	223.73	24.49	**82.48** / 24.32	**90.33** / 21.77	96.65 / 84.41
Dual-TS	160.49	928.52	48.19	80.84 / **30.28**	88.87 / **27.24**	**96.81** / **87.25**
VETime	157.96	787.41	51.02	**88.57** / **32.98**	**94.31** / **33.85**	**97.15** / **88.61**

a temporal Transformer. While VETime incurs higher overhead than the baseline, it yields substantial gains. Under comparable costs, VETime significantly outperforms Dual-TS. This confirms our gains stem from the visual-temporal dual-branch design rather than mere parameter expansion, making the complexity a worthwhile trade-off.

D Ablation studies and in-depth analysis

D.1 Ablation study on vision backbones

We evaluate VETime across different vision architectures, pre-training paradigms, and scales, with the comprehensive results summarized in Table 13.

- **Architecture:** VETime implemented with a patch-tokenized ViT backbone significantly outperforms representative CNNs (ResNet, ConvNeXt-Base). This superiority arises because ViT's patch-based tokenization inherently aligns with our patch-level temporal alignment design, whereas CNN's spatial downsampling inherently disrupts the strict temporal alignment required for precise anomaly localization.
- **Pre-training Paradigm:** Among ViT architectures, MAE-based encoders consistently perform better than CLIP-based ones. This is likely because reconstruction-style pre-training better preserves the fine-grained structural cues that are critical for identifying anomalous patterns.
- **Scaling:** MAE-Base delivers the optimal balance of efficiency and performance. Scaling up the vision encoder to MAE-Large yields only marginal and inconsistent gains, which do not justify the $3.5\times$ increase in parameters.

D.2 Comparison of the TS-to-image strategy

To quantitatively validate the effectiveness of visual representations, the time series from 11 univariate datasets are categorized as: Point (short-duration spikes), Context (long-duration structural shifts), and Mixed. Empirical visualizations (Figure 11) and the comparison (Table 14) confirm our method outperforms others in identifying contextual anomalies. Unlike unaligned transformations (GAF/RP) that disrupt temporal semantics or sparse line plots (ViT4TS) causing massive spatial redundancy, VETime constructs a dense, patch-alignable 2D representation.

Motivated by the success of folded 2D spaces in capturing intra- and inter-period variations, we utilize dense intensity mappings combined with adaptive folding to preserve strict temporal alignment. Furthermore, mapping the raw series, trend, and remainder components into distinct RGB

Table 13: Ablation study and comparison of vision encoders (Affiliation-F1 / VUS-PR (%)). Best result is in red, second-best is in blue.

Model	Params. (M)	NAB	WSD	YAHOO
Resnet	25.63	78.41 / 22.96	75.17 / 13.57	70.85 / 15.75
ConvNeXt-Base	88.58	78.51 / 23.36	82.15 / 25.38	79.73 / 38.90
CLIP-Base	85.64	88.42 / **32.72**	93.01 / 30.90	95.87 / 86.12
MAE-Base	85.64	**88.57** / **32.98**	**94.31** / **33.85**	**97.15** / **88.61**
MAE-Large	303.10	**88.67** / 31.54	**93.94** / **34.10**	**97.03** / **87.20**

