No Need For Real Anomaly: MLLM Empowered Zero-Shot Video Anomaly Detection

Supplementary Material

Contents

Abstract: This appendix provides additional discussions (Appendix A), implementation details (Appendix B), runtime complexity (Appendix C), additional experiments (Appendix D), additional results (Appendix E) and future works (Appendix F).

A. Additional Discussions

A.1. Motivation

Traditional VAD models operate within fixed scenarios and closed-set anomaly taxonomies. In contrast, real-world VAD applications demand systems that can detect previously unseen anomalies, avoid predefined taxonomies, and continually acquire new knowledge without interruption. This open-world capability is essential for intelligent video systems in surveillance, autonomous driving, industrial inspection, and safety-critical monitoring. More fundamentally, open-world VAD forms a key component of world models, enabling them to perceive, interpret, and adapt to unexpected events and environmental shifts.

A critical challenge is the extreme scarcity of video anomaly data, stemming from the low occurrence rate of anomalies and privacy or security constraints that prevent data disclosure. As shown in Tab. 1, existing public VAD datasets predominantly consist of simulated campus scenarios or movie clips, which diverge significantly from real-world conditions and contain only fixed anomaly types, making them inadequate for training truly open-world models. By contrast, video segmentation data collection faces no such constraints, and the sources are readily accessible as they come from publicly available videos.

Dataset	Scenario
Subway	Subway Surveillance
UCSD Ped	Campus
CUHK Avenue	Campus
ShanghaiTech Campus	Campus
UCF-Crime	Crime Surveillance
Street Scene	Street Surveillance
XD-Violence	Online Videos & Movies
UBNormal	Synthetic data
NWPU Campus	Campus

Tab. 1. **Scenarios of existing VAD datasets.**

Our motivation is to transcend these closed-set limitations and develop a VAD model capable of open-world deployment.

A.2. Comparison with LLM-based Video Segmentation Models

LLM-based video segmentation models [2, 16, 18] produce pixel-level predictions conditioned on text prompts. In segmentation tasks, targets are precisely described, with the objective being to localize targets and delineate their boundaries. In contrast, VAD tasks exhibit infrequent anomalous events that occupy minimal frame areas, with variable target types and quantities. The primary objective in VAD shifts from localization to anomaly detection. Consequently, directly applying segmentation methods to VAD tasks yields poor performance.

As demonstrated in Fig. 1, LLM-based video segmentation models exhibit poor performance on VAD tasks. Despite training on the subset of their training data without any VAD-specific samples, our model significantly outperforms

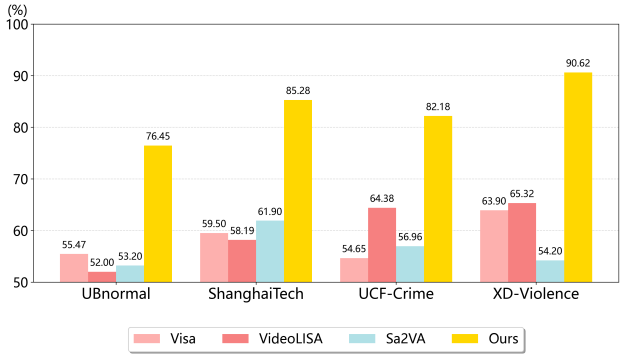

Fig. 1. **Evaluation of LLM-based video segmentation models on VAD tasks**. We assess the performance of Visa, VideoLISA, and Sa2VA across multiple VAD benchmarks. AUC is employed as the evaluation metric for UBnormal, ShanghaiTech, and UCF-Crime, and AP is utilized for the XD-Violence.

these methods, demonstrating the effectiveness of our training strategy for VAD tasks.

A.3. Comparison with LLM-based VAD Models

While training-free VAD methods offer strong generalization capabilities via MLLM, they commonly extract anomaly scores from the LLM's frame-wise or clip-wise text outputs. This approach introduces two critical limitations that hinder their practical applicability:

- **Prohibitive Time Complexity:** Consider an LLM with hidden dimension d. For an image containing N patches, the time complexity of a single forward pass is $O(N^2d + Nd^2) \approx O(N^2d)$. For training-free methods, predicting frame-level or clip-level outputs requires iterative forward passes. Given a video with T frames, where predicting a single score requires M iterative forward passes, the total time complexity becomes $O(\cdot \sum_{i=1}^{M}(TN+i)^2 \cdot d) \approx O(M^3T^2N^2d)$. With the use of KV cache, the runtime complexity can be reduced to $O(MT^2N^2d)$, which is also prohibitive for practical applications. In contrast, our model requires only one forward pass per video segment for pixel-level predictions, yielding a total complexity of $O(T^2N^2d)$. Detailed runtime complexity analyses are presented in Sec. C.
- **Coarse-Grained Anomaly Localization:** For VAD tasks, precise anomaly localization is crucial. Traditional VAD methods typically localize anomalies using local reconstruction errors, text-image similarity, or activation values. However, other LLM-based VAD methods extract anomaly scores from frame-wise or clip-wise text outputs, yielding only temporal scores without spatial localization. Our model directly predicts pixel-level outputs, enabling precise anomaly boundary delineation.

