Alert-CLIP: Abnormality-aware Latent-Enhanced Representation Tuning of CLIP for Video Anomaly Detection

Supplementary Material

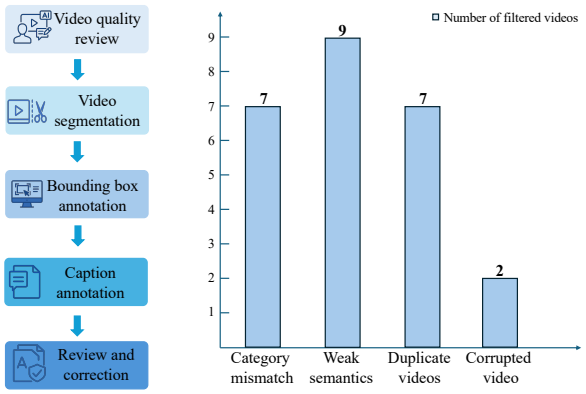
This supplementary document provides additional technical details, extended analyses, and further experimental results that complement the main paper. The contents are organized as follows:

- **Section A: Dataset Construction and Annotation Details.** This section presents the overall VAGTA construction pipeline, together with our video filtering criteria, quality control procedure, and region-level caption construction, including Qwen-VL based hard-negative caption generation rules and concrete examples.
- **Section B: Additional Experimental Results.** This section reports supplementary experimental results not included in the main paper, including additional open-vocabulary anomaly detection results on UCF-Crime and XD-Violence, cross-domain zero-shot evaluation with BLIP, and a controlled data-vs.-framework disentanglement study.
- **Section C: Training and Implementation Details.** This section details the temporal Transformer architecture, ROI Align based region feature extraction, and a comprehensive summary of all training hyper-parameters used in the two-stage learning scheme.
- **Section D: Computational Complexity and Inference Efficiency.** We report the FLOPs, training cost, memory usage, and inference speed of Alert-CLIP under the default 256-frame setting.
- **Section E: Failure Cases and Dataset Limitations.** This section presents qualitative failure cases together with attention maps, and discusses how extreme video degradation and low-visibility anomalies limit the performance of Alert-CLIP and future directions to mitigate these issues.

A. Dataset Construction and Annotation Details

A.1 Overview of the VAGTA Construction Pipeline

Figure 1 summarizes the overall construction pipeline of VAGTA. Starting from candidate videos collected from the official splits of UCF-Crime and MSAD, we first conduct video quality review to remove low-quality abnormal clips with category mismatch, weak anomaly semantics, duplicated content, or corrupted signals. We then perform normal-video segmentation, bounding-box annotation, region- and clip-level caption annotation, and final review and correction to obtain the multi-granularity annotations used in our training framework.



(a) Data Preparation Pipeline (b) Video Filtering Statistics

Figure 1. Overview of the VAGTA construction pipeline. The left panel illustrates the annotation workflow, including video quality review, video segmentation, bounding-box annotation, caption annotation, and final review and correction. The right panel summarizes the number of filtered abnormal videos for different removal reasons. More detailed score-based quality control statistics are provided in Sec. A.2.

A.2 Video Filtering Criteria and Quality Control

To ensure semantic consistency and anomaly discriminability in the training corpus, we conduct a structured quality screening over all candidate videos from UCF-Crime and MSAD. Each video is evaluated along four dimensions: (i) category–content consistency, (ii) clarity of the abnormal event, (iii) stability of visual quality, and (iv) redundancy with existing samples. All scores range from 1 to 5, but the four criteria receive different weights (1:2:2:5) according to their importance for fine-grained alignment. The final score is computed as:

$$\text{Score} = 1 \cdot S_{\text{class}} + 2 \cdot S_{\text{clarity}} + 2 \cdot S_{\text{quality}} + 5 \cdot S_{\text{redundancy}},$$

with a maximum of 50. We set 25 as the minimum acceptable threshold. Videos below this threshold are excluded from all subsequent annotation steps.