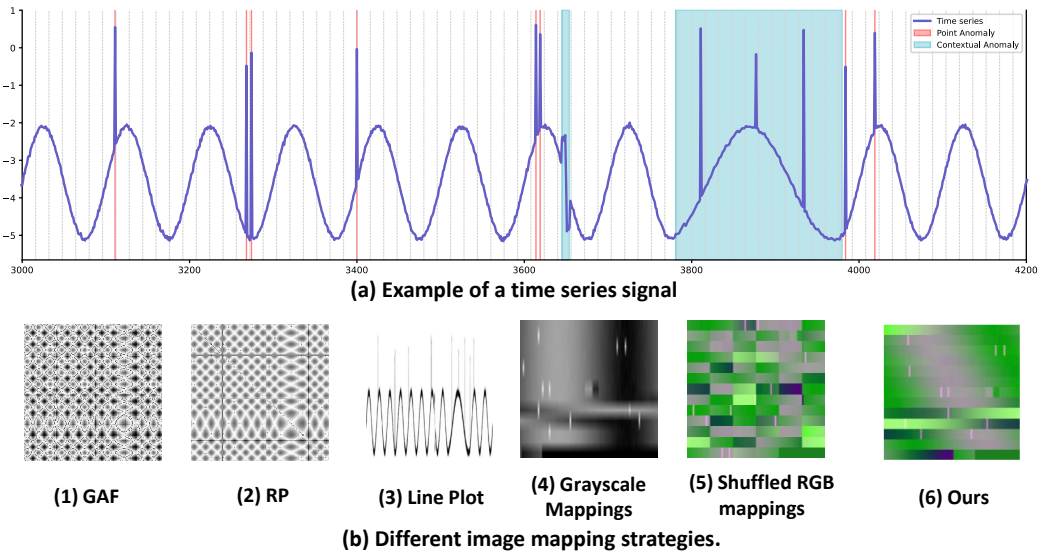


Figure 11: Qualitative comparison of different imaging strategies.

channels disentangles complex temporal signals. This design enables the Vision Transformer (ViT) to effectively aggregate periodicity and amplitude variations, thereby exposing long-range disruptions.

Table 14: Performance comparison under different imaging strategies (Affiliation-F1 / VUS-PR (%)). The time series from 11 univariate datasets are categorized as: *Point (short-duration spikes), Contextual (long-duration structural shifts), and Mixed*. *Base*: TS Encoder + Classifier. *Shuffled RGB mappings*: Temporal order shuffled before mapping and restored before alignment. Best result is in **bold red**, second-best is in **bold blue**.

Imaging Strategies	Point	Contextual	Mixed
Base	85.65 / 67.31	77.18 / 23.89	83.90 / 39.35
(1) GAF	86.12 / 72.88	77.01 / 23.57	84.63 / 42.05
(2) RP	85.79 / 73.91	77.07 / 23.87	84.88 / 42.65
(3) Line plot	86.14 / 74.82	78.20 / 24.92	84.77 / 45.31
(4) Gray mappings	87.97 / 77.74	**79.50** / **26.65**	**86.07** / **46.48**
(5) Shuffled RGB mappings	**88.75** / **78.42**	78.94 / 25.51	84.43 / 46.36
(6) Ours	**89.15** / **79.78**	**80.54** / **28.35**	**86.71** / **47.99**

Effectiveness of Traceable Image Conversion (TIC). While basic operations such as seasonal-trend decomposition or RGB mapping are individually established, the technical novelty of Traceable Image Conversion (TIC) lies in their synergistic integration into an informative, artifact-free 1D-to-2D interface tailored for TSAD. Specifically, TIC operates through four core mechanisms:

1. **Semantic Channel Mapping:** Allocating the raw series, trend, and residual components to separate RGB channels jointly exposes low-frequency structural deviations and high-frequency residual spikes to the visual backbone.
2. **High-Density Context Aggregation:** Condensing a 5000-step time series into 196 visual tokens aggregates long-range context while exploiting pre-trained vision models, accelerating inference $\sim 100\times$ compared to plot-rendering baselines (e.g., 0.03s/iter vs. ViT4TS at 3.14s/iter).
3. **Dimension-Aware Scaling:** Decoupling scaling along the two axes respects their distinct physical semantics: temporal interpolation ensures continuity along the time axis, while copy-padding along the period axis prevents visual boundary artifacts.
4. **Traceable Patch Alignment:** Constraining the folding width exactly to the ViT patch size (16) guarantees that each visual token encapsulates strictly continuous 1D timestamps, enabling precise 2D-to-1D feature alignment.

Table 15: Performance comparison of different multimodal fusion strategies (Affiliation-F1 / VUS-PR (%)). Best result is in red, second-best is in blue.