B. Implementation Details

B.1. Datasets

Our training dataset comprises of several semantic segmentation datasets, encompassing: 1) image-based datasets: ADE20K [19], Mapillary [11], CityScape [3], refCLEF, refCOCO, refCOCO+ [6], and refCOCOg [10]; 2) video-based datasets: YouTube-VOS [17], Refer-YouTube-VOS [12], MeViS [4], and Refer-DAVIS-17 [7]. The evaluation dataset includes UBNormal[1], ShanghaiTech Campus [8], XD-Violence [15], UCF-Crime [13], and UCSD Ped2 [14].

B.2. Training Pipeline

The training process adopts an end-to-end pipeline. We freeze the parameters of both the CLIP text encoder and vision backbone to preserve their pretrained representations. The MLLM is finetuned using LoRA for efficient parameter updates, while the multi-scale semantic projector and SAM2 mask decoder undergo full parameter training.

B.3. Training and Inference Setting

We train our model on a single NVIDIA 80G A800 GPU with training code based on Accelerate [5]. The AdamW [9] optimizer is employed with both learning rate and weight decay set to 1e-5. We adopt WarmupDecayLR as the learning rate scheduler with warmup iterations set to 100. The loss weight λ_{txt}, λ_{seg} are both set to 1. The batch size is set to 2. For video data, we set each video clip to contain 8 frames. For large-scale datasets such as XD-Violence and UCF-Crime, we sample every 32 frames and use the score of the sampled frame as the anomaly score for all frames within the corresponding sampling interval. The training procedure involved 10 epochs, with each epoch comprising 2,000 batches, requiring approximately 8 hours of training time in total. The total number of model parameters is 8.5B.

B.4. Evaluation Metrics

We adopt standard evaluation metrics commonly used in video anomaly detection. Specifically, for frame-level analysis, we employ frame-level Area Under the Curve (AUC) as the evaluation metric on UBNormal, ShanghaiTech, and UCF-Crime datasets, whereas Average Precision (AP) is utilized for the XD-Violence dataset. For pixel-level analysis, we use pixel-level AUC as the evaluation metric on UCSD Ped2 dataset.

B.5. Model Details

Anomaly Exposure Sampler: The maximum sampling number for anomaly categories $\max(K_E)$ is set to 30, and the anomaly probability p is set to 0.5. The examples of prompts for the sampled pseudo-anomalies and normal samples are shown in Fig. 2.

Label: 1

Prompt:
Q: Find abnormal events in this video. Abnormal events include dog sitting on the ground near a human, shifting sea, the vehicle moving in the right direction, horse standing, person drive, a boy is squeezing his fists , panda climbing branch, the duck that was initially at the back, man sitting shaking head shaking arms, a human sorting dishes.
A: Sure, it is <SEG>.

Label: 1

Prompt:
Q: Detect anomalies in this video. Target types: a dark white monkey is sitting on a black rock, a cat playing with a toy on a chair, a person in the back of a pack of bikers in black shorts with white writing, a giant panda at the top of a tree, a person fell down from his bike and stand and ride again, a red train on the tracks, a boat with many people in yellow jackets.
A: <SEG>.

Label: 1

Prompt:
Q: Monitor anomalies in this video. Anomalies include left girl, fridge door, toddler in front, person in front with blue clothes, dude on the right, brown chair, woman far left, brown shirt guy, brown boat fourth from bottom, cow, orange.
A: It's <SEG>.

Label: 0

Prompt:
Q: Please find the anomalies in this video. Abnormal cases contain a man with a black backpack, a man snowboarding wearing a black jacket, a black rope, a person capturing video while walking, a man on the left wearing a white tshirt, a man performing a windmill move.
A: The result is <SEG>.

Fig. 2. **Prompt for pseudo anomalies sampled by the anomaly exposure sampler.** A label of 1 denotes an anomalous sample, and a label of 0 denotes a normal sample. Pseudo-anomalous categories are shown in red and correspond directly to the video content, while normal categories are shown in green and are unrelated to the video content.

Anomaly Semantics Extraction: To capture high-level semantic feature of video anomalies, we utilize the Qwen2-VL-7B model as the semantic feature extractor. The model outputs semantic feature representations $f_{sem} \in \mathbb{R}^{3584}$. For parameter-efficient adaptation, we employ LoRA fine-tuning with rank $r = 8$ and scaling factor $\alpha = 16$.

