Controllable Multi-label Video Safety Detection via Adaptive Tversky Policy Optimization

Guangyu Yang[1,2,*,†] **Jingbiao Mei**[1,†] **Mingsheng Sun**[3,†] **Jinghong Chen**[1]
Yingtong Bu[2] **Pengda Qin**[4] **Da Chen**[5,‡] **Bill Byrne**[1,‡]

[1]University of Cambridge [2]Xiaohongshu Inc. [3]AntGroup
[4]Tencent Company, China [5]University of Bath

gy266@cam.ac.uk jm2245@cam.ac.uk sunms5513@gmail.com
jc2124@cam.ac.uk buyingtong819@gmail.com derekpdqin@tencent.com
da.chen@bath.edu wjb31@cam.ac.uk

Abstract

The rapid growth of video-based social media has increased users' exposure to harmful content, creating a need for reliable automated video safety detection. Although recent Vision-Language Models (VLMs) show strong video understanding capabilities, existing harmful video detection systems face two key limitations: they typically reduce safety detection to binary classification, overlooking the inherently multi-label nature of unsafe videos, and they rely on static training objectives that do not support controllable precision-recall trade-offs, though the desired operating point may vary across moderation pipelines and unsafe categories. To address these gaps, we propose Adaptive Tversky Policy Optimization (ATPO), a reinforcement learning framework for Multi-label Video Safety Detection (Multi-VSD). ATPO introduces the Adaptive Tversky Reward (ATR), which dynamically adjusts false-positive and false-negative penalties during training to enable controllable precision–recall trade-offs. Experiments on SafeWatch-Bench and XD-Violence show that ATPO substantially improves multi-label performance, increasing the Jaccard Index from 40.66 to 75.44 on SafeWatch-Bench-Real. Moreover, ATR enables reliable steering of the precision–recall operating point, supporting deployment scenarios with heterogeneous policy requirements. Code and checkpoints are provided at `https://bruceyg.github.io/ATPO-project-page/`.

1 Introduction

The rapid growth of video-based social media platforms has led to an increasing amount of harmful video content[1]. Recent advances in Vision-Language Models (VLMs) have enabled strong performance on video understanding tasks [1, 2, 5, 17, 25, 41], and there is a growing amount of work using VLMs for video safety detection [4, 10, 33, 42, 46]. However, existing work on harmful video detection typically formulates the task as binary classification, distinguishing safe from unsafe content while treating all unsafe categories as a single class. Such formulations overlook the multi-class structure of unsafe content and are limited in their ability to provide the fine-grained and interpretable classification outcomes required by human content moderators [3, 26]. In practice, **video safety de-**

[*]Work done during internship at Xiaohongshu Inc.
[†]Equal contribution.
[‡]Corresponding authors.
[1]This paper contains content for demonstration purposes that may be disturbing for some readers.

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

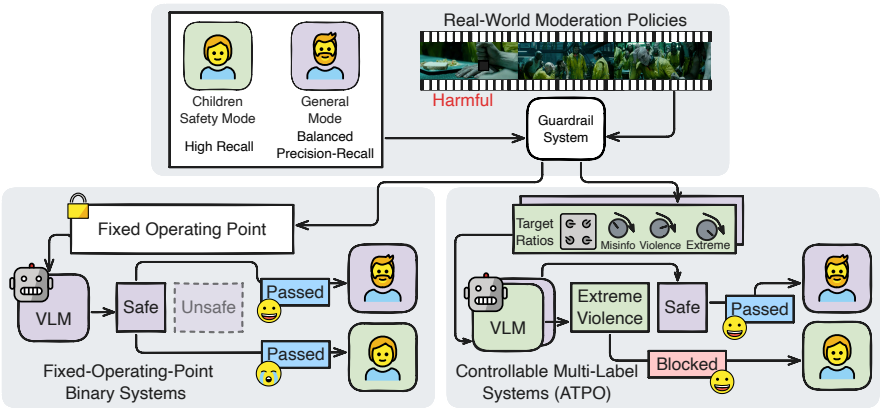


Figure 1: Real-world moderation policies require different precision–recall trade-offs: fixed-operating-point binary systems may miss harmful content and incorrectly pass it to sensitive audiences, whereas our proposed ATPO enables category-aware precision–recall control in multi-label video safety detection.

tection is inherently multi-label: unsafe videos often involve overlapping categories (e.g., violence and abuse) due to their rich and complex semantics.

Moreover, prior approaches typically optimize a static training objective and evaluate models at fixed operating settings [9, 10, 47]. This setup does not explicitly enable **controllable precision–recall trade-offs**, and is therefore misaligned with real-world deployment settings. In practice, safety detection systems are often integrated as guardrails within multi-stage moderation pipelines, where the desired precision-recall operating point depends critically on policy requirements. For example, guardrail systems designed for children must prioritize high recall to minimize harmful exposure. In contrast, general-purpose moderation systems require a more balanced trade-off between precision and recall [6, 38]. Importantly, the optimal precision-recall trade-off may also differ across unsafe categories. Multi-label safety detection inherently involves heterogeneous risk profiles. Certain categories (e.g., misinformation) may prioritize higher precision to reduce over-censorship, while others (e.g., violent or extremist content) may prioritize higher recall to minimize the risk of harmful exposure.

To address these gaps, we propose Adaptive Tversky Policy Optimization (ATPO), a Reinforcement Learning (RL) framework built upon Group Relative Policy Optimization (GRPO) [30]. We first move beyond binary supervision by designing fine-grained multi-label safety training and rewards for GRPO, and systematically study reward formulations for multi-label optimization. Building on this, we draw inspiration from feedback loops in control theory to develop the Adaptive Tversky Reward (ATR). ATR reformulates the Tversky coefficients [35], which govern False Positive (FP) and False Negative (FN) penalties, as feedback-driven variables that dynamically adjust during training based on observed model errors. It enables controllable and category-aware safety behavior within a unified multi-label detection model. Experiments on SafeWatch-Bench [4] and XD-Violence [39] show that ATPO substantially improves Multi-label Video Safety Detection (Multi-VSD) over large general-purpose VLMs while maintaining strong binary performance. Furthermore, ATPO provides reliable control over the precision–recall trade-off both globally and at the category level, offering a practical mechanism for deployment.

Our contributions can be summarized as:

1. We formulate multi-label harmful video detection under RL-based fine-tuning and systematically study reward formulations for optimizing multi-label safety objectives.

2. We propose Adaptive Tversky Policy Optimization (ATPO), a feedback-driven RL framework that dynamically adjusts false-positive and false-negative penalties during training via the Adaptive Tversky Reward (ATR). Extensive experiments show that ATPO substantially improves multi-label video safety detection, increasing the Jaccard Index from 40.66 to 75.44 (Section 4.2 and 4.3).

3. Our experiments show that ATR enables reliable control over the precision–recall trade-off both globally and at the category level, supporting deployment scenarios with different risk sensitivities (Section 4.4).

2 Related Work

Prior research on video safety detection has largely been studied through the paradigm of Video Anomaly Detection (VAD) [19, 23, 24, 28, 31, 39], where the objective is to determine whether a video, typically recorded by surveillance cameras, contains abnormal events. Early approaches commonly formulated VAD as a binary classification problem that distinguishes normal from anomalous activities [31]. More recently, a line of work has leveraged Vision-Language Models (VLMs) for video anomaly detection and understanding [10, 42, 45–47]. For example, Holmes-VAD [46] trains a VLM with temporal sampling to detect anomalies and generate explanations for abnormal events in long videos, while VAD-R1 [10] introduces Video Anomaly Reasoning (VAR), a structured chain-of-thought pipeline further optimized with an RL-based AVA-GRPO algorithm. Beyond binary detection, recent work has begun to study fine-grained video safety and anomaly recognition. Ex-VAD [9] extends VAD to recognize multiple abnormal event categories using vision-language supervision, but formulates the task as multi-class classification, where each video is assigned a single abnormal category. Open-Vocabulary VAD [13, 40] uses pretrained VLMs and textual semantics to detect both seen and unseen anomaly categories. More recently, SafeWatch-Bench [4] explicitly frames video safety detection as a multi-category moderation task. Despite this progress, most prior methods either focus on binary anomaly detection or assign a single abnormal category to each video. In contrast, we study Multi-label Video Safety Detection (Multi-VSD), where multiple harmful categories may co-occur in the same video.

Reinforcement learning has also been explored for improving the reasoning and perception capabilities of VLMs for videos [5, 10, 14–16, 41]. Recent works apply GRPO-style RL fine-tuning to video understanding tasks, where reward functions are designed to encourage stronger temporal reasoning, spatio-temporal perception, and instruction-following abilities. Video-R1 [5] introduces reinforcement fine-tuning for video reasoning and designs temporal rewards to improve the model's ability to reason over video frames. VideoChat-R1 [15, 41] further studies reinforcement fine-tuning for video dialogue and spatio-temporal perception, using task-specific rewards for video question answering and temporal understanding. Other recent works investigate reward design and data efficiency for VideoLLM RL [14], and use RL to improve temporal sampling and temporal grounding in video LLMs [16]. For safety-related tasks, GuardReasoner-VL [20] trains VLM guardrails with reasoning SFT and GRPO, while VAD-R1 [10] applies RL to structured video anomaly reasoning. These studies show that reinforcement learning provides an effective post-training mechanism for adapting VLMs to video reasoning and safety tasks. Building on these developments, we study RL-based reward design for Multi-VSD, introducing adaptive rewards that account for partial label correctness and asymmetric FP/FN costs to enable controllable precision-recall trade-offs during training.

Recent work also studies video safety beyond anomaly detection. For generative models, T2VSafetyBench [27] evaluates safety risks in text-to-video generation, while SAFREE [44] provides a training-free safeguard for text-to-image and video generation. For video and multimodal LLMs, VideoSafety-R1 [32] combines alarm-token-guided SFT with safety-guided GRPO to improve defensive reasoning. SafeVid [36] constructs video-specific safety preference data and applies Direct Preference Optimization (DPO) [29], whereas SEA [22] uses synthetic modality embeddings to enable low-resource safety alignment from textual data. Complementary security and privacy studies examine video-modality jailbreaks through VideoJail [8] and protective adversarial watermarks that disrupt unauthorized video annotation [18]. These directions address safe generation, safe-response alignment, or protection against misuse of video inputs. ATPO instead studies moderation of existing videos through fine-grained multi-label predictions.