Strategy	IOPS	MGAB	NAB	NEK	Power	SED	Stock	TODS	UCR	WSD	YAHOO
Direct feature-level fusion	**87.40**/**22.67**	**67.51**/**0.73**	**83.31**/26.99	79.07/50.03	83.62/21.34	95.58/**79.05**	68.18/**74.38**	**86.30**/**93.14**	84.06/**22.48**	**89.53**/**22.76**	**94.97**/80.33
Score-level fusion	86.78/20.35	67.09/0.71	81.91/**27.04**	**81.09**/**60.98**	**83.96**/**21.36**	93.87/77.53	**68.33**/74.33	86.04/92.04	**84.33**/22.32	89.51/22.43	94.22/**82.34**
Decision-level fusion	86.24/16.07	**67.93**/0.70	80.89/25.50	79.50/48.66	83.04/21.16	**95.79**/77.32	66.93/71.92	86.12/92.57	82.27/18.38	89.44/18.20	93.47/79.43
Alignment feature-level fusion	**88.40**/**25.75**	67.36/**0.75**	**82.93**/**27.75**	**79.54**/**50.63**	**85.63**/**21.41**	**96.36**/**84.84**	**68.65**/**75.84**	**86.53**/**93.55**	**84.88**/**23.37**	**90.56**/**24.70**	**95.19**/**83.82**

D.3 Comparison of multimodal fusion strategies

To justify feature-level temporal alignment, we evaluated late-fusion baselines as well as direct feature fusion strategies, with detailed performance comparison reported in Table 15. Our Patch-level Alignment outperforms them by resolving two critical limitations:

1. **Limitations of Late Fusion:** Late fusion restricts the model to local optima and cannot leverage early cross-modal interaction. Consequently, visual global context cannot guide temporal features, and precise temporal features cannot rectify visual localization errors, inevitably leading to error accumulation.
2. **Limitations of Direct Feature Fusion:** Visual patches inherently lack explicit temporal indices. Naively concatenating multimodal features introduces severe spatio-temporal noise. Our alignment anchors visual tokens to corresponding temporal tokens, which is crucial for point-wise TSAD localization.

Furthermore, our superiority stems from structurally indispensable cross-modal interactions that resolve late-fusion's architectural bottleneck.

D.4 Ablation study on training objectives

To analyze how different multi-task optimization objectives interact during joint training, we systematically evaluate combinations of Binary Cross-Entropy ($\mathcal{L}_{BCE}$), Mean Squared Error ($\mathcal{L}_{MSE}$), Anomaly Window Contrastive Loss ($\mathcal{L}_{aw}$), and Routing Entropy Regularization ($\mathcal{L}_e$). The ablation study in Table 16 illustrates these objective interactions. $\mathcal{L}_{aw}$ (anomaly contrastive) generally works cooperatively with $\mathcal{L}_{BCE}$ (anomaly supervision) increasing separation between normal and anomalous features. Conversely, the temporal fidelity regularized by $\mathcal{L}_{MSE}$ conflicts with the sharp anomaly boundaries required by $\mathcal{L}_{BCE}$. Crucially, $\mathcal{L}_e$ helps reconcile these objectives via Task-Adaptive Routing to achieve peak performance.

D.5 Detailed Ablation Study

To assess cross-domain robustness without dataset-specific tuning and evaluate the individual necessity of each core module, we extend the incremental ablation study (variants V1-V7) across all 11 benchmark datasets. Each variant is pretrained on the synthetic corpus and evaluated without target-domain retraining or fine-tuning. The quantitative results are summarized in Table 17. As observed, the removal of any key module leads to a discernible performance drop, confirming that all components are integral to achieving optimal detection capabilities.

Traceable Image Conversion (TIC): Removing TIC (e.g., variants V6 and V7) causes the most severe performance collapse, with VUS-PR plummeting from 33.85% to 20.20% (V7) on WSD and

Table 16: Ablation study on training objectives (Affiliation-F1 / VUS-PR (%)). Base: $\mathcal{L}_{BCE}$. Best result is in red, second-best is in blue.