Anomaly Categories Feature Encoding: We employ the text encoder of CLIP-ViT-Base-Patch32 to extract semantic features for anomaly categories c_i, yielding feature representations $f_c \in \mathbb{R}^{K \times D_t}$, where K is the number of categories and $D_t = 512$ denotes the text embedding dimension. The parameters of the text encoder are kept frozen during training.

Visual Feature Encoding: In the feature encoding stage, visual inputs x_i are encoded into representations $f_v \in \mathbb{R}^{T \times N_p \times D_v}$. The visual encoder, initialized from SAM2's Hiera-based vision encoder, generates multi-scale features with dimensions $D_v \in \{32, 64, 256\}$ across different hierarchical layers. We utilize the deepest layer features with $D_v = 256$ as input to the Multi-Scale Semantic Projector. The parameters of the visual encoder are kept frozen during training.

Multi-Scale Semantic Projector: The Multi-Scale Semantic Projector adopts a two-way transformer architecture. We employ a 5-layer projector to progressively refine the multi-scale semantic features. In our implementation, the attention hidden dimension is set to $d_a = 768$, the output feature dimension is $D_m = 256$, and the number of learnable queries is configured as 48.

Multi-level Mask Decoder: The Multi-level Mask Decoder is initialized from SAM2's prompt encoder and mask decoder. We incorporate an additional embedding layer into the prompt encoder to project f_{proj} into the feature space of the mask decoder. The decoder outputs both pixel-level and frame-level anomaly predictions.

C. Runtime Complexity

As shown in Tab. 2, our model exhibits substantially lower time complexity compared with other LLM-based zero-shot VAD approaches. Consequently, it achieves a significantly

Methods	Time Complexity	FPS	Processing Time Per-frame (ms)			
			Visual Embedding	LLM Forward	Mask Decoding	Others
LAVAD	$O(MT^2N^2d)$	1.02	-	-	-	-
AnyAnomaly	$O(MT^2N^2d)$	2.67	-	-	-	-
Ours	$O(T^2N^2d)$	7.33	58.65	26.96	44.08	4.07

Tab. 2. **Runtime complexity analysis.** T, N, d denote the number of video frames, the number of patches per frame, and the hidden dimension respectively. M represents the number of LLM forward iterations in a single response.

Train Datasets					Performance			
Image	VOS	RefVOS	MeViS	RefDAVIS	ShanghaiTech (AUC)	UCF-Crime (AUC)	XD-Violence (AP)	UBnormal (AUC)
×	✓	✓	✓	✓	83.34	79.00	88.15	74.63
✓	×	×	×	×	61.58	54.57	45.75	52.84
✓	✓	✓	✓	×	80.21	73.31	86.08	70.17
✓	✓	✓	×	✓	74.38	70.96	83.80	68.65
✓	✓	×	✓	✓	75.70	71.62	85.11	71.98
✓	×	✓	✓	✓	81.06	77.25	88.65	73.44

Tab. 3. **Ablation studies on the training sets selection.** The datasets utilized are denoted by ✓, and those omitted are marked as ×. The evaluation metrics employed remain consistent with those described in Sec. B.4.

higher FPS than those methods.

In terms of runtime distribution, the majority of the computational cost comes from three components: visual embedding, LLM forward passes, and mask decoding. Other modules—such as token compression and multi-scale projection—incur only 4 ms, accounting for only 3.1% of the total runtime.

D. Additional Experiments

D.1. Analysis of the Visual Token Compression

D.1.1. Impact on Visual Token Embedding

To investigate the effect of visual token compression on the visual token embedding process, we visualize the embeddings of test dataset before and after token compression. The result is shown in Fig. 5. Before token compression, the token features are densely clustered and lack discriminability due to the influence of background tokens. After token compression, the background information are reduced, resulting in a more distinct separation between token features, which facilitates the model in recognizing non-background objects.

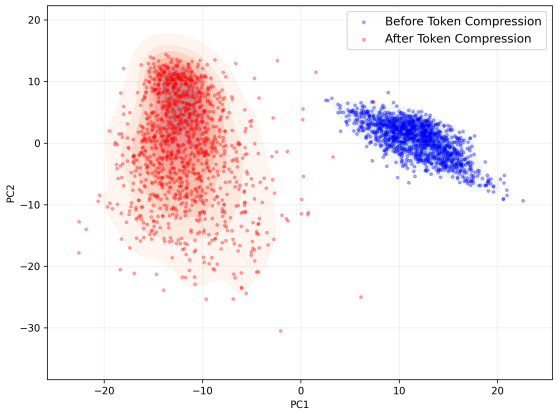

Fig. 3. **Impact of visual token compression on visual token embedding.** The points represent the coordinates of visual token features after PCA dimensionality reduction, with the kernel density estimation overlaid.