3 Proposed Method

Our training pipeline consists of Supervised Fine-Tuning (SFT) warmup followed by reinforcement learning (RL). We first introduce the task formulation and notation for Multi-label Video Safety Detection (Multi-VSD) (Section 3.1). We study different reward formulations, including standard

overlap-based rewards (Section 3.2) and our proposed Adaptive Tversky Policy Optimization (ATPO) (Section 3.3). The overall pipeline is illustrated in Figure 2.

3.1 Multi-label Video Safety Detection Formulation

We formulate the task of Multi-label Video Safety Detection (Multi-VSD) by extending previous binary detection. Given a video v, the model predicts whether the content belongs to any of a predefined set of unsafe categories $\mathcal{C} = \{c_1, \ldots, c_K\}$, where each c represents a type of harmful content (e.g., violence, misinformation, extremism). For each video, the model produces a predicted label set $P \subseteq \mathcal{C}$, while the ground-truth annotation is a label set $G \subseteq \mathcal{C}$. We denote True Positive (TP), False Positive (FP), and False Negative (FN) as

$$TP = |P \cap G|, \qquad FP = |P \setminus G|, \qquad FN = |G \setminus P|. \tag{1}$$

These quantities characterize the quality of the predicted label set and form the basis of the reward functions used for reinforcement learning and evaluation metrics.

3.2 GRPO with Static Rewards

Our training follows a two-stage pipeline consisting of SFT warmup followed by RL training, following recent works on post-training VLMs [5, 14, 49]. We first perform SFT on the backbone VLM using a multi-label classification objective, as shown in Figure 2, where the target outputs correspond to the ground-truth label sets, without auxiliary annotations such as Chain-of-Thought (CoT) [37] reasoning.

Following the SFT warmup stage, we further improve the multi-label prediction capability of VLMs using RL training. Group Relative Policy Optimization (GRPO) [30] is adopted to optimize the model using reward functions defined over predicted and ground-truth label sets. Given an input video v and detection instruction x, the old policy model $\pi_{\theta_{\text{old}}}$ generates N sampled outputs $\{y_j\}_{j=1}^{N}$. Each output contains a predicted label set P_j. We evaluate each prediction using a reward function $R_j = \mathcal{R}(P_j, G)$ that measures its agreement with the ground-truth label set G. Following GRPO, advantages are computed by normalizing rewards within the sampled group:

$$A_j = \frac{R_j - \text{mean}(\{R1, ..., R_N\})}{\text{std}(\{R1, ..., R_N\})}. \tag{2}$$

The policy is then updated by the following GRPO objective, where $\mathbf{x} = (x, v)$:

$$\mathcal{L}_{\text{G}}(\theta) = -\frac{1}{N}\sum_{j=1}^{N}\Big[\min\Big(\frac{\pi_\theta(y_j \mid \mathbf{x})}{\pi_{\theta_{\text{old}}}(y_j \mid \mathbf{x})}A_j, \text{clip}\Big(\frac{\pi_\theta(y_j \mid \mathbf{x})}{\pi_{\theta_{\text{old}}}(y_j \mid \mathbf{x})}, 1-\epsilon, 1+\epsilon\Big)A_j\Big) - \beta' D_{\text{KL}}(\pi_\theta \,\|\, \pi_{\text{ref}})\Big] \tag{3}$$

Exact Match reward. As a basic reward function, we consider Exact Match (EM), which assigns a reward of 1 only when the predicted label set exactly matches the ground-truth set:

$$\mathcal{R}^{\text{EM}}(P, G) = \mathbf{1}[P = G]. \tag{4}$$

While simple, EM is a strict objective that provides sparse feedback. As we show in Section 4.3, such sparse rewards lead to suboptimal learning for Multi-VSD, motivating the use of partial reward functions based on TP, FP, and FN statistics.

Partial rewards based on set overlap. The Jaccard Index (JI) [11] is defined as the Intersection-over-Union between P and G:

$$\mathcal{R}^{\text{Jaccard}}(P, G) = \frac{|P \cap G|}{|P \cup G|} = \frac{TP}{TP + FP + FN} \tag{5}$$

It treats false positives and false negatives symmetrically and corresponds to a special case of the Tversky Index with equal weights.

The Tversky Index (TI) [35] generalizes the Jaccard Index by introducing asymmetric penalties:

$$\mathcal{R}^{\text{STR}}_{\alpha,\beta}(P, G) = \frac{TP}{TP + \alpha\, FP + \beta\, FN}, \tag{6}$$

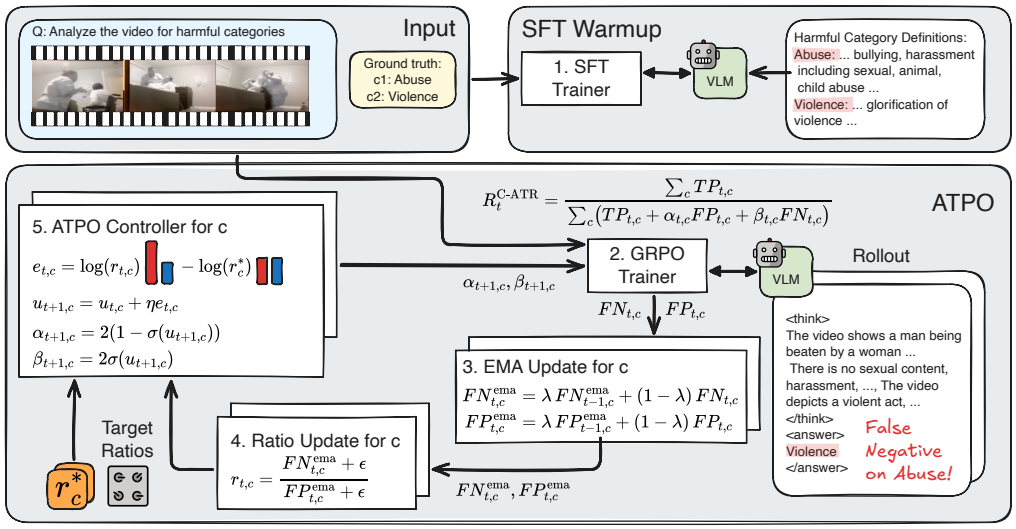


Figure 2: Overview of Adaptive Tversky Policy Optimization (ATPO). The training pipeline consists of five stages. (1) SFT warmup: The VLM is first supervised fine-tuned using harmful video category definitions. (2) GRPO training: The model generates predictions through GRPO rollouts, which are evaluated using Adaptive Tversky Reward (ATR) and model parameters are updated. (3) Category-wise FNs and FPs from the rollouts are aggregated and tracked using EMA. (4) The FN-to-FP ratio is updated using the EMA statistics. (5) ATPO controller compares the ratio with a user-specified target ratio r_c^* and updates the Tversky coefficients $\alpha_{t+1,c}, \beta_{t+1,c}$ for the next training step.

where $\alpha, \beta > 0$ control the penalties of precision and recall. We note that the coefficient β here is unrelated to the KL regularization coefficient β' in Equation 3. We refer to this reward function as Static Tversky Reward (STR) because the coefficients are fixed throughout training. When $\alpha = \beta = 1$, STR reduces to the Jaccard reward. Increasing α emphasizes precision (penalizing FP more), while increasing β emphasizes recall (penalizing FN more).

3.3 Adaptive Tversky Policy Optimization (ATPO)

In practical detection pipelines, different deployment scenarios may require different precision-recall trade-offs. Instead of fixing (α, β) statically, we propose to adapt them dynamically during training so that the model approaches a desired false-negative to false-positive ratio. This leads to our proposed ATPO, which uses a feedback controller that dynamically adjusts the reward function during training. The overall training pipeline is illustrated in Figure 2. Training first performs SFT warmup, then the model is optimized using ATPO, starting with the GRPO trainer with initialized Tversky coefficients. We instantiate ATPO with two variants of reward formulation. For brevity, we omit the GRPO group notation. The full algorithm is provided in Appendix B.

Category-Aware Adaptive Tversky Reward (C-ATR) maintains separate controllers and adaptive Tversky coefficients $(\alpha_{t,c}, \beta_{t,c})$ for each category $c \in \mathcal{C}$. At each training step t, GRPO rollouts produce predictions from which we compute batch-level counts of TP, FP, and FN:

$$TP_{t,c} = \sum_i \mathbf{1}[c \in P_i \cap G_i], \quad FP_{t,c} = \sum_i \mathbf{1}[c \in P_i \setminus G_i], \quad FN_{t,c} = \sum_i \mathbf{1}[c \in G_i \setminus P_i], \quad (7)$$

where P_i and G_i denote the predicted and ground-truth label sets for sample i. Following Step 3 in Figure 2, we maintain Exponential Moving Averages (EMA) of false positives and false negatives:

$$FP_{t,c}^{\text{ema}} = \lambda FP_{t-1,c}^{\text{ema}} + (1-\lambda) FP_{t,c}, \quad FN_{t,c}^{\text{ema}} = \lambda FN_{t-1,c}^{\text{ema}} + (1-\lambda) FN_{t,c} \quad (8)$$

where $\lambda \in [0, 1)$ is a smoothing factor. Using the EMA statistics, we update the estimate of the observed FN-to-FP ratio for category c in Step 4: $r_{t,c} = (FN_{t,c}^{\text{ema}} + \varepsilon)/(FP_{t,c}^{\text{ema}} + \varepsilon)$, where $\varepsilon > 0$ ensures numerical stability. Then in Step 5, the ATPO controller for category c measures the error by comparing the observed FN-to-FP ratio with user-given category-specific target ratio r_c^* and updates

a logit parameter $u_{t,c}$ that governs the relative weighting between false positives and false negatives:

$$e_{t,c} = \log r_{t,c} - \log r_c^*, \qquad u_{t+1,c} = u_{t,c} + \eta\, e_{t,c}, \tag{9}$$

where $\eta > 0$ is the step size. We then compute the Tversky coefficients for the next training step that satisfy

$$\log \frac{\beta_{t+1,c}}{\alpha_{t+1,c}} = u_{t+1,c}, \qquad \alpha_{t+1,c} + \beta_{t+1,c} = 2. \tag{10}$$

The second constraint fixes the overall scale of the coefficients and recovers the Jaccard Index at the neutral operating point $\alpha_{t+1,c} = \beta_{t+1,c} = 1$. Solving the system yields $\alpha_{t+1,c} = 2\big(1 - \sigma(u_{t+1,c})\big)$ and $\beta_{t+1,c} = 2\sigma(u_{t+1,c})$, where $\sigma(\cdot)$ denotes the sigmoid function. C-ATR uses micro-style aggregation across categories to calculate the reward:

$$R_t^{\text{C-ATR}} = \frac{\sum_c TP_{t,c}}{\sum_c \big(TP_{t,c} + \alpha_{t,c} FP_{t,c} + \beta_{t,c} FN_{t,c}\big)} \tag{11}$$

for training step t. By adapting $(\alpha_{t,c}, \beta_{t,c})$ independently, C-ATR allows each category to converge toward its own target precision-recall operating point within a unified multi-label objective.