Variant	$\mathcal{L}_{MSE}$	$\mathcal{L}_{aw}$	$\mathcal{L}_e$	NAB	WSD	YAHOO
Base				82.93 / 27.75	90.56 / 24.70	95.19 / 83.82
Only $\mathcal{L}_{MSE}$	✓			82.05 / 27.64	90.10 / 24.86	95.20 / 83.29
Only $\mathcal{L}_{aw}$		✓		83.99 / 29.52	90.11 / 26.30	96.86 / 85.31
w/o $\mathcal{L}_e$	✓	✓		84.62 / 28.81	90.72 / 27.47	96.64 / 85.56
w/o $\mathcal{L}_{MSE}$		✓	✓	**86.96** / 28.44	90.87 / 28.63	96.88 / 86.58
w/o $\mathcal{L}_{aw}$	✓		✓	86.84 / **31.23**	**91.63** / **29.00**	**97.11** / **87.26**
Full Model	✓	✓	✓	**88.57** / **32.98**	**94.31** / **33.85**	**97.15** / **88.61**

Table 17: Ablation on all 11 datasets (Affiliation-F1 / VUS-PR (%)). Best results are in red, and second-best are in blue. Base: TS Encoder + Classifier, TIC: Traceable Image Conversion, PTA: Patch-level Temporal Alignment, AWCL: Anomaly Window Contrastive Learning, TMF: Task-adaptive Multi-modal Fusion.

Variant	TIC	PTA	AWCL	TMF	IOPS	MGAB	NAB	NEK	Power	SED	Stock	TODS	UCR	WSD	YAHOO
Base					83.28/20.23	**70.69**/**1.05**	82.48/24.32	**79.73**/27.88	85.51/21.25	96.87/80.75	**71.84**/**77.28**	86.37/93.46	84.63/23.09	90.33/21.77	96.65/84.41
V1	✓				87.40/22.67	67.51/0.73	83.31/26.99	79.07/50.03	83.62/21.34	95.58/79.05	68.18/74.38	86.30/93.14	84.06/22.48	89.53/22.76	94.97/80.33
V2	✓	✓			88.40/25.75	67.36/0.75	82.93/27.75	79.54/50.63	**85.63**/**21.41**	96.36/84.84	68.65/75.84	**86.53**/93.55	**84.88**/23.37	90.56/24.70	95.19/83.82
V3	✓	✓	✓		90.53/26.15	68.23/0.77	83.99/29.52	**79.74**/53.10	**85.78**/**21.48**	**97.16**/86.94	**70.89**/**76.58**	**86.73**/93.76	**85.06**/23.99	90.11/26.30	96.86/85.31
V4	✓	✓		✓	**90.84**/**28.15**	**68.73**/**0.78**	**86.84**/**31.23**	79.32/**55.38**	79.08/11.54	96.79/**87.39**	69.07/75.68	85.60/**93.90**	84.46/**24.83**	**91.63**/**29.00**	**97.11**/**87.26**
V5	✓			✓	90.56/26.26	67.22/0.75	83.23/24.81	78.85/48.04	78.63/11.33	95.78/85.37	69.60/75.60	86.05/93.20	84.75/24.40	90.54/24.91	96.06/82.62
V6		✓		✓	90.29/24.56	67.20/0.73	82.89/25.64	77.50/46.10	78.62/11.18	94.68/82.29	68.85/74.36	84.96/93.24	84.26/22.59	88.63/22.32	94.84/81.95
V7			✓	✓	90.53/24.86	66.24/0.70	81.84/24.86	75.07/45.05	77.32/10.91	94.13/77.84	67.51/74.22	85.48/93.54	83.16/22.33	88.34/20.20	94.12/77.68
Full	✓	✓	✓	✓	**90.53**/**30.79**	68.03/0.70	**88.57**/**32.98**	79.56/**57.54**	78.78/11.78	**97.31**/**89.05**	69.60/76.08	85.85/**94.20**	**85.06**/**24.99**	**94.31**/**33.85**	**97.15**/**88.61**

from 57.54% to 45.05% (V7) on NEK. This result supports the conclusion that the proposed image conversion lays a solid foundation for subsequent alignment and fusion. Furthermore, without this visual-temporal alignment foundation, multimodal interaction fails, which in most cases degrades performance below the unimodal baseline (e.g., on datasets like YAHOO and UCR).