D.1.2. Impact on Runtime

The primary impact of token compression on runtime lies in its ability to reduce the LLM forward time. As shown in Tab. 4, when the token compression rate is set to 0.2, the

Compression Ratio	1	0.5	0.2	0.1
LLM Forward Time (ms)	31.71	29.33	26.96	26.78

Tab. 4. **Impact of token compression on runtime.**

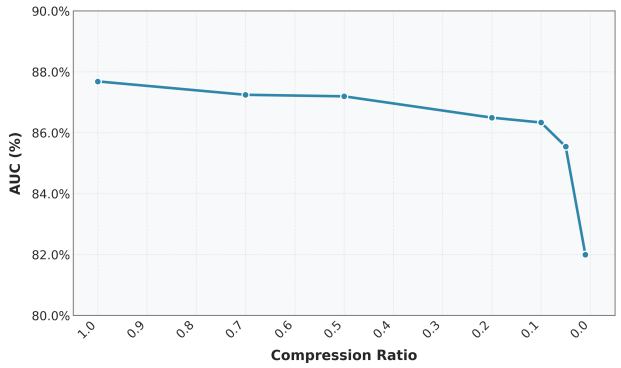

Fig. 4. **Impact of visual token compression on pixel-level prediction.** We evaluate the effect of different compression ratios on pixel-level AUC using the UCSD Ped2 dataset.

LLM forward time is reduced by 15.17%.

D.1.3. Impact on Pixel-Level Prediction

We further assess the influence of visual token compression on pixel-level prediction using the UCSD Ped2 dataset. As shown in Fig. 4, setting the compression ratio to 0.1 results in only a 1.3% drop in pixel-level AUC compared with the non-compressed baseline. This suggests that, when the compression ratio remains above 0.1, the impact of visual token compression on pixel-level performance is minimal. In contrast, when the ratio becomes smaller than 0.1, the pixel-level AUC exhibits a noticeable decline, indicating that more aggressive compression begins to substantially compromise prediction accuracy.

D.1.4. Cross-Task Analysis

To investigate the impact of visual token compression on MLLMs, we apply it to video anomaly understanding tasks. Specifically, we integrate visual token compression into Qwen2-VL and examine how it affects the model's comprehension ability, as illustrated in Fig. 7. The results show that when the compression ratio is larger than 0.1, both answering and explanation abilities remain almost unaffected. At a compression ratio of 0.1, the model occasionally produces incorrect explanation. However, when the ratio is reduced to 0.05, the MLLM frequently fails in both anomaly judgment and interpretation. These findings indicate that compression ratios above 0.1 introduce negligible degradation, whereas smaller ratios substantially impair MLLM performance. This is consistent with the effect of token compression on VAD tasks observed in the experiment section.

D.2. Analysis of the Impact of Video Resolution

In this section, we elucidate the reasons for our method's inferior performance on the UCF-Crime dataset compared to other datasets, and analysis the influence of video resolution on our model.

Dataset	**Resolution**	**Total Pixels**
UBnormal	1200×720	864k
ShanghaiTech	856×480	410k
XD-Violence	625×330	206k
UCF-Crime	320×240	76k

Tab. 5. **Video resolution of VAD datasets.** The resolution and total pixel counts are both computed as average values derived from the test sets of each dataset.

Conventional VAD methods typically utilize sequence models to capture video dynamics for identifying anomalous events; however, due to inherent scene variations, these approaches often suffer from limited generalization. Nonetheless, by incorporating temporal dependencies in the data, such methods demonstrate reduced sensitivity to input video resolution.

In contrast, the efficacy of our model in anomaly detection primarily stems from the visual semantic comprehension capabilities of the MLLM, which confers robust generalization while diminishing reliance on inter-frame temporal relationships, thereby rendering our model more susceptible to variations in video resolution. VAD datasets exhibit diverse resolutions, with average values for each dataset presented in Tab. 5. The UCF-Crime dataset features a substantially lower resolution compared to others, impeding MLLM's ability to accurately interpret the semantics of smaller anomalous targets under insufficient visual information.

To further elucidate the influence of resolution on model efficacy, we systematically varied the input video frame resolution for MLLM, with outcomes illustrated in Fig. 5. When the input resolution exceeds the original, detection performance remains largely stable, with only minor degradation. In contrast, performance declines rapidly when the input resolution falls below the dataset's original. Notably, at resolutions comparable to that of UCF-Crime (76k pixels), the model exhibits slightly inferior performance on ShanghaiTech compared to UCF-Crime, corroborating our hypothesis that inadequate resolution primarily accounts for the suboptimal results on UCF-Crime.