In total, we assess 4,237 candidate videos. Figure 2 summarizes the distribution of the final scores using three bins: 0–25, 25–40, and 40–50. Only 25 clips (0.6%) fall into the 0–25 range and are removed by our filtering procedure, whereas the majority of accepted clips lie in the 40–50 range (80.5%), with the remaining 18.9% scoring between 25 and 40. This distribution confirms that most videos entering the subsequent annotation stages are of high semantic

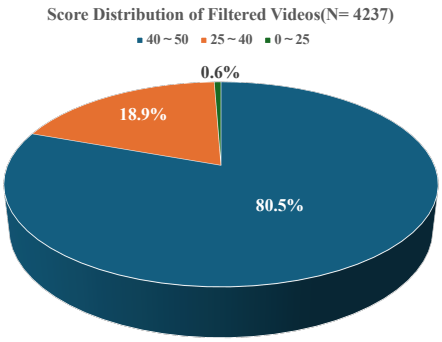


Figure 2. Score distribution of all 4,237 candidate videos under our four-criterion quality metric. The vast majority of clips fall in the high-quality 40–50 range, while only a small fraction with scores below 25 is discarded before annotation.

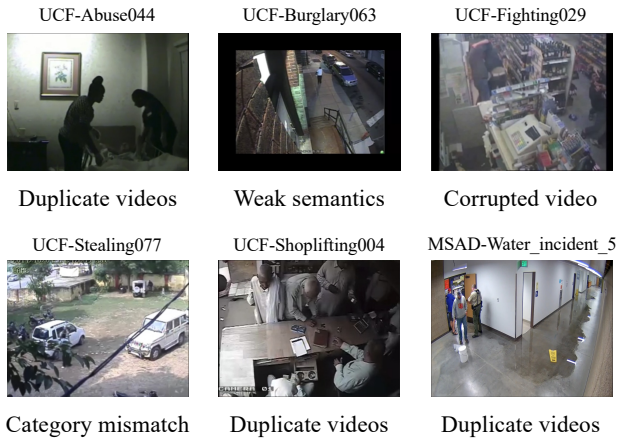


Figure 3. Representative abnormal videos removed during filtering. From left to right: duplicate videos, weak anomaly semantics, corrupted video, category mismatch, and additional duplicate videos from near-identical scenes.

and visual quality, while a small fraction of low-quality or ambiguous samples is effectively discarded.

A.3 Examples of Removed Abnormal Videos

To facilitate readers' understanding of our quality control process, we present examples of partially rejected videos in Figure 3. Typical issues include: (1) substantial mismatch between the visual content and the annotated category label; (2) heavy occlusion or extreme camera distance that renders the abnormal action barely recognizable; (3) near-duplicate footage from identical surveillance sources; and (4) compression artifacts, missing frames, or severe illumination interference. Although limited in number, retaining such clips would degrade region-level alignment and hard-negative learning, hence they are excluded from the final training set.

A.4 Hard-Negative Caption Generation Rules and Prompt Design

To support region-level alignment and hard-negative learning, we construct three adversarial captions for each region caption using Qwen-VL. These captions remain visually consistent with the original region— preserving scene layout, number of people, object arrangement, and motion style— while inverting the semantic polarity (normal ↔ abnormal). This encourages the model to discriminate visually similar yet semantically opposite scenes.

Two task-specific prompts are designed for abnormal-to-normal and normal-to-abnormal rewriting. All generated captions must be in English, contain no more than 20 tokens (to fit CLIP's input constraints), and be returned as a JSON list of three strings. The actual prompts are:

```
SYSTEM_PROMPT = (
    "You are a vision-language assistant for
        academic video understanding."
)

PROMPT_ABNORMAL_TO_NORMAL = (
    "Task:\n"
    "Given several frames from an abnormal
        activity clip and its region-level
        caption, "
    "write 3 alternative NORMAL captions that
        stay visually similar (same people, scene
        , "
    "layout, and motion style) "
    "but describe a peaceful everyday situation
        instead.\n"
    "Rules: English only; each caption <= 20
        tokens. Output JSON list of 3 strings."
)

PROMPT_NORMAL_TO_ABNORMAL = (
    "Task:\n"
    "Given a region-level caption of a NORMAL
        daily activity clip, "
    "write 3 alternative ABNORMAL captions that
        remain visually similar (same people, "
    "scene, layout), "
    "but describe an unusual or unexpected event
        instead.\n"
    "Rules: English only; each caption <= 20
        tokens. Output JSON list of 3 strings."
)
```