Patch-level Temporal Alignment (PTA): Omitting PTA (e.g., variant V5) restricts fusion to coarse semantic interactions. The resulting misalignment and loss of fine-grained local correspondences lead to a noticeable decline in detection capability, evidenced by the VUS-PR dropping to 26.26% on the IOPS dataset compared to the full model's 30.79%.

Anomaly Window Contrastive Learning (AWCL): Although constructing effective contrastive pairs is challenging on densely anomalous datasets, AWCL helps the model distinguish normal from anomalous patterns more sharply. Specifically, AWCL is essential for capturing both fine-grained local variations (e.g., pushing NAB VUS-PR from 31.23% in V4 to 32.98% in the full model) and global contextual disparities in structural anomalies (e.g., elevating SED from 87.39% to 89.05%). This multi-scale discriminability confirms the highly complementary nature of our intra- and inter-window contrastive losses.

Task-adaptive Multi-modal Fusion (TMF): Eliminating TMF (e.g., variant V3, which relies on simple concatenation) causes consistent performance regression across benchmarks, such as a drop in VUS-PR from 94.20% to 93.76% on TODS and from 88.61% to 85.31% on YAHOO. This validates that the task-aware gating mechanism dynamically regulates feature integration, ensuring robust performance by balancing the conflicting demands of reconstruction and classification.

Building on this component-wise validation, we also observe performance degradations on specific datasets (MGAB, Stock, Power). Analyzing the variants reveals failure modes tied to intrinsic dataset properties:

- **Non-Periodic Signals (MGAB, Stock):** These datasets feature deterministic chaos (MGAB) or stochastic random walks (Stock) without meaningful periodicity. Forcing them into a 2D grid generates visual noise. Consequently, performance drops immediately upon introducing Traceable Image Conversion (**V1**), and downstream fusion cannot recover it.
- **Periodic Spike Anomalies (Power):** In the Power dataset, normal sequences lack spikes, whereas anomalies manifest as periodic spike noise. Because our visual conversion (TIC) explicitly constructs 2D representations based on periodicity, it inadvertently encodes these periodic anomalies as normal background structures. While the temporal modality correctly identifies them (making **V3** the best performer), the zero-shot fusion router (**V4**) incorrectly assigns high weight to this deceived visual representation, triggering massive negative transfer.

D.6 Ablation study on each component

Beyond high-level modular ablations, we conduct a micro-level sub-component ablation to isolate the granular impact of internal mechanisms within PTA (adaptive folding, positional encoding, attention), AWCL (intra- vs. inter-window contrast), and TMF (fusion operations and auxiliary reconstruction head) on the NAB, WSD, and YAHOO datasets. Table 18 summarizes these component-level empirical findings.

Table 18: Ablation study grouped by model components: PTA (Patch-level Temporal Alignment), AWCL (Anomaly Window Contrastive Learning), and TMF (Task-adaptive Multi-modal Fusion). Metrics are reported as Affiliation-F1 / VUS-PR (%). Best result is in red, second-best is in blue.

Model	Component	NAB	WSD	YAHOO
PTA	w/o adaptive folding	**87.81** / 29.99	91.49 / 30.41	97.10 / 86.70
	w/o positional encoding	**88.16** / 30.27	92.29 / **31.18**	**97.25** / 86.81
	w/o attention	85.69 / 29.88	91.57 / 29.46	96.92 / 85.53
AWCL	w/o inter-window contrast	86.69 / **31.17**	**92.85** / 30.54	97.15 / **87.29**
	w/o intra-window contrast	86.15 / **30.86**	**92.93** / **30.57**	**97.77** / **88.29**
TMF	Concatenation	85.07 / 28.54	89.80 / 26.39	95.69 / 85.72
	Addition	83.99 / 29.52	90.11 / 26.30	96.86 / 85.31
	w/o reconstruction head	86.96 / 28.44	90.87 / 28.63	96.88 / 86.58

Effectiveness of patch-level temporal alignment (PTA). We investigated the contributions of adaptive folding, positional encoding, and the attention mechanism within the PTA module. As observed, removing the **attention mechanism** resulted in the most significant performance degradation across all datasets (e.g., Affiliation-F1 dropped to 85.69% on NAB). This indicates that the attention mechanism is pivotal for capturing long-range dependencies and accurately aligning visual patches with temporal contexts. Furthermore, the exclusion of **adaptive folding** also led to a notable decline, validating its role in preserving intrinsic temporal structures during feature transformation. These results confirm that the complete PTA design is essential for ensuring fine-grained semantic correspondence.