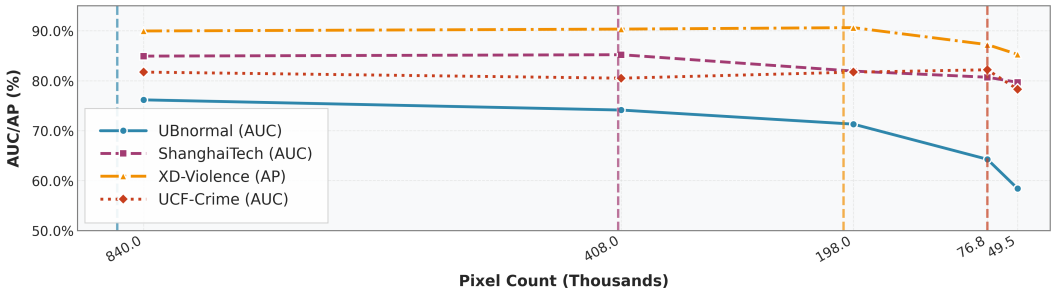

Fig. 5. **Impact of video resolution to model performance.** We evaluate the efficacy of our model across varying resolutions, with the original resolutions of each datasets denoted by dashed lines in the figure.

Layers	UBnormal AUC (%)	ShanghaiTech AUC (%)	UCF-Crime AUC (%)	XD-Violence AP (%)
1	55.86	62.85	57.11	59.24
2	62.04	79.75	78.14	89.40
3	68.47	79.65	78.03	88.28
4	71.95	82.30	79.32	90.30
5	76.30	84.46	82.06	90.51
6	77.12	83.52	80.06	87.70

Tab. 6. **Ablation study about the number of projector layers.**

D.3. Ablation Study

D.3.1. Analysis of the Train Datasets Selection

We have detailed the training datasets employed in this study in Sec. B.1. To systematically evaluate the impact of each training dataset on the model, we conducted ablation experiments. Specifically, we performed these experiments by excluding datasets from the training process, while maintaining all other configurations unchanged. Then we evaluated the performance of the resulting models post-training. The empirical results are presented in Tab. 3.

It can be observed that the video datasets play a dominant role in the training process. Since the evaluation is conducted on video data, image datasets provide only marginal performance gains, and training solely with image data yields very limited improvement for the VAD tasks. Among the video datasets, the VOS dataset contributes less compared to the other three referring datasets. The incorporation of referring data enhances the global comprehension and relational understanding between objects in MLLMs, which is also beneficial for the VAD tasks.

D.3.2. Analysis of the Multi-Scale Semantic Projector

Our proposed Multi-Scale Semantic Projector can effectively integrate video-level semantic features with frame-level anomalous information. The frame-level anomalous information is manifested through the frame-level feature $f_a \in \mathbb{R}^{T\times K\times D_a}$. We measure its anomaly perception capability using the average correlation across anomaly category dimensions, denoted as $\bar{\rho}_{ac}$. When anomalous events occur, dimensions corresponding to the anomaly category deviate significantly from others, causing $\bar{\rho}_{ac}$ to decrease.

Fig. 6 shows the temporal variation of $\bar{\rho}_{ac}$, with detected anomalous people marked in green. The results demonstrate that $\bar{\rho}_{ac}$ remains high during normal periods but drops significantly when anomaly occurs, with the decline magnitude proportional to the anomaly's spatial extent. This validates the Multi-Scale Semantic Projector's capability to perceive temporal anomalous information.

Tab. 6 illustrates the impact of projector layer depth on performance. The results show that the model's zero-shot performance initially improves with increasing layers before subsequently declining. This trend can be attributed to that while additional layers increase learnable parameters and facilitate the capture of more complex data patterns. And excessive depth leads to convergence difficulties during training, ultimately resulting in performance degradation.

D.3.3. Analysis of Video Clip Length

Tab. 7 demonstrates the impact of video frame count within a single clip on model performance. As clip length increases, the model exhibits slight performance improvements, which we attribute to the MLLM's enhanced ability to capture contextual information across frames. However,

Clip Length	UBnormal AUC (%)	ShanghaiTech AUC (%)	UCF-Crime AUC (%)	XD-Violence AP (%)	FPS	Processing Time Per-frame (ms)	
						Mask Decoding	Others
4	75.77	81.59	80.27	88.66	4.78	44.08	165.27
8	76.45	85.26	81.26	90.51	7.33	45.31	91.10
12	76.04	84.35	81.54	90.30	8.80	44.59	69.02
16	75.93	83.40	82.04	89.57	11.31	45.16	43.26