As shown in Fig. 4, the generated hard negatives preserve visual plausibility but introduce strong semantic contrast. This encourages Alert-CLIP to rely on fine-grained semantic cues rather than coarse scene context when distinguishing normal from abnormal events, and significantly enhances its robustness when facing visually similar yet semantically divergent videos.

Normal Caption : " Two men attacked a woman in white and black, one pulling a red cloth from her before both fled after the second punched her."

Hard-Negative Caption:

- "A woman in white and black bends down to pick up a dropped item while another person walks by."
- "Two individuals interact near a table; one kneels to retrieve something from the floor."
- "A woman crouches beside a table, adjusting an object, as another person passes nearby."

Anomalous Caption : "A group of students are walking in the school hallway."

Hard-Negative Caption:

- "A group of students are floating in the school hallway."
- "A group of students are walking backward in the school hallway."
- "A group of students are walking through a glowing school hallway."

Figure 4. Illustration of Qwen-VL generated hard-negative captions. The upper block shows an *abnormal* region caption (red) and its visually aligned but semantically *normal* hard negatives (green). The lower block shows a *normal* caption and its corresponding *abnormal* hard negatives. Note that the scene layout, number of people, and coarse motions remain similar across each group, while the semantic polarity between normal and abnormal descriptions is deliberately flipped.

B. Additional Experimental Results

B.1 Additional Results for Open-Vocabulary VAD

In the main paper, we reported the open-vocabulary performance on UCF-Crime and XD-Violence, where only AUC was presented for XD-Violence. For completeness, we provide in Table 1 the AUC and AP results for both datasets under the *Overall*, *Base*, and *Novel* splits.

All numbers in the main paper and this supplementary material are obtained from our unified implementation rather than copied from previous works. All models are trained from scratch under identical hyper-parameters, computing environments, and random seeds to ensure fairness and reproducibility.

Across both datasets, Alert-CLIP (Full Model) consistently improves upon the CLIP baseline on all evaluation splits. The gains are especially pronounced on the *Novel* split, demonstrating that our multi-level alignment and hard-negative reconstruction substantially enhance CLIP's ability to generalize to unseen anomaly semantics.

Table 1. Supplementary open-vocabulary VAD results on UCF-Crime and XD-Violence. All numbers are percentages. Best results are in bold.

Dataset / Split	CLIP		Alert-CLIP (Full)	
	AUC	AP	AUC	AP
UCF-Crime				
Overall	85.70	28.04	**86.62**	**36.16**
Base	**94.21**	42.09	94.03	**57.81**
Novel	86.36	21.26	**86.69**	**22.35**
XD-Violence				
Overall	90.08	75.81	**91.63**	**79.43**
Base	88.95	54.01	**89.62**	**57.73**
Novel	94.33	83.69	**95.88**	**87.01**

B.2 Generality Beyond CLIP: Zero-Shot Evaluation with BLIP

To further verify that our training framework is not restricted to CLIP, we additionally apply the same training strategy to BLIP and evaluate it under the cross-domain zero-shot setting on XD-Violence. As shown in Table 2, the proposed framework consistently improves the original BLIP backbone under both frame-level and video-level evaluation. This result suggests that the benefit of our multi-level alignment strategy is not limited to CLIP-style encoders, but can also generalize to other vision–language backbones.