Effectiveness of anomaly window contrastive learning (AWCL). The ablation results for AWCL demonstrate the necessity of combined contrastive strategies, though their individual impacts vary by dataset. The removal of **intra-window contrast** causes a severe drop in detection accuracy on datasets with fine-grained local anomalies like NAB. Conversely, the absence of **inter-window contrast** is more detrimental on datasets like WSD, highlighting its importance in distinguishing broader contextual anomalies globally. Interestingly, on densely anomalous datasets like YAHOO, removing intra-window contrast can even marginally improve performance. Overall, the synergy of both strategies is crucial for maximizing the discriminability of diverse anomaly patterns across different scenarios.

Effectiveness of task-adaptive multi-modal fusion (TMF). In the TMF module, we compared our proposed adaptive fusion against naive strategies (Concatenation and Addition) and examined the role of the reconstruction head. The results reveal that simple **Concatenation** and **Addition** yielded the poorest performance, proving that simple linear combinations fail to capture complex cross-modal interactions. Additionally, removing the **reconstruction head** led to a consistent performance drop. This empirical evidence supports the conclusion that sequence reconstruction serves as a critical auxiliary constraint, promoting deep feature interaction and enhancing the robustness of fused representations for the primary anomaly classification task.

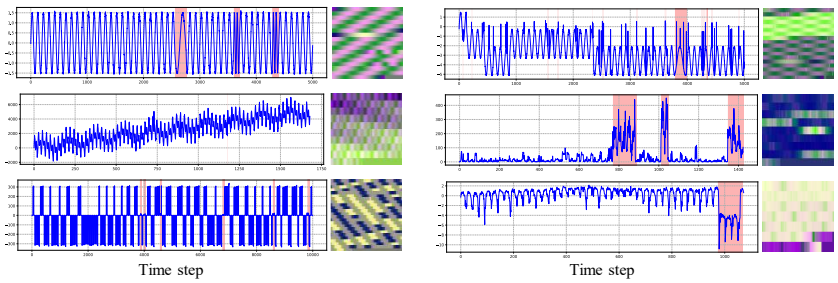


Figure 12: Initial time series and mapping images.