Tab. 7. **Effect of video frame count in a single video clip.**

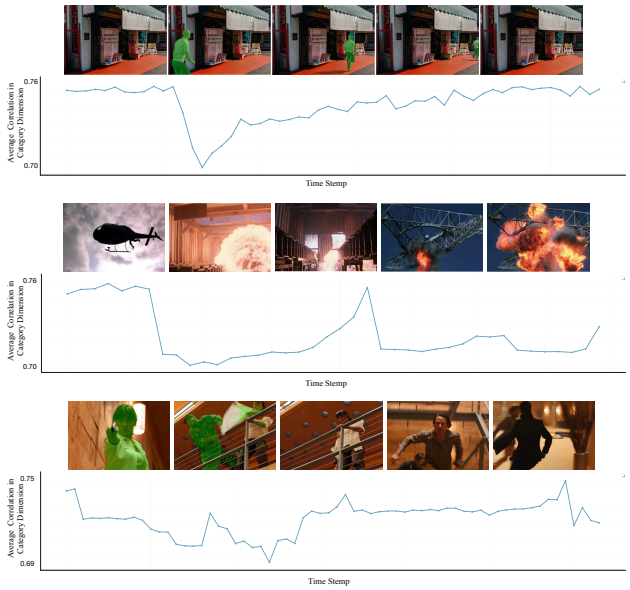

Fig. 6. **Temporal dynamics of $\bar{\rho}_{ac}$.** To facilitate the distinction between normal and abnormal people, the detected anomalous people are highlighted in green.

performance marginally decreases beyond a certain threshold, likely due to the MLLM's diminished capacity to process fine-grained local information when handling excessive input frames.

As for runtime, the MLLM performs only a single forward pass for each video clip, making this component largely insensitive to the clip length. As the clip length increases, the average per-frame cost of this part decreases, thereby improving the overall FPS. In contrast, mask decoding operates on a frame-by-frame basis, and its runtime does not diminish as the clip length grows.

E. Additional Results

Our objective is to perform video anomaly detection in open-world scenarios. Fig. 8 demonstrates the performance of the proposed method on standard VAD datasets, while Fig. 9 illustrates the detection effectiveness of our approach in open-world settings.

F. Future Works

Our method currently faces the following issues, which we plan to address in future improvements:

- **Expanding Model Interpretability:** We aim to further enhance the interpretability of the MLLM for anomaly detection. By incorporating anomaly understanding datasets and causal reasoning capabilities, we entend to enable the model to not only provide precise localization but also generate human-interpretable explanations for the underlying causes of detected anomalies.
- **Enhancing Long-Range Temporal Dependency Modeling:** Currently our model processes each video clip independently. Although it achieves strong performance, it lacks the ability to capture broader contextual information across the entire video. In future work, we aim to strengthen the model's capability for global video understanding rather than focusing solely on isolated clips.

Through these improvements, we expect to further enhance the performance of our method and broaden its applications, thereby providing more possibilities for research and practice in the field of video anomaly detection.

References

[1] Andra Acsintoae, Andrei Florescu, Mariana-Iuliana Georgescu, Tudor Mare, Paul Sumedrea, Radu Tudor Ionescu, Fahad Shahbaz Khan, and Mubarak Shah. Ubnormal: New benchmark for supervised open-set video anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20143–20153, 2022. 2

[2] Zechen Bai, Tong He, Haiyang Mei, Pichao Wang, Ziteng Gao, Joya Chen, Zheng Zhang, and Mike Zheng Shou. One token to seg them all: Language instructed reasoning segmentation in videos. *Advances in Neural Information Processing Systems*, 37:6833–6859, 2024. 1

[3] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 2

[4] Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, and Chen Change Loy. Mevis: A large-scale benchmark for

video segmentation with motion expressions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2694–2703, 2023. 2

[5] Sylvain Gugger, Lysandre Debut, Thomas Wolf, Philipp Schmid, Zachary Mueller, Sourab Mangrulkar, Marc Sun, and Benjamin Bossan. Accelerate: Training and inference at scale made simple, efficient and adaptable. https://github.com/huggingface/accelerate, 2022. 2

[6] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 787–798, 2014. 2

[7] Anna Khoreva, Anna Rohrbach, and Bernt Schiele. Video object segmentation with language referring expressions. *arxiv: 1803.08006*, 2018. 2

[8] W. Liu, D. Lian W. Luo, and S. Gao. Future frame prediction for anomaly detection – a new baseline. In *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[9] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 2

[10] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20, 2016. 2

[11] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE international conference on computer vision*, pages 4990–4999, 2017. 2

[12] Seonguk Seo, Joon-Young Lee, and Bohyung Han. Urvos: Unified referring video object segmentation network with a large-scale benchmark. In *European conference on computer vision*, pages 208–223. Springer, 2020. 2

[13] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6479–6488, 2018. 2

[14] Shu Wang and Zhenjiang Miao. Anomaly detection in crowd scene. In *IEEE 10th International Conference on Signal Processing Proceedings*, pages 1220–1223. IEEE, 2010. 2