B.3 Disentangling the Effects of Data and Framework

To disentangle the respective contributions of curated data and the proposed training framework, we further conduct a controlled swap study on UCF-Crime under the weakly supervised setting. Specifically, we compare four combinations: (1) original data with the original framework, (2) original data with Alert-CLIP, (3) VAGTA with the original framework, and (4) VAGTA with Alert-CLIP. As shown in Table 3, the best performance is achieved only when the

Table 2. Cross-domain zero-shot results on XD-Violence using BLIP as the backbone. All numbers are percentages. Best results are in bold.

Setting	AUC	AP
Frame (BLIP-base)	47.88	21.43
Frame (Alert-BLIP)	**64.85**	**35.86**
Video (BLIP-base)	35.04	39.79
Video (Alert-BLIP)	**62.19**	**61.53**

Table 3. Disentangling the effects of data and framework on UCF-Crime under the weakly supervised setting. Best results are in bold.

Setting	AUC	AP
Original data + Original framework	88.02	32.72
Original data + Alert-CLIP	88.15	32.31
VAGTA + Original framework	87.67	32.44
VAGTA + Alert-CLIP	**89.32**	**33.57**

curated VAGTA data and the proposed Alert-CLIP framework are combined. This indicates that the final improvement does not come from better data alone, but from the synergy between high-quality multi-granularity annotations and the proposed multi-level alignment framework.

C. Training and Implementation Details

C.1 Temporal Transformer Architecture

Our framework incorporates a lightweight temporal modeling module that operates on frame-level visual embeddings extracted by the CLIP ViT-B/32 encoder. For an input sequence of $T = 256$ frames, each frame is first processed by the visual encoder to obtain a feature vector of dimension $d = 512$, producing

$$\mathbf{V} = [v_1, v_2, \ldots, v_T] \in \mathbb{R}^{T\times 512}.$$

To capture short-range temporal dependencies, we apply a Transformer encoder with two layers, where each layer follows the standard Pre-LN formulation (LayerNorm $\rightarrow$ Multi-Head Attention $\rightarrow$ FFN). The module uses 8 attention heads, a feed-forward hidden dimension of $4d$, learnable 1D absolute positional embeddings, and a dropout rate of 0.1. The Transformer operates in batch-first mode and produces temporally enriched representations

$$\mathbf{Z} = \text{TemporalTransformer}(\mathbf{V}) \in \mathbb{R}^{T\times 512}.$$

Rather than simple averaging, we adopt a learnable attention-based aggregation mechanism to identify frames that contribute more strongly to anomaly semantics. Specifically, the output sequence $\mathbf{Z}$ is passed through a two-layer MLP to compute frame importance weights, which are normalized and applied to generate the final video-level embedding:

$$z_{\text{video}} = \sum_{t=1}^{T} \alpha_t z_t,$$

where z_t is the t-th row of $\mathbf{Z}$ and α_t denotes the learned temporal attention weight.

This temporal module is integrated into the overall vision branch and is trained jointly with all alignment objectives. Gradient checkpointing is enabled to support efficient training with 256-frame inputs on a single A800 GPU.

C.2 ROI Align for Region-Level Feature Extraction

For region–text alignment, we obtain localized visual features by applying ROI Align to the patch-level embeddings produced by the CLIP ViT-B/32 visual encoder. Given an input frame resized to 224×224, the encoder yields a spatial grid of patch embeddings with resolution 7×7 and channel dimension 512. We treat this grid as a pseudo feature map

$$\mathbf{F}_t \in \mathbb{R}^{7\times7\times512}.$$

Bounding boxes provided in pixel coordinates are mapped to this 7×7 patch grid by linear scaling with respect to the input resolution. Since ROI Align supports continuous coordinates, no discretization is required. The region features for frame t are then extracted using

$$r_t = \text{ROIAlign}(\mathbf{F}_t, \text{box}_t) \in \mathbb{R}^{7\times7\times512}.$$

Each region tensor is average-pooled to obtain a 512-dimensional vector f_t^{roi}, and multiple region features within the same temporal segment are aggregated by mean pooling:

$$f^{\text{roi}} = \frac{1}{T'} \sum_{t=1}^{T'} f_t^{\text{roi}}.$$

The resulting feature is used in the region-level contrastive objective to reinforce spatially grounded alignment between anomaly-related visual regions and their corresponding textual descriptions.