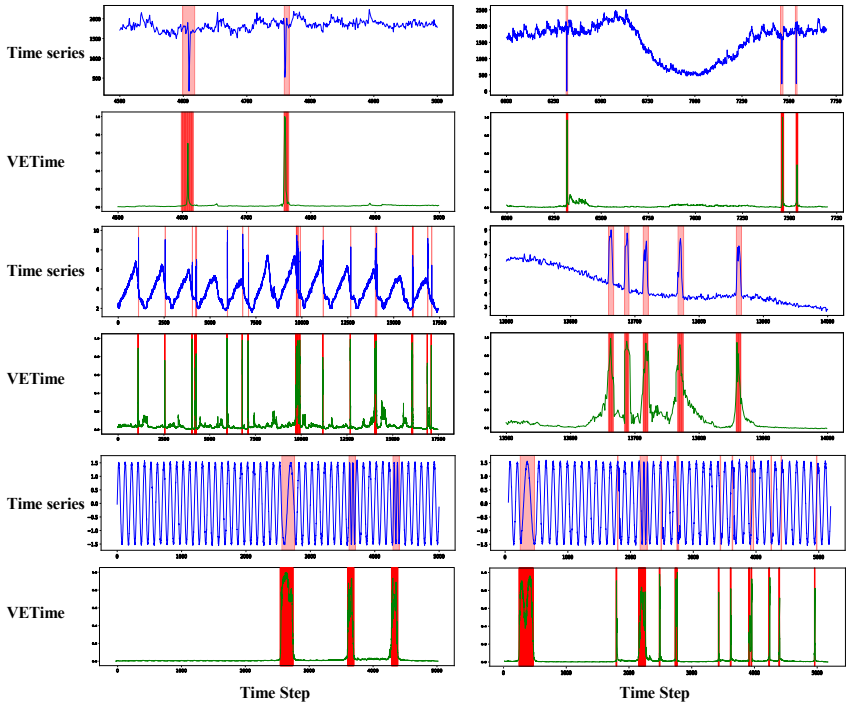


Figure 13: Qualitative visualization of anomaly detection results. Blue curve: original time series; red markers: ground-truth anomaly labels; green curve: predicted anomaly scores.

E Visualized results

E.1 Visualized mappings

Figure 12 visualizes the representations generated by our Traceable Image Conversion module across diverse data samples. The left panels display the raw 1D time-series sequences with ground-truth anomalies highlighted in red, while the right panels present their corresponding 2D visual transformations. As observed, the generated images effectively encode key temporal characteristics, such as periodicity, trends, and noise, into distinct visual textures and patterns. This visualization confirms that our conversion method successfully preserves information-dense global contexts, providing a discriminative basis for the subsequent multi-modal fusion.

E.2 Visualized anomaly scores

Figure 13 presents qualitative anomaly detection results across heterogeneous time series patterns, validating the method's robustness to diverse anomaly patterns. For point anomalies—such as abrupt spikes or isolated dips (visible in Rows 1–2), the green anomaly score curve exhibits sharp, well-localized peaks that precisely align with annotated labels. For context anomalies characterized by extended deviations like sustained trend shifts or contextual drifts (Row 3), the anomaly scores maintain elevated levels throughout the entire anomalous interval, maintaining continuous and coherent alignment with the red-labeled events.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims made in the abstract and introduction, including the proposed VETime framework and its superior zero-shot performance, are well supported by the theoretical explanations in Section 3 and the extensive experimental results in Section 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We explicitly discuss the limitations of our work in the Limitations.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: This paper focuses on proposing a novel deep learning architecture and empirical evaluations, and does not include formal theoretical proofs or theorems.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide comprehensive details regarding datasets, baselines, evaluation metrics, and implementation hyperparameters (e.g., learning rate, batch size, LoRA settings) in Appendix B and Appendix C to ensure reproducibility.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The TSB-AD datasets are publicly available, and the VETime implementation, trained model, and reproduction instructions will be made publicly available with the final release.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Detailed experimental settings, including the optimizer, learning rate, batch size, loss coefficients, and network architecture specifics, are well documented in Appendix B.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Appendix C.2 reports mean $\pm$ standard deviation over five independent runs and sequence-level bootstrap 95% confidence intervals, together with Friedman tests and Holm-corrected Wilcoxon signed-rank tests.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix B reports the core training hardware (one NVIDIA H20 24 GB GPU), model memory consumption, and inference runtime comparisons needed to reproduce and assess the computational requirements of the experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our research focuses on general time-series anomaly detection algorithms and conforms to the NeurIPS Code of Ethics without violating any ethical guidelines.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: VETime may improve fault detection in industrial operations and medical monitoring. However, false positives or missed anomalies caused by distribution shift or training-data bias could trigger unnecessary intervention or delayed responses, so high-stakes deployments should retain human oversight and domain-specific validation.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The proposed time-series anomaly detection model and the benchmarks used do not pose a high risk for misuse (unlike generative LLMs or deepfakes), so no special safeguards are necessary.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We cite the creators of TSB-AD and the pre-trained models used in our experiments, respect their applicable licenses and terms of use, and will document the corresponding license information in the public code repository.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The paper proposes a new method but does not release a new dataset, trained-model checkpoint, or standalone asset package; experiments rely on existing public benchmarks.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: This algorithmic research does not involve any crowdsourcing or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: Our research does not involve human subjects, making IRB approval inapplicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: Large Language Models (LLMs) are not utilized as a component in our core methodology.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.