[15] Peng Wu, Jing Liu, Yujia Shi, Yujia Sun, Fangtao Shao, Zhaoyang Wu, and Zhiwei Yang. Not only look, but also listen: Learning multimodal violence detection under weak supervision. In *European conference on computer vision*, pages 322–339. Springer, 2020. 2

[16] Cilin Yan, Haochen Wang, Shilin Yan, Xiaolong Jiang, Yao Hu, Guoliang Kang, Weidi Xie, and Efstratios Gavves. Visa: Reasoning video object segmentation via large language models. In *European Conference on Computer Vision*, pages 98–115. Springer, 2024. 1

[17] Linjie Yang, Yuchen Fan, and Ning Xu. The 2nd large-scale video object segmentation challenge-video object segmentation track, 2019. 2

[18] Haobo Yuan, Xiangtai Li, Tao Zhang, Yueyi Sun, Zilong Huang, Shilin Xu, Shunping Ji, Yunhai Tong, Lu Qi, Jiashi Feng, and Ming-Hsuan Yang. Sa2va: Marrying sam2 with llava for dense grounded understanding of images and videos. *arXiv pre-print*, 2025. 1

[19] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. 2

Fig. 7. **Impact of visual token compression on video anomaly understanding.** We assess the accuracy of the MLLM's anomaly judgments and explanations by progressively decreasing the compression ratio. Correct assessments are denoted by ✓, while incorrect ones are indicated by×.