C.3 Comprehensive Hyper-Parameter Summary

Table 4 presents a unified summary of all training hyperparameters used across Stage 1 (Global Alignment) and Stage 2 (Global+Region+Hard-Negative). The table consolidates batch configuration, optimizer settings, LoRA parameters, mixed precision, and reproducibility settings, reflecting the exact configuration of our released training scripts.

Table 4. **Training hyper-parameters for Stage 1 and Stage 2.** The full configuration is shown for reproducibility.

Category	Parameter	Stage 1 (Global)	Stage 2 (Global + Region + HardNeg)
Input	num_frames	256	256
Batch	batch size	8	8
	grad accumulation	16	16
Optimizer	base lr	5e-6	4e-6
	text lr	2e-6	1.5e-6
	weight decay	0.01	0.01
	warmup ratio	0.05	0.10
	grad norm	1.0	1.0
	scheduler	cosine	cosine
Training	epochs	100	20
	mixed precision	bf16 (False)	bf16 (True)
Model Freeze	freeze backbone	False	Phase2 only
	global loss	✓	✓(weighted)
	region / hard-neg losses	×	✓
LoRA	enable	✓	✓
	rank r	4	16
	alpha	8	32
	dropout	0.05	0.05
	include k	True	True
Checkpointing	save steps	50	100
	limit	10	2
Reproducibility	seed	42	42

D. Computational Complexity and Inference Efficiency

This section provides a detailed account of the computational cost of Alert-CLIP under the default configuration used in the main paper: CLIP-B/32 as the vision backbone, 256 input frames, two-stage training, and LoRA-based adaptation. All measurements are produced on a single NVIDIA A800 (80GB) GPU.

D.1 FLOPs and Model Complexity

We analyze the end-to-end forward computation using fvcore's FlopCountAnalysis, decomposing the total FLOPs into vision, temporal, and text pathways, as well as the ROI projection branch used during training. Table 5 summarizes the results.

The computation is overwhelmingly dominated by the CLIP-B/32 vision encoder, which accounts for over 99% of the total FLOPs. The temporal modeling and text pathways add only a small overhead, and the ROI projection head contributes a negligible cost during training and is not used during inference. Consequently, Alert-CLIP maintains nearly identical computational complexity to CLIP-B/32 when deployed.

Table 5. FLOPs breakdown of Alert-CLIP with CLIP-B/32. Numbers are measured for a 256-frame 224×224 input. The ROI branch is enabled only during training.

Module	FLOPs
Vision encoder (CLIP-B/32)	4.501 T
Visual projection	0.101 G
Temporal Transformer (2 layers)	1.076 G
Text encoder	10.133 G
Text projection	0.262 G
Total (inference, no ROI)	**4.512 T**
ROI projection (train only)	0.503 G

D.2 Training Time and Memory Usage

We provide detailed measurements of end-to-end training cost under the two-stage learning paradigm. Stage 1 performs global video–text alignment and is trained for 100 epochs, requiring a total of **66.85 hours**. Stage 2 incorporates region-level alignment and semantic hard-negative learning and is trained for 20 epochs, taking **19.4 hours**. All experiments employ mixed precision (bf16 in Stage 2), gradient accumulation, and gradient checkpointing to support 256-frame inputs.

Table 6. Training time and memory usage of Alert-CLIP. Measurements obtained on a single NVIDIA A800 (80GB).

Stage and epochs	Time(h)	Memory(GB)
Stage 1 (Global Alignment, 100)	66.85	68.79–71.53 GB
Stage 2 (Global+Region+Hard, 20)	19.4	76.28–79.81 GB
Total	**86.25 h**	—

Despite the long 256-frame input, memory consumption remains stable and well within the limits of a single 80GB GPU. This efficiency comes primarily from the use of LoRA adapters and gradient checkpointing, which avoid full fine-tuning of the CLIP backbone while preserving training stability.

D.3 Inference Throughput and Latency

To assess end-to-end efficiency, we benchmark Alert-CLIP using a batch size of one and a 256-frame input. The model processes each video in 213.26 ms, corresponding to a throughput of 4.69 videos per second (approximately 1200 frames per second). Table 7 summarizes the results.

Table 7. Inference efficiency of Alert-CLIP. Results measured with batch size 1 on a single NVIDIA A800 (80GB).

Frames	Latency (ms)	Videos/s	Frames/s
256	213.26	4.69	1200

Since the temporal modeling and projection modules account for less than one percent of the total FLOPs, the end-to-end inference time is effectively dominated by the CLIP-B/32 backbone. Consequently, Alert-CLIP achieves nearly identical real-time performance as vanilla CLIP while providing substantially improved anomaly-awareness.

E. Failure Cases and Dataset Limitations

While Alert-CLIP achieves strong anomaly-awareness across diverse scenes, several failure cases reveal inherent limitations rooted primarily in the quality of the underlying video data rather than deficiencies in the model architecture. Many low-performing samples originate from surveillance footage containing severe compression artifacts, heavy motion blur, extremely low resolution, or near-dark illumination. In these situations, the anomalous subjects occupy only a very small number of pixels and are often indistinguishable even to human observers. Consequently, the extracted visual features become significantly degraded, constraining the upper bound of cross-modal reasoning.

To illustrate these limitations, Figure 5 presents two representative examples. In the first case (UCF-Abuse011), the

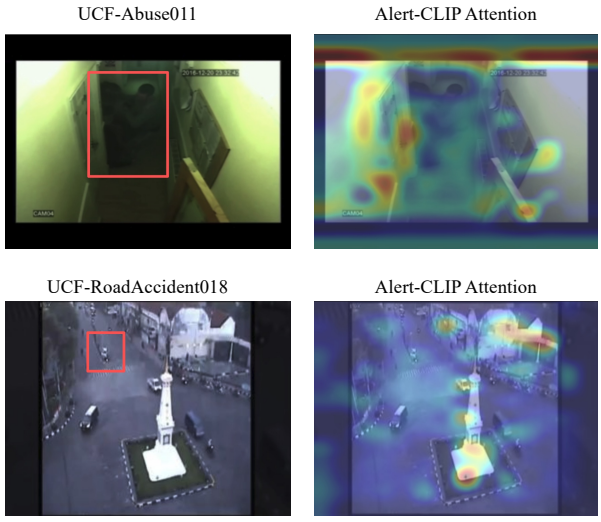


Figure 5. Typical failure cases of Alert-CLIP. Top: UCF-Abuse011 suffers from severe blur and low illumination, causing the model to spread attention across background edges rather than the true abnormal interaction. Bottom: UCF-RoadAccident018 contains a small distant vehicle that is barely distinguishable in the original frames, leading the model to focus on high-contrast static structures instead of the anomaly.

abnormal interaction occurs in a poorly illuminated, narrow corridor, with the actors heavily occluded and blurred. The model's attention map shows that Alert-CLIP struggles to isolate the interacting individuals and instead disperses attention across high-contrast edges and background structures. This leads to an underestimation of anomaly intensity. In the second example (UCF-RoadAccident018), the anomalous event involves a small vehicle entering the junction at long distance, occupying only a tiny region in the frame. Despite the model attempting to reason globally about the scene, the attention heatmap demonstrates a drift toward bright static structures such as the central monument and surrounding roads. The true anomalous agent is nearly invisible due to the low resolution and scale, resulting in unstable anomaly scoring.

These observations highlight the fundamental challenge of detecting fine-grained or low-visibility anomalies in degraded surveillance footage. Improving robustness in such extreme cases likely requires either higher-quality video sources or auxiliary modules specifically designed for fine-scale object enhancement or motion magnification.