Global Adaptive Tversky Reward (G-ATR) is a special case of C-ATR that enforces a single global trade-off shared across all categories. Instead of maintaining per-category statistics, we first aggregate errors:

$$TP_t = \sum_c TP_{t,c}, \quad FP_t = \sum_c FP_{t,c}, \quad FN_t = \sum_c FN_{t,c}. \tag{12}$$

We then compute global EMA statistics and the global FN-to-FP ratio and apply the same log-ratio control mechanism as in Equation 9, but with a single logit parameter u_t and target ratio r^*. The resulting adaptive coefficients (α_t, β_t) are shared across all categories, and the reward reduces to the standard global Tversky Index:

$$R_t^{\text{G-ATR}} = \frac{TP_t}{TP_t + \alpha_t FP_t + \beta_t FN_t}. \tag{13}$$

Compared to C-ATR, G-ATR enforces a uniform precision-recall trade-off across all categories, whereas C-ATR enables category-specific operating points.

4 Experiments

In this section, we evaluate ATPO and the proposed Adaptive Tversky Reward (ATR) on two video safety benchmarks. We first compare our models against large general-purpose VLMs and GRPO with static reward designs. We then analyze ATR's ability to control the precision-recall trade-off both globally and at the category level, and study the training dynamics of the adaptive controller.

4.1 Experiment Setup

Models are evaluated on two public video safety benchmarks: SafeWatch-Bench [4] and XD-Violence [39]. SafeWatch-Bench contains 199604 training videos and 1400 test videos across six unsafe categories. We use 11036 full training videos under two minutes for SFT and GRPO, and evaluate on the 810-video SafeWatch-Bench-Real test set. For XD-Violence, we use 3520 training videos under five minutes and evaluate on the full 800-video test set. Additional dataset statistics, including category co-occurrence, are provided in Appendix C.1.

We adopt Qwen2.5-VL-7B-Instruct [2] and Qwen3-VL-8B-Instruct [1] as backbone models, both of which support direct video inputs via uniform frame sampling. For the two training stages, SFT is performed with LoRA [7] using ground-truth safety labels as targets. GRPO training starts from the SFT one-epoch checkpoints and optimizes the reward formulations described in Section 3.2 and 3.3, together with a small format reward. More implementation details are provided in Appendix C.2.

Four standard multi-label evaluation metrics are reported [12, 34, 48]: Jaccard Index, Micro-F1, Macro-F1, and Binary F1. Jaccard Index measures the intersection-over-union between predicted and ground-truth label sets, as defined in Section 3.2. Micro-F1 computes F1 after aggregating true positives, false positives, and false negatives across categories, while Macro-F1 averages category-wise F1 scores. Binary F1 collapses all unsafe categories into a single positive class, following prior video safety detection works [10, 20].

Table 1: Main results on **SafeWatch-Bench-Real** and **XD-Violence**. We report multi-label metrics (Jaccard, Micro F1, Macro F1) and binary detection performance (Binary F1). Models are grouped into general-purpose VLMs, video reasoning VLMs, guardrail systems, and our ATPO-trained models with G-ATR and C-ATR. **Bold** and underlining indicate the best and second-best results. *: further trained on the respective benchmark training sets. †: SafeWatch-8B uses full videos and clips from SafeWatch-Bench.

#	Model	SafeWatch-Bench-Real				XD-Violence			
		Jaccard	Micro	Macro	Binary	Jaccard	Micro	Macro	Binary
	General-purpose VLMs								
1	Qwen2.5VL-7B	40.66	42.17	28.74	65.94	76.54	72.29	62.78	90.21
2	Qwen2.5VL-32B	41.66	43.60	32.11	64.63	77.69	73.28	62.71	91.85
3	Qwen3VL-8B	48.69	52.45	40.32	77.76	81.46	77.15	70.87	94.76
4	Qwen3VL-235B-A22B	55.85	62.10	56.83	80.29	82.53	78.03	**72.29**	95.36
5	InternVL3.5-38B	60.61	64.05	53.55	85.80	83.38	79.59	70.87	93.59
	Video reasoning VLMs								
6	Video-R1	39.99	39.73	23.98	66.07	76.60	72.55	66.02	90.37
7	VideoChat-R1	44.20	47.47	33.99	75.02	72.62	66.82	58.88	89.31
	Guardrails								
8	GuardReasoner-VL	19.90	1.17	0.98	1.22	35.34	3.09	2.78	4.50
9	VAD-R1	47.78	50.24	43.90	80.95	-	-	-	-
10	VAD-R1*	46.36	51.44	48.93	81.13	74.12	68.60	68.81	89.69
11	SafeWatch-8B†	70.69	73.15	71.37	89.04	79.84	77.19	69.65	86.80
	ATPO-trained (ours)								
12	ATPO-G-7B	75.44	77.25	76.10	94.82	**88.17**	**85.24**	71.95	**96.69**
13	ATPO-C-7B	75.15	**77.42**	**76.61**	**95.25**	87.60	84.51	71.24	95.59
14	ATPO-G-8B	**75.50**	77.08	75.25	95.22	85.93	82.82	69.86	94.59
15	ATPO-C-8B	74.63	77.19	76.09	93.79	86.60	83.30	70.51	95.75

4.2 Multi-Label Video Safety Detection Performance

We evaluate several representative VLMs as baselines, including large general-purpose VLMs of different scales from the Qwen-VL and InternVL3.5 families. We also evaluate recent GRPO-trained video reasoning models, Video-R1 [5] and VideoChat-R1 [41], which are built on Qwen2.5-VL-7B and trained for general video understanding tasks. Our guardrail baselines include GuardReasoner-VL [20], VAD-R1 [10], and SafeWatch-8B [4], a specialized multi-label video guardrail trained on SafeWatch-Bench.

Evaluation results for baseline models, prior work, and our models on SafeWatch-Bench-Real and XD-Violence are reported in Table 1. Our models are denoted as ATPO-G and ATPO-C, corresponding to ATPO trained with G-ATR and C-ATR, respectively, and are based on Qwen2.5-VL-7B and Qwen3-VL-8B backbones. The results reveal several key findings.

Multi-VSD remains challenging. Even very large VLMs (row 4 and row 5) still achieve only moderate Multi-VSD performance on SafeWatch-Bench-Real (Jaccard < 61), despite relatively strong binary detection (Binary F1 > 85). In contrast, most models obtain consistently higher scores on XD-Violence, suggesting XD-Violence is an easier benchmark under the evaluation metrics. Notably, models trained with GRPO for general video reasoning do not yield substantial gains. Video-R1 (row 6) and VideoChat-R1 (row 7) both build on the Qwen2.5-VL-7B backbone. Compared to the backbone's results (row 1), Video-R1 shows no improvement (Jaccard 39.99 *vs.* 40.66), and VideoChat-R1 provides only a modest gain (Jaccard 44.20 *vs.* 40.66). This indicates that general video reasoning training alone is insufficient for Multi-VSD, and that dedicated in-domain training is likely necessary.

ATPO improves Multi-VSD performance. ATPO substantially improves Multi-VSD performance over the backbone models. For Qwen2.5-VL-7B, G-ATR and C-ATR (rows 12-13) improve SafeWatch-Bench-Real Jaccard from 40.66 (row 1) to 75.44/75.15 (a $\sim$+35 point gain). For Qwen3-VL-8B, G-ATR and C-ATR (rows 14-15) improve Jaccard from 48.69 (row 3) to 75.50/74.63. Overall, ATPO-trained models substantially outperform both larger general-purpose VLMs and video reasoning VLMs based on the same backbone model.

Table 2: Training comparison on Qwen2.5-VL-7B. GRPO uses static rewards (Exact Match and STR), whereas ATPO-G and ATPO-C use adaptive rewards G-ATR and C-ATR, respectively. All RL methods start from the SFT-1ep checkpoint. Micro, Macro, and Binary denote F1 scores. **Bold** and underlining indicate the best and second-best results per column.

Training	SafeWatch-Bench-Real				SafeWatch-Bench-GenAI			
	Jaccard	Micro	Macro	Binary	Jaccard	Micro	Macro	Binary
Zero-shot	40.66	42.17	28.74	65.94	47.18	43.30	27.18	75.54
SFT-1ep *(RL initialization)*	44.33	44.73	38.47	63.53	66.34	71.40	67.90	90.20
SFT-3ep *(extended SFT)*	67.93	72.89	71.05	92.02	62.32	70.49	68.11	87.86
RL from the same SFT-1ep checkpoint								
GRPO (EM)	63.36	67.59	63.88	92.30	62.23	68.84	64.79	89.48
GRPO (STR)	70.09	72.27	63.88	93.97	69.12	73.79	64.38	91.12
ATPO-G	**75.44**	<u>77.25</u>	<u>76.10</u>	<u>94.82</u>	**74.29**	**76.24**	**74.46**	<u>93.30</u>
ATPO-C	<u>75.15</u>	**77.42**	**76.61**	**95.25**	<u>73.35</u>	<u>74.30</u>	<u>74.44</u>	**94.05**

ATPO outperforms specialized guardrails. SafeWatch-8B (row 11) provides a strong in-domain baseline, achieving the best non-ATPO results across all four metrics on SafeWatch-Bench-Real. Nevertheless, all four ATPO variants outperform SafeWatch-8B across all eight dataset-metric combinations. In particular, ATPO-G-7B achieves Jaccard scores of 75.44 and 88.17 on SafeWatch-Bench-Real and XD-Violence, compared with 70.69 and 79.84 for SafeWatch-8B, corresponding to gains of 4.75 and 8.33 points. In Appendix H we also show that ATPO outperforms SafeWatch-8B and other training baselines on videos with multiple labels only.

Besides Multi-VSD, **ATPO-trained models also excel at binary classification.** All ATPO-trained models achieve Binary F1 > 93 on both datasets (rows 12-15), indicating that optimizing for multi-label safety detection also leads to strong binary safety detection performance. Notably, the binary performance of these models is higher than guardrail models that are trained specifically for binary detection. Another finding from the results is that **G-ATR and C-ATR have comparable overall performance.** Across both backbones, G-ATR and C-ATR are very close (rows 12 *vs.* 13 and rows 14 *vs.* 15), with minor trade-offs across metrics. This suggests robustness to the choice of global *vs.* category-wise control in our ATPO training.

Overall, Table 1 presents our best results using in-domain training with ATPO, showing substantial improvements over zero-shot general-purpose models. We further examine transfer across the SafeWatch-Bench and XD-Violence taxonomies, including adaptation with only 10% of the target training data, in Appendix L. In the subsequent table, we show the relative contributions of our in-domain training techniques.

4.3 Comparison of Static and Adaptive Reward Optimization

Table 2 compares zero-shot evaluation, SFT, GRPO with static rewards (EM and STR), and ATPO with adaptive rewards (G-ATR and C-ATR), using Qwen2.5-VL-7B. We evaluate on both the real and AI-generated test subsets of SafeWatch-Bench [4], referred to as Real and GenAI below.

ATPO substantially improves over its initialization. Starting from SFT-1ep, ATPO-G and ATPO-C improve Real Jaccard from 44.33 to 75.15, respectively, corresponding to gains of 31.11 and 30.82 points. GenAI Jaccard also increases from 66.34 to 74.29 and 73.35, respectively. Both variants improve all four metrics on both subsets relative to their shared initialization.

Extended SFT trades off Real and GenAI performance. AI-generated videos are less represented in the SafeWatch-Bench training set compared to real videos. Extending SFT from one to three epochs increases Real Jaccard from 44.33 to 67.93, but reduces GenAI Jaccard from 66.34 to 62.32. GenAI Micro and Binary F1 also decline. Thus, the gains from extended SFT do not hold across both video types. In contrast, both ATPO variants outperform SFT-3ep across all four metrics on both subsets. Cross-dataset evaluation on VHD11K-Video [43] shows a similar pattern for extended SFT and ATPO-C (Appendix G).

Adaptive rewards outperform static rewards. Among the static rewards, STR achieves higher Jaccard than EM on both subsets, consistent with the benefit of graded feedback for partially correct

Table 3: Global precision-recall control on SafeWatch-Bench-Real. Comparison between Static Tversky Reward (STR) and Global Adaptive Tversky Reward (G-ATR) under varying control ratios. We report micro Precision (P), Recall (R), their ratio (P/R), and Error Type Ratio (ETR = FN/FP).

STR					G-ATR				
α/β	Precision	Recall	P/R	ETR	r^*	Precision	Recall	P/R	ETR
1 / 5	81.24	73.21	1.11	1.58	0.2	66.33	86.55	0.77	0.31
1 / 2	79.11	69.88	1.13	1.63	0.5	76.18	80.71	0.94	0.76
1 / 1	71.93	72.62	0.99	0.97	1	75.74	78.81	0.96	0.84
2 / 1	75.03	71.55	1.05	1.20	2	75.72	69.05	1.10	1.40
5 / 1	73.48	77.50	0.95	0.80	5	78.56	67.62	1.16	1.75

predictions. Both ATPO variants further outperform EM and STR across all eight dataset-metric combinations. Relative to GRPO with STR, ATPO-G improves Jaccard by 5.35 on Real (70.09 to 75.44) and 5.17 points on GenAI (69.12 to 74.29). ATPO-C achieves corresponding gains of 5.06 and 4.23 points. These matched-RL comparisons support the contribution of adaptive reward design and suggest that adaptively rebalancing FN/FP during training provides a more informative learning signal than EM or STR (qualitative analysis in Appendix I). Moreover, while SFT and static rewards impose a fixed operating point, ATR enables stable RL optimization with controllable error trade-offs, which we examine in the next section. Additional Qwen3-VL-8B results are provided in Appendix F.

4.4 Controlling Recall and Precision with ATR

Next, we investigate the question of whether ATR can be used to control the global precision-recall trade-off through the target FN/FP ratio r^*, and at the category level through per-category target ratio r_c^*.

4.4.1 Global Control via G-ATR

To quantify global control, we report micro Precision (P), micro Recall (R), and their ratio P/R in Table 3. We additionally report Error Type Ratio (ETR), defined as FN/FP, which directly measures the relative prevalence of type-II errors (false negatives) *vs.* type-I errors (false positives). For G-ATR, we vary the target ratio r^*: a smaller r^* is expected to suppress FN and thus encourage a recall-dominant operating point, whereas a larger r^* should suppress FP and shift the model toward a precision-dominant regime. For comparison, we evaluate the Static Tversky Reward (STR) defined in Equation 6, where the coefficients α and β are fixed. Under STR, a smaller α/β places greater penalty on FN, while a larger α/β penalizes FP more. We make the following two observations from Table 3.

G-ATR yields monotonic global control. Across the full sweep of target ratios, G-ATR produces a clear and monotonic progression in the achieved precision-recall ratio. As r^* increases, precision increases while recall decreases, and the achieved P/R rises steadily from $0.77(r^* = 0.2)$ to $1.16(r^* = 5)$. Notably, ETR follows the same intended direction: as r^* grows, FN/FP increases monotonically from 0.31 to 1.75. Overall, varying r^* smoothly shifts the model from a recall-favoring regime to a precision-favoring regime, suggesting that G-ATR functions as a stable feedback controller for global steering. The corresponding precision–recall curves are provided in Appendix D. In contrast, we find that **STR does not provide reliable control** under fixed FN and FP weightings. The achieved P/R under STR is non-monotonic and sometimes moves counter to the desired direction as α/β increases. The lack of consistency is further reflected by ETR, which does not exhibit a clean monotonic relationship with α/β. This instability indicates that simply fixing the ratio $\frac{\alpha}{\beta}$ is insufficient for predictable global precision-recall steering during GRPO training. We additionally compare ATPO-G with post-hoc threshold tuning of SFT and GRPO (STR) in Appendix K.

4.4.2 Category-Level Control via C-ATR

To evaluate category-level control, we select the two strongest general-purpose models from Table 1, InternVL3.5-38B and Qwen3-VL-235B-A22B, as reference models. We focus on C4 (*Misinformation*) and C6 (*Extreme*) because misinformation detection may prioritize higher precision to reduce

Table 4: Category-level precision-recall control on SafeWatch-Bench-Real. Comparison between zero-shot large general-purpose VLMs (InternVL3.5-38B and Qwen3-VL-235B-A22B) and ATPO with C-ATR under symmetric and asymmetric target ratios for *Misinformation* and *Extreme*. We report category-wise Precision, Recall, their ratio P/R, and Error Type Ratio (ETR).

Model	r_4^*	r_6^*	C4 (*Misinformation*)				C6 (*Extreme*)			
			Precision	Recall	P/R	ETR	Precision	Recall	P/R	ETR
InternVL	-	-	94.44	19.54	4.83	70.00	94.74	15.79	6.00	96.00
Qwen3-VL	-	-	95.83	22.55	4.25	79.00	96.43	20.93	4.61	102.00
C-ATR	1.0	1.0	55.28	66.67	0.83	0.62	80.31	79.07	1.02	1.08
C-ATR	5.0	0.2	70.41	67.65	1.04	1.14	31.77	94.57	0.36	0.03

over-censorship, whereas extreme content prioritizes higher recall to minimize harmful exposure. We report category-wise Precision, Recall, P/R, and ETR on SafeWatch-Bench-Real in Table 4.

Over-conservative behavior of large VLMs. We find that large VLMs exhibit high-precision, low-recall "over-conservative" behavior. Both InternVL and Qwen3-VL show conservative behavior on both C4 and C6. They achieve very high precision (> 94) but low recall (< 23), resulting in disproportionately large P/R and very high ETR values. This indicates that, without task-specific alignment, even highly capable models tend not to flag these two evaluated categories, leading to many missed harmful videos. In comparison, **C-ATR enables controllable, category-level precision-recall trade-offs.** The controlled setting ($r_4^* = 5.0, r_6^* = 0.2$) demonstrates the desired asymmetric steering: C4 shifts toward higher precision (from 55.28 to 70.41), while C6 shifts toward high recall (79.07 to 94.57). Overall, these results show that C-ATR can override the "overly conservative" zero-shot behavior and provide practical, category-aware operating points aligned with heterogeneous safety requirements. We report complete results for all six categories in Appendix J and analyze controller dynamics under different step sizes in Appendix E. Practical guidance on selecting error-ratio targets for deployment is provided in Appendix M.

5 Conclusion

In this paper, we introduced ATPO, an adaptive reward framework for multi-label harmful video detection that enables controllable precision–recall trade-offs during RL training. Experiments show that ATPO improves multi-label detection while maintaining strong binary safety performance. Furthermore, ATPO provides a flexible mechanism for aligning model behavior with different moderation policies by adjusting the desired precision–recall operating point.

6 Acknowledgments

Guangyu Yang is supported by Cambridge Commonwealth, European and International Trust for the undertaking of the PhD in Engineering at the University of Cambridge. This work was conducted during an internship at Xiaohongshu Inc.

Jingbiao Mei is supported by Cambridge Commonwealth, European and International Trust for the undertaking of the PhD in Engineering at the University of Cambridge.

Jinghong Chen is supported by the Warwick Postgraduate Studentship from Christ's College and the Huawei HiSilicon Studentship for the undertaking of the PhD in Engineering at the University of Cambridge.

Prof. Bill Byrne holds concurrent appointments as a Professor of Information Engineering at Cambridge University and as an Amazon Scholar. This publication describes work performed at Cambridge University and is not associated with Amazon.

We would also like to thank all the reviewers for their knowledgeable reviews.

References

[1] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025. URL https://arxiv.org/abs/2502.13923.

[3] Agostina Calabrese, Leonardo Neves, Neil Shah, Maarten Bos, Björn Ross, Mirella Lapata, and Francesco Barbieri. Explainability and hate speech: Structured explanations make social media moderators faster. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 398–408, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-short.38. URL https://aclanthology.org/2024.acl-short.38/.

[4] Zhaorun Chen, Francesco Pinto, Minzhou Pan, and Bo Li. Safewatch: An efficient safety-policy following video guardrail model with transparent explanations. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu, editors, *International Conference on Learning Representations*, volume 2025, pages 76566–76608, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/beac6bfb7eac3d651307c16ac747df01-Paper-Conference.pdf.

[5] Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Junfei Wu, Xiaoying Zhang, Benyou Wang, and Xiangyu Yue. Video-r1: Reinforcing video reasoning in MLLMs. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=a2JTVVvcEl.

[6] Vaishali U. Gongane, Mousami V. Munot, and Alwin D. Anuse. Detection and moderation of detrimental content on social media platforms: current status and future directions. *Social Network Analysis and Mining*, 12(1):129, 2022. ISSN 1869-5469. doi: 10.1007/s13278-022-00951-3.

[7] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

[8] Wenbo Hu, Shishen Gu, Youze Wang, and Richang Hong. Videojail: Exploiting video-modality vulnerabilities for jailbreak attacks on multimodal large language models. In *ICLR 2025 Workshop on Building Trust in Language Models and Applications*, 2025. URL https://openreview.net/forum?id=fSAIDcPduZ.

[9] Chao Huang, Yushu Shi, Jie Wen, Wei Wang, Yong Xu, and Xiaochun Cao. Ex-VAD: Explainable fine-grained video anomaly detection based on visual-language models. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 25750–25761. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/huang25ad.html.

[10] Chao Huang, Benfeng Wang, Wei Wang, Jie Wen, Chengliang Liu, Li Shen, and Xiaochun Cao. Vad-r1: Towards video anomaly reasoning via perception-to-cognition chain-of-thought. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=6eLlczxbrp.

[11] Paul Jaccard. The distribution of the flora in the alpine zone. *New Phytologist*, 11(2):37–50, 1912. doi: https://doi.org/10.1111/j.1469-8137.1912.tb05611.x. URL https://nph.onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-8137.1912.tb05611.x.

[12] Oluwasanmi O Koyejo, Nagarajan Natarajan, Pradeep K Ravikumar, and Inderjit S Dhillon. Consistent multilabel classification. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper_files/paper/2015/file/85f007f8c50dd25f5a45fca73cad64bd-Paper.pdf.

[13] Fei Li, Wenxuan Liu, Jingjing Chen, Ruixu Zhang, Yuran Wang, Xian Zhong, and Zheng Wang. Anomize: Better open vocabulary video anomaly detection. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 29203–29212, 2025. doi: 10.1109/CVPR52734.2025.02719.

[14] Hongyu Li, Songhao Han, Yue Liao, Junfeng Luo, Jialin Gao, Shuicheng Yan, and Si Liu. Reinforcement learning tuning for videollms: Reward design and data efficiency. *arXiv preprint arXiv:2506.01908*, 2025.

[15] Xinhao Li, Ziang Yan, Desen Meng, Lu Dong, Xiangyu Zeng, Yinan He, Yali Wang, Yu Qiao, Yi Wang, and Limin Wang. Videochat-r1: Enhancing spatio-temporal perception via reinforcement fine-tuning, 2025. URL https://arxiv.org/abs/2504.06958.

[16] Yunheng Li, Jing Cheng, Shaoyong Jia, Hangyi Kuang, Shaohui Jiao, Qibin Hou, and Ming-Ming Cheng. Tempsamp-r1: Effective temporal sampling with reinforcement fine-tuning for video llms, 2025. URL https://arxiv.org/abs/2509.18056.

[17] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-LLaVA: Learning united visual representation by alignment before projection. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5971–5984, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.342. URL https://aclanthology.org/2024.emnlp-main.342/.

[18] Haitong Liu, Kuofeng Gao, Yang Bai, Jinmin Li, Jinxiao Shan, Tao Dai, and Shu-Tao Xia. Protecting Your Video Content: Disrupting Automated Video-based LLM Annotations . In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24056–24065, Los Alamitos, CA, USA, June 2025. IEEE Computer Society. doi: 10.1109/CVPR52734.2025.02240. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52734.2025.02240.

[19] 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.

[20] Yue Liu, Shengfang Zhai, Mingzhe Du, Yulin Chen, Tri Cao, Hongcheng Gao, Cheng Wang, Xinfeng Li, Kun Wang, Junfeng Fang, Jiaheng Zhang, and Bryan Hooi. Guardreasoner-VL: Safeguarding VLMs via reinforced reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=Ku3XdvO88g.

[21] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

[22] Weikai Lu, Hao Peng, Huiping Zhuang, Cen Chen, and Ziqian Zeng. SEA: Low-resource safety alignment for multimodal large language models via synthetic embeddings. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long*

Papers), pages 24894–24913, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1212. URL https://aclanthology.org/2025.acl-long.1212/.

[23] Weixin Luo, Wen Liu, and Shenghua Gao. A revisit of sparse coding based anomaly detection in stacked rnn framework. *ICCV, Oct*, 1(2):3, 2017.

[24] Weixin Luo, Wen Liu, Dongze Lian, Jinhui Tang, Lixin Duan, Xi Peng, and Shenghua Gao. Video anomaly detection with sparse coding inspired deep neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[25] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Khan. Video-ChatGPT: Towards detailed video understanding via large vision and language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12585–12602, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.679. URL https://aclanthology.org/2024.acl-long.679/.

[26] Jingbiao Mei, Mingsheng Sun, Jinghong Chen, Pengda Qin, Yuhong Li, Da Chen, and Bill Byrne. ExPO-HM: Learning to explain-then-detect for hateful meme detection. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=bEejbORUI5.

[27] Yibo Miao, Yifan Zhu, Lijia Yu, Jun Zhu, Xiao-Shan Gao, and Yinpeng Dong. T2vsafetybench: evaluating the safety of text-to-video generative models. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9798331314385.

[28] Hyunjong Park, Jongyoun Noh, and Bumsub Ham. Learning memory-guided normality for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14372–14381, 2020.

[29] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: your language model is secretly a reward model. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

[30] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y.K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

[31] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[32] Yiwei Sun, Peiqi Jiang, Chuanbin Liu, Luohao Lin, Zhiying Lu, and Hongtao Xie. From evaluation to defense: Advancing safety in video large language models, 2026. URL https://arxiv.org/abs/2505.16643.

[33] Jiaqi Tang, Hao Lu, Ruizheng Wu, Xiaogang Xu, Ke Ma, Cheng Fang, Bin Guo, Jiangbo Lu, Qifeng Chen, and Ying-Cong Chen. Hawk: Learning to understand open-world video anomalies. In *Neural Information Processing Systems (NeurIPS)*, 2024.

[34] Grigorios Tsoumakas, Ioannis Katakis, and Ioannis Vlahavas. *Mining Multi-label Data*, pages 667–685. Springer US, Boston, MA, 2010. ISBN 978-0-387-09823-4. doi: 10.1007/978-0-387-09823-4_34. URL https://doi.org/10.1007/978-0-387-09823-4_34.

[35] Amos Tversky. Features of similarity. *Psychological Review*, 84(4):327–352, 1977. doi: https://doi.org/10.1037/0033-295X.84.4.327.

[36] Yixu Wang, Jiaxin Song, Yifeng Gao, Xin Wang, Yang Yao, Yan Teng, Xingjun Ma, Yingchun Wang, and Yu-Gang Jiang. Safevid: Toward safety aligned video large multimodal models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2026. URL https://openreview.net/forum?id=SeNFo7JGly.

[37] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

[38] Johnny Tian-Zheng Wei, Frederike Zufall, and Robin Jia. *Operationalizing Content Moderation "Accuracy" in the Digital Services Act*, page 1527–1538. AAAI Press, 2025.

[39] 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 (ECCV)*, 2020.

[40] Peng Wu, Xuerong Zhou, Guansong Pang, Yujia Sun, Jing Liu, Peng Wang, and Yanning Zhang. Open-vocabulary video anomaly detection, 2024. URL `https://arxiv.org/abs/2311.07042`.

[41] Ziang Yan, Yinan He, Xinhao Li, Zhengrong Yue, Xiangyu Zeng, Yali Wang, Yu Qiao, Limin Wang, and Yi Wang. Videochat-r1.5: Visual test-time scaling to reinforce multimodal reasoning by iterative perception. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL `https://openreview.net/forum?id=oVDAfLuRie`.

[42] Muchao Ye, Weiyang Liu, and Pan He. Vera: Explainable video anomaly detection via verbalized learning of vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 8679–8688, June 2025.

[43] Chen Yeh, You-Ming Chang, Wei-Chen Chiu, and Ning Yu. T2vs meet vlms: A scalable multimodal dataset for visual harmfulness recognition. In *Advances in Neural Information Processing Systems*, 2024.

[44] Jaehong Yoon, Shoubin Yu, Vaidehi Patil, Huaxiu Yao, and Mohit Bansal. Safree: Training-free and adaptive guard for safe text-to-image and video generation. In *International Conference on Learning Representations*, 2025.

[45] Luca Zanella, Willi Menapace, Massimiliano Mancini, Yiming Wang, and Elisa Ricci. Harnessing large language models for training-free video anomaly detection. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18527–18536, 2024. URL `https://api.semanticscholar.org/CorpusID:268819369`.

[46] Huaxin Zhang, Xiaohao Xu, Xiang Wang, Jialong Zuo, Chuchu Han, Xiaonan Huang, Changxin Gao, Yuehuan Wang, and Nong Sang. Holmes-vad: Towards unbiased and explainable video anomaly detection via multi-modal llm. *arXiv preprint arXiv:2406.12235*, 2024.

[47] Huaxin Zhang, Xiaohao Xu, Xiang Wang, Jialong Zuo, Xiaonan Huang, Changxin Gao, Shanjun Zhang, Li Yu, and Nong Sang. Holmes-vau: Towards long-term video anomaly understanding at any granularity. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13843–13853, 2025. doi: 10.1109/CVPR52734.2025.01292.

[48] Min-Ling Zhang and Zhi-Hua Zhou. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837, 2014. doi: 10.1109/TKDE.2013.39.

[49] Ruohong Zhang, Bowen Zhang, Yanghao Li, Haotian Zhang, Zhiqing Sun, Zhe Gan, Yinfei Yang, Ruoming Pang, and Yiming Yang. Improve vision language model chain-of-thought reasoning. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1662, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.82. URL `https://aclanthology.org/2025.acl-long.82/`.

[50] Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand, 2024. Association for Computational Linguistics. URL `http://arxiv.org/abs/2403.13372`.

[51] Yaowei Zheng, Junting Lu, Shenzhi Wang, Zhangchi Feng, Dongdong Kuang, and Yuwen Xiong. Easyr1: An efficient, scalable, multi-modality rl training framework. `https://github.com/hiyouga/EasyR1`, 2025.

A Limitations

Our evaluation follows the category definitions and annotations provided by existing benchmarks (e.g., SafeWatch-Bench and XD-Violence), which reflect commonly adopted operational definitions of harmful content. However, the boundaries of categories such as misinformation, violence, or extremism may vary across cultural, legal, and platform-specific contexts. As a result, model behavior may differ under alternative definitions or policies. While ATPO provides a mechanism to adjust error trade-offs, adapting to different definitions may require dataset-specific annotation and validation.

ATPO enables controllable precision-recall trade-offs, but selecting appropriate target ratios requires application-specific tuning and validation. In particular, different operating points may lead to over-moderation (false positives) or missed harmful content (false negatives). Therefore, ATPO is best used as an assistive component in moderation pipelines to support human decision-making rather than serving as a fully automated standalone system.

Algorithm 1 ATPO with Category-Aware Adaptive Tversky Reward (C-ATR)

Require: Training set $\mathcal{D} = \{(x, v_i, G_i)\}_{i=1}^{|\mathcal{D}|}$, where x is the detection instruction, v_i is the input video and G_i is the ground-truth label set, initial policy π_θ, smoothing factor λ, step size η, stability constant ε, target ratios $\{r_c^*\}_{c\in\mathcal{C}}$, group size N

Ensure: Trained policy π_θ

Perform supervised fine-tuning (SFT) on $\mathcal{D}$ to initialize π_{θ_0}
for each category $c \in \mathcal{C}$ **do**
  Initialize controller state $u_{0,c} \leftarrow 0$, EMA statistics $FP_{0,c}^{\text{ema}} \leftarrow 0$, $FN_{0,c}^{\text{ema}} \leftarrow 0$, and coefficients $\alpha_{0,c} \leftarrow 1$, $\beta_{0,c} \leftarrow 1$
end for
for training step $t = 0$ to $T-1$ **do**
  Sample minibatch $\mathcal{B} = \{(x, v_i, G_i)\}_{i=1}^{M}$ from $\mathcal{D}$
  for each sample $(x, v_i, G_i) \in \mathcal{B}$ **do**
    Sample a group of N outputs $\{y_{i,j}\}_{j=1}^{N} \sim \pi_{\theta_t}(\cdot \mid x, v_i)$
    Extract predicted label sets $\{P_{i,j}\}_{j=1}^{N}$ from $\{y_{i,j}\}_{j=1}^{N}$
    for each output $j = 1$ to N **do**
      for each category $c \in \mathcal{C}$ **do**
        Compute $TP_{t,c,i,j} \leftarrow \mathbf{1}[c \in P_{i,j} \cap G_i]$, $FP_{t,c,i,j} \leftarrow \mathbf{1}[c \in P_{i,j} \setminus G_i]$, $FN_{t,c,i,j} \leftarrow \mathbf{1}[c \in G_i \setminus P_{i,j}]$
      end for
      Compute reward using current coefficients $\{\alpha_{t,c}, \beta_{t,c}\}_{c\in\mathcal{C}}$
      $R_{t,i,j} \leftarrow \frac{\sum_c TP_{t,c,i,j}}{\sum_c (TP_{t,c,i,j} + \alpha_{t,c} FP_{t,c,i,j} + \beta_{t,c} FN_{t,c,i,j})}$
    end for
    Compute group-normalized advantages
  end for
  Update policy from π_{θ_t} to $\pi_{\theta_{t+1}}$ with GRPO
  for each category $c \in \mathcal{C}$ **do**
    Update EMA statistics
    $FP_{t,c} \leftarrow \sum_{i=1}^{M}\sum_{j=1}^{N} FP_{t,c,i,j}, \quad FP_{t+1,c}^{\text{ema}} \leftarrow \lambda FP_{t,c}^{\text{ema}} + (1-\lambda) FP_{t,c}$
    $FN_{t,c} \leftarrow \sum_{i=1}^{M}\sum_{j=1}^{N} FN_{t,c,i,j}, \quad FN_{t+1,c}^{\text{ema}} \leftarrow \lambda FN_{t,c}^{\text{ema}} + (1-\lambda) FN_{t,c}$
    Compute observed ratio $r_{t,c} \leftarrow \frac{FN_{t+1,c}^{\text{ema}} + \varepsilon}{FP_{t+1,c}^{\text{ema}} + \varepsilon}$
    Compute control error $e_{t,c} \leftarrow \log r_{t,c} - \log r_c^*$
    Update controller $u_{t+1,c} \leftarrow u_{t,c} + \eta e_{t,c}$
    Update next-step coefficients $\alpha_{t+1,c} \leftarrow 2\big(1 - \sigma(u_{t+1,c})\big), \quad \beta_{t+1,c} \leftarrow 2\sigma(u_{t+1,c})$
  end for
end for
return π_{θ_T}

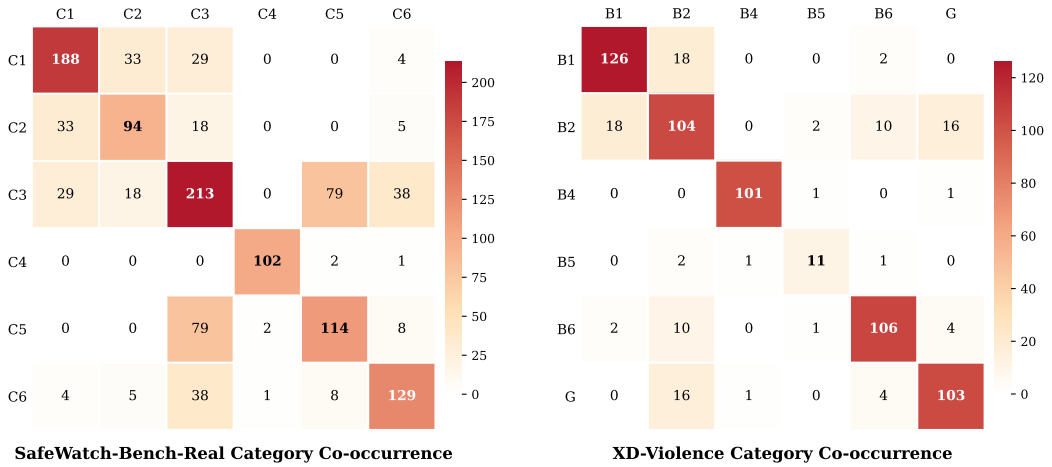


Figure 3: Category co-occurrence matrices for unsafe videos in the test sets of SafeWatch-Bench-Real (left) and XD-Violence (right), showing the number of videos where each pair of categories appears together.

B Full Algorithm

In Section 3.3, we present the training pipeline of Adaptive Tversky Policy Optimization (ATPO). Here we provide the full training procedure of ATPO instantiated with the Category-Aware Adaptive Tversky Reward (C-ATR) in Algorithm 1. The training begins with Supervised Fine-Tuning (SFT) to initialize the Vision-Language Model (VLM) with harmful category definitions. ATPO then iteratively performs GRPO rollouts and policy updates [30], followed by EMA updates, ratio estimation, and controller updates that adapt the Tversky coefficients [35] toward the target ratios.

C Additional Experiment Setup Details

In this section, we provide additional details on the experimental setup for Section 4.1.

C.1 Statistics of Datasets

In Section 4.1, we introduce the two test sets used for evaluation: SafeWatch-Bench-Real [4] and XD-Violence [39]. Here we present additional dataset statistics to illustrate the characteristics of these datasets and the challenges they pose for Multi-label Video Safety Detection (Multi-VSD). Table 5 shows the distribution of samples with each harmful category and multi-label samples among all unsafe samples. We can see that the proportion of multi-label videos is substantially higher in SafeWatch-Bench-Real (25.52%) than in XD-Violence (9.40%). This indicates that SafeWatch more frequently contains videos exhibiting multiple harmful categories simultaneously, which increases prediction ambiguity and helps explain the greater difficulty observed for SafeWatch in Section 4.2. This highlights the importance of methods that can reliably handle multi-label predictions. In addition, XD-Violence exhibits notable category imbalance, with the Abuse category accounting for only 2.2% of unsafe samples. Such imbalance could make it difficult for models to maintain consistent precision–recall trade-offs across categories.

Figure 3 further illustrates the relationships between categories by showing the category co-occurrence matrices for unsafe samples in the two test sets. We can see that several categories frequently co-occur in SafeWatch-Bench-Real, such as C3 (*Violence*) with C5 (*Illegal*) and C6 (*Extreme*), whereas category overlap in XD-Violence is generally more limited. These patterns highlight the inherently multi-label nature of harmful videos, which motivates the development of ATPO for Multi-VSD.

Table 5: Label distribution of unsafe categories in the SafeWatch-Bench-Real test set (650 harmful videos) and XD-Violence test set (500 violent videos). Category identifiers follow the original dataset annotations. Since the task is multi-label, videos may belong to multiple categories, so counts do not sum to the total number of videos.

Category	Count	Percentage (%)
SafeWatch-Bench-Real		
C1 (Sexual)	188	28.92
C2 (Abuse)	94	14.46
C3 (Violence)	213	32.77
C4 (Misinformation)	102	15.69
C5 (Illegal)	114	17.54
C6 (Extreme)	129	19.85
Multi-label	166	25.52
XD-Violence		
B1 (Fighting)	126	25.20
B2 (Shooting)	104	20.80
B4 (Riot)	101	20.20
B5 (Abuse)	11	2.20
B6 (Car Accident)	106	21.20
G (Explosion)	103	20.60
Multi-label	47	9.40

C.2 Implementation Details

Video Processing. Videos are processed using the official preprocessing pipeline `qwen-vl-utils` [1, 2]. Frames are sampled at 1 frame per second (FPS) without enforcing a fixed number of frames per video, so the sequence length varies with video duration. Frames are sampled uniformly across the video. Each frame is resized using dynamic resolution while preserving the original aspect ratio.

Prompt Format. The model receives a task instruction describing the video safety detection objective together with the category definitions. The prompt instructs the model to provide reasoning and output the predicted safety categories in a structured format following the original prompt used in SafeWatch-Bench [4]. The exact prompt template used during ATPO training and inference on SafeWatch-Bench is provided in Figure 4.

Supervised Fine-Tuning (SFT). Training begins with an SFT stage to align the base VLM with the Multi-label Video Safety Detection (Multi-VSD) task. The SFT stage is trained for 1 epoch on the training split of each dataset using the standard language modeling objective. Additional hyperparameters follow common VLM fine-tuning practices, including the optimizer AdamW [21], learning rate $= 10^{-5}$, effective batch size $= 8$, and maximum sequence length $= 24576$. The LoRA rank is set to 64 and the scaling factor is set to 128.

Reinforcement Learning Training. After SFT, we further optimize the model using the proposed ATPO framework. The RL stage runs for 3 epochs. For each input sample, the model generates a group of responses to estimate policy advantages. The key RL hyperparameters are: group size $= 8$, sampling temperature $= 1.0$, top-$p = 1.0$, and KL penalty coefficient $\beta_{\mathrm{KL}} = 10^{-2}$. The reference policy is the SFT-initialized model. For the ATPO controller, we use controller step size $\eta = 0.01$, EMA smoothing factor $\lambda = 0.95$, and stability constant $\varepsilon = 10^{-8}$. These parameters are shared across both the global and category-level variants of ATR.

Training Infrastructure. All experiments are conducted on four NVIDIA H800 80GB GPUs. Training is performed using BF16 mixed-precision. The following libraries/packages were used for implementing the experiments: `LLaMA-Factory` [50], a modified version of `EasyR1` [51], `PyTorch 2.8.0`, `vLLM 0.11.0`, `Transformers 4.57.3`, and `CUDA 12.8`. On average, RL training for three epochs takes approximately 20 hours on our GPU cluster.

```
You are a helpful video guardrail model to detect if a video contains unsafe, harmful content that needs to be moderated.

**As a guardrail model, your task is:**
**Analyze the video for the presence of the following harmful content categories:**

<BEGIN HARMFUL CATEGORIES>
C1: Sexual Content; C2: Harassment & Bullying; C3: Threats, Violence & Harm; C4: False & Deceptive Information; C5: Illegal/Regulated
Activities; C6: Hateful Content & Extremism;
<END HARMFUL CATEGORIES>

**Refer to the detailed explanations of each category below:**

<BEGIN HARMFUL CATEGORIES EXPLANATIONS>
...
<END HARMFUL CATEGORIES EXPLANATIONS>

**Provide the moderation result in the following JSON format:**

GUARDRAIL = {{
"C1(Sexual Content)": boolean,
"C2(Harassment & Bullying)": boolean,
"C3(Threats, Violence & Harm)": boolean,
"C4(False & Deceptive Information)": boolean,
"C5(Illegal/Regulated Activities)": boolean,
"C6(Hateful Content & Extremism)": boolean
}}

**Set the value to true ONLY for the most relevant categories found in the video. If no harmful content is detected, set all values to false.**

**The moderation result must be the JSON format of "GUARDRAIL", beginning with "GUARDRAIL:"**

Output the thinking process in <think> </think> and final answer in <answer> </answer> tags.
```

Figure 4: Prompt template used during ATPO training and inference on SafeWatch-Bench. Detailed descriptions of harmful categories are omitted for brevity.

C.3 Prior Work

We compare ATPO with recent approaches exploring Reinforcement Learning (RL) and reasoning capabilities in VLMs for video understanding and safety detection. Here we provide a more detailed description of those approaches that we evaluate in Section 4.2.

GuardReasoner-VL. GuardReasoner-VL [20] proposes a reasoning-based VLM guardrail model for multi-modal safety moderation. The model is trained to generate an explicit reasoning process before producing moderation decisions. The model is first initialized with reasoning SFT and then further optimized with online GRPO using a length-aware safety reward. Although the original GuardReasoner-VL is trained and evaluated on text, image, and text-image data rather than video, its backbone model, Qwen2.5-VL-7B-Instruct, supports direct video inputs. Therefore, we evaluate GuardReasoner-VL on video inputs in our experiments for comparison with other models.

VAD-R1. VAD-R1 [10] proposes a reinforcement-learning-based framework for Video Anomaly Reasoning (VAR) that encourages VLMs to perform structured reasoning when analyzing anomalous events in videos. The model is trained using a two-stage pipeline: after initial supervised alignment, the model is optimized using Anomaly Verification Augmented GRPO (AVA-GRPO). AVA-GRPO extends GRPO with an anomaly verification reward that re-evaluates predictions by performing reasoning on temporally trimmed versions of the video. In our experiments, we further train VAD-R1 on the training set of SafeWatch-Bench using AVA-GRPO and report the evaluation results in Section 4.2.

Video-R1 and VideoChat-R1. Video-R1 [5] extends the R1-style rule-based RL paradigm to video understanding by introducing Temporal Group Relative Policy Optimization (T-GRPO), a variant of GRPO designed to encourage general temporal reasoning over video frames. T-GRPO contrasts model performance on temporally ordered versus randomly shuffled frame sequences. VideoChat-R1 [15] investigates reinforcement fine-tuning of video VLMs using GRPO with task-specific spatio-temporal rewards. The training incorporates reward functions tailored to several video perception tasks, including temporal grounding and video question answering, enabling the model to improve spatio-temporal perception while maintaining general dialogue capabilities.

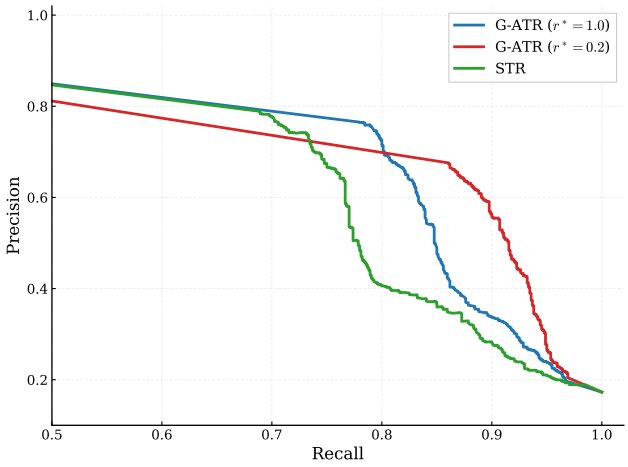


Figure 5: Micro precision-recall curves comparing ATPO (G-ATR) with different target ratios r^* and GRPO (STR). G-ATR with $r^* = 1.0$ improves precision across most recall levels compared with the GRPO model, while a smaller target ratio $r^* = 0.2$ shifts the curve toward higher recall.

SafeWatch-8B. SafeWatch-8B [4] is a video guardrail model built on InternVL2-8B that produces multi-label safety decisions and content-specific explanation conditioned on safety policies. It introduces Parallel Equivalent Policy Encoding (PEPE) to encode policies independently and Policy-Aware Adaptive Pruning (PAP) to retain policy-relevant visual tokens. Its training pipeline comprises multi-task supervised fine-tuning, guardrail-specific fine-tuning with pruning on SafeWatch-Bench, and Direct Preference Optimization (DPO) [29] to improve explanations and reduce false positives. We include SafeWatch-8B as a specialized multi-label video guardrail baseline in Table 1.

All models are evaluated at 1 fps and the same inference prompt to ensure fair comparison.

D Precision-Recall Curves

In Section 4.4, we investigate global precision–recall control using the target ratio defined in ATPO. In this section, we provide the corresponding micro precision–recall curves for models trained with the GRPO (STR) and the proposed ATPO (G-ATR) under different target ratios in Figure 5. Compared with the Jaccard reward (green curve), G-ATR with target ratio $r^* = 1.0$ (blue curve) consistently achieves higher precision across most recall levels, indicating a stronger overall performance. When the target ratio is reduced to $r^* = 0.2$ (red curve), the curve shifts toward higher recall regions, demonstrating that the adaptive reward can steer the model toward more recall-oriented operating points while maintaining comparable precision.

E Effects of Step Size

We investigate the effects of the step size η, which determines how strongly the current log-ratio error contributes to the update of u_{t+1} in Equation 9. Figure 6 shows the evolution of ETR (FN/FP) on a small validation set as training proceeds under different step sizes η. The target ratio is fixed at $r^* = 1.0$ (dashed horizontal line). Overall, we find that **ATPO steers the model toward the target ratio.** Across all four settings, ETR starts above the target, indicating a recall-dominant regime. As training progresses, ATPO drives ETR downward toward the target of 1. This suggests that ATPO effectively behaves as a feedback controller, guiding the model toward the specified target.
η controls controller responsiveness. For larger η (0.2, 0.1), ETR decreases rapidly during early training, which indicates faster convergence and stronger reaction to deviations from the target. However, this stronger control introduces noticeable oscillations. $\eta = 0.2$ exhibits larger fluctuations after approaching the target compared to $\eta = 0.1$. In contrast, for small η (0.05, 0.01), ETR decreases more gradually and convergence takes longer. The resulting trajectories are smoother with dampened oscillations, suggesting better stability of the controller. This indicates that η determines how

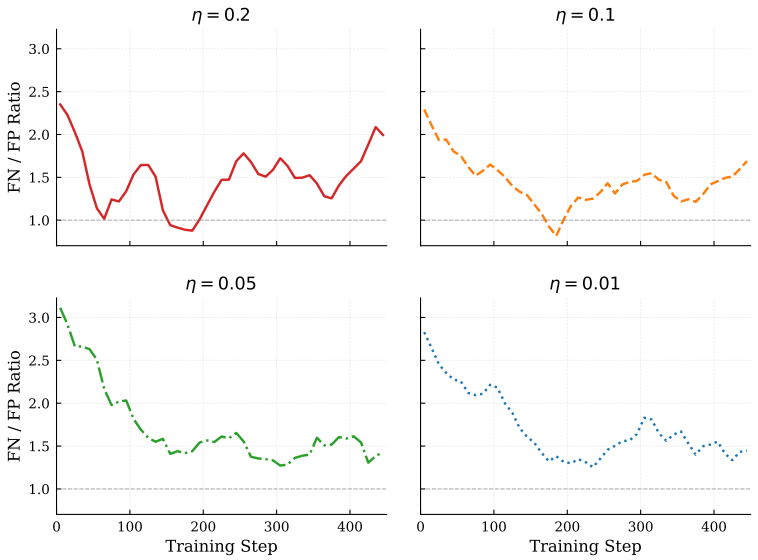


Figure 6: Evolution of Error Type Ratio (ETR=FN/FP) during ATPO training under different step sizes η, with target ratio $r^* = 1.0$ (dashed line).

Table 6: Training comparison on Qwen3-VL-8B. GRPO uses static rewards (Exact Match and STR), whereas ATPO-G and ATPO-C use adaptive rewards G-ATR and C-ATR, respectively. All RL methods start from SFT-1ep. Micro, Macro, and Binary denote F1 scores. **Bold** and underlining indicate the best and second-best results per column.

Training	SafeWatch-Bench-Real			
	Jaccard	Micro	Macro	Binary
Zero-shot	48.69	52.45	40.32	77.76
SFT-1ep *(RL initialization)*	64.70	72.70	71.37	82.35
SFT-3ep *(extended SFT)*	74.14	**78.07**	**77.30**	93.33
RL from the same SFT-1ep checkpoint				
GRPO (EM)	59.95	62.50	56.79	84.33
GRPO (STR)	67.32	68.42	65.61	86.05
ATPO-G	**75.50**	77.08	75.25	**95.22**
ATPO-C	<u>74.63</u>	<u>77.19</u>	<u>76.09</u>	<u>93.79</u>

aggressively the controller reacts to instantaneous error. These effects of η align with the behavior of a step size parameter in learning and feedback control.

F Comparison of Training Objectives on Qwen3-VL-8B

We extend the training objective comparison in Section 4.3 to Qwen3-VL-8B with results on SafeWatch-Bench-Real reported in Table 6. As in the Qwen2.5-VL-7B comparison, all RL methods share the same one-epoch SFT initialization, training data, and GRPO configuration. Consistent with the Qwen2.5-VL-7B results, both ATPO variants outperform the static-reward baselines and their shared initialization across all four metrics. Static-reward RL, however, does not consistently improve over its initialization on this backbone. Compared with extended three-epoch SFT, both ATPO variants achieve higher Jaccard and Binary F1 but slightly lower Micro and Macro F1. Nevertheless, our Qwen2.5-VL-7B experiments show that stronger in-domain performance from extended SFT does not necessarily generalize: extending SFT reduces Jaccard on both SafeWatch-Bench-GenAI (Table 2) and VHD11K-Video-Real [43] (Appendix G).

Table 7: Cross-dataset evaluation of Qwen2.5-VL-7B checkpoints on VHD11K-Video-Real, without target-dataset fine-tuning. Micro, Macro, and Binary denote F1 scores. **Bold** and underlining indicate the best and second-best results per column.

Model / training	VHD11K-Video-Real			
	Jaccard	Micro	Macro	Binary
Zero-shot	54.20	38.49	34.74	63.60
SFT-1ep *(RL initialization)*	63.64	63.12	50.95	80.40
SFT-3ep *(extended SFT)*	57.63	59.29	49.08	76.00
RL from the same SFT-1ep checkpoint				
GRPO (STR)	59.03	58.28	42.17	78.20
ATPO-C	**68.50**	**66.59**	**56.16**	**84.00**

Table 8: Multi-label-only evaluation on SafeWatch-Bench-Real, restricted to videos with at least two ground-truth harmful labels. The SFT, GRPO, and ATPO checkpoints are based on Qwen2.5-VL-7B.

Model / training	Jaccard	Micro	Macro	Binary
SafeWatch-8B	56.43	69.72	60.67	92.90
SFT-1ep	28.66	43.93	31.94	76.12
GRPO (STR)	63.10	73.95	52.19	99.39
ATPO-G-7B	**66.11**	**76.97**	**66.37**	**99.70**

G Distribution-Shift Evaluation

Complementing the SafeWatch-Bench-GenAI evaluation in Table 2, we assess cross-dataset generalization on the 500 real-world videos from VHD11K [43], denoted VHD11K-Video-Real. We exclude the generated-video subset because lower generation quality can make unsafe content ambiguous. Table 7 compares the zero-shot Qwen2.5-VL-7B backbone with its SafeWatch-Bench-trained SFT and RL checkpoints, without further fine-tuning on VHD11K. Both RL methods start from the same SFT-1ep checkpoint.

ATPO improves cross-dataset generalization. Despite improving in-domain performance (Table 2), extended SFT reduces all four metrics on VHD11K-Video-Real, with Jaccard falling from 63.64 to 57.63. GRPO with STR also underperforms SFT-1ep across all four metrics. In contrast, ATPO-C achieves the highest scores across all four metrics, with Jaccard exceeding SFT-1ep and GRPO-STR by 4.86 and 9.47 points, respectively. These results support the benefit of adaptive reward optimization for transfer beyond the training distribution.

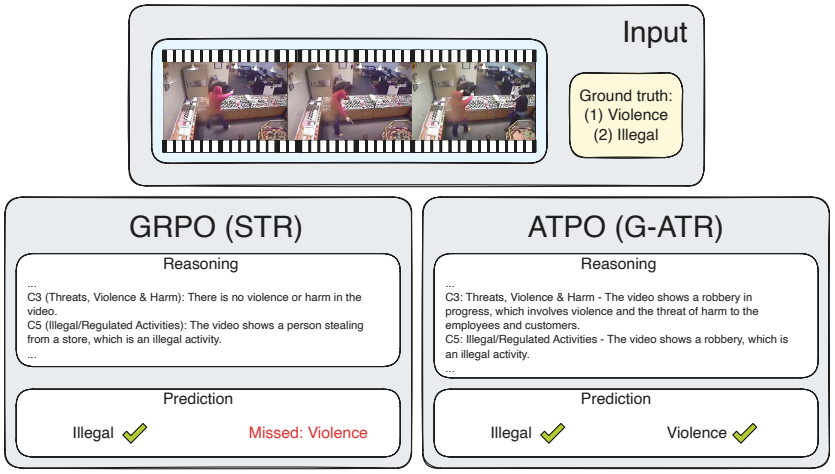
H Evaluation on Multi-Label Examples

To examine whether ATPO's gains extend beyond single-label cases, we restrict evaluation to SafeWatch-Bench-Real videos with at least two ground-truth harmful labels. We compare SafeWatch-8B with the Qwen2.5-VL-7B-based SFT-1ep, GRPO (STR), and ATPO-G checkpoints.

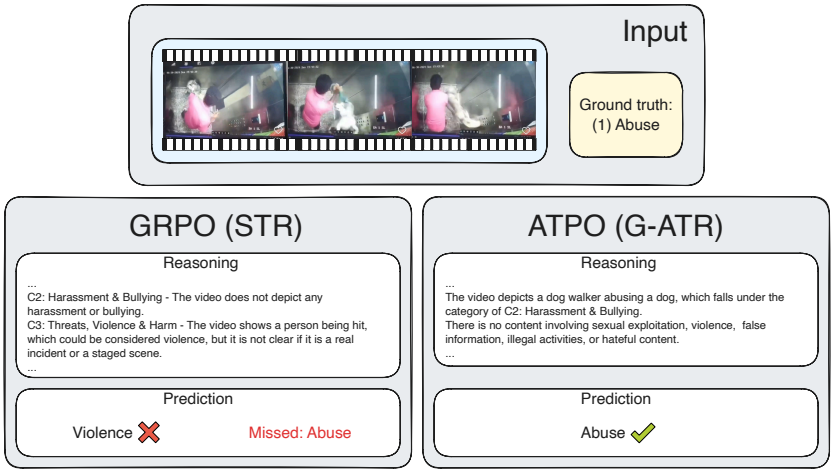
As shown in Table 8, ATPO-G-7B achieves the highest scores across all four metrics, improving Jaccard and Macro F1 over GRPO (STR) by 3.01 and 14.18 points, respectively. These results show that ATPO's improvements persist on examples with multiple harmful categories and are not confined to single-label cases.

I Qualitative Analysis

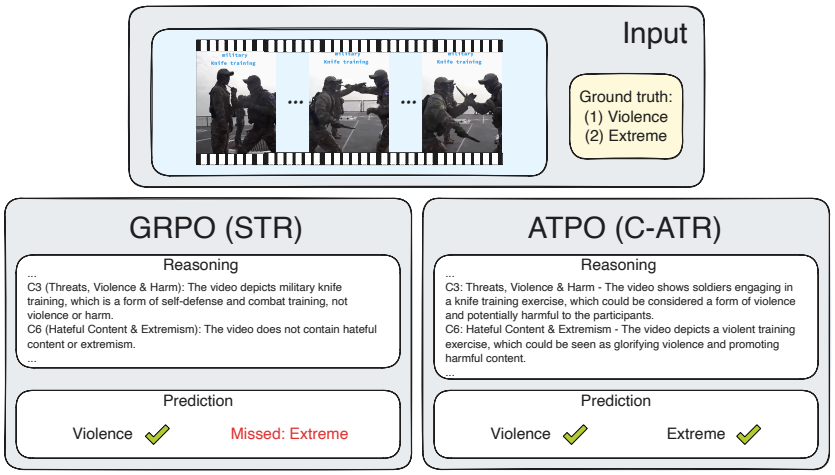
In this section, we provide qualitative comparisons between models trained with GRPO (STR) and our ATPO framework in Figure 7. Across these examples, the GRPO (STR) model tends to focus on a single salient aspect of the scene, which can lead to missed categories or incorrect interpretations, while ATPO-trained models produce more comprehensive reasoning and more accurate predictions. In the first case, the video shows a robbery taking place inside a shop. The individuals are attempting



(a) Case 1: a robbery scene.



(b) Case 2: a dog walker bullies his dog.



(c) Case 3: a military knife-training video.

Figure 7: Qualitative comparison between GRPO with STR and ATPO with Adaptive Tversky Rewards (G-ATR and C-ATR).

Table 9: Complete category-level control results on SafeWatch-Bench-