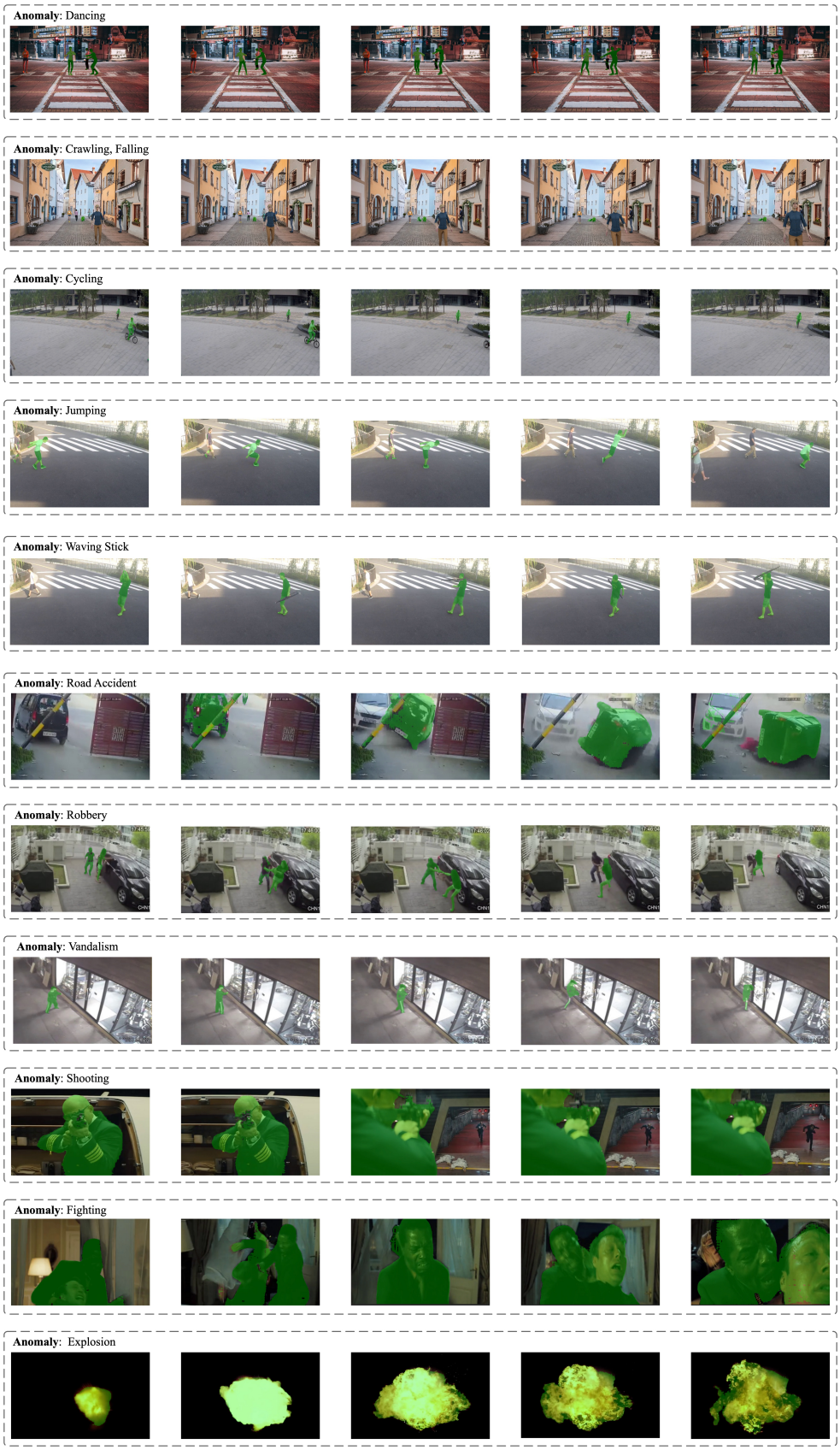

Fig. 8. **Additional results on VAD datasets.**

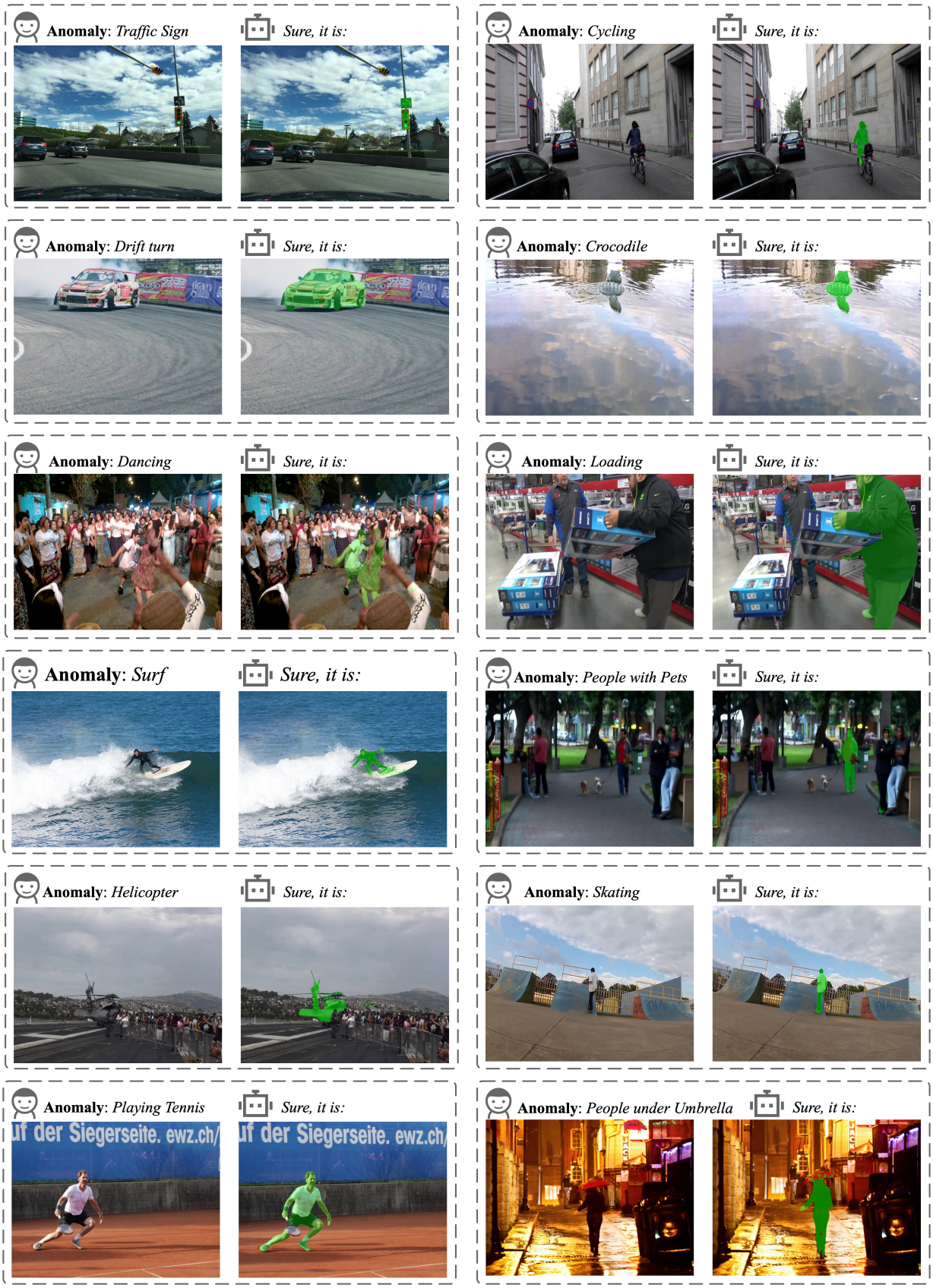

Fig. 9. **Additional results in open-world scenarios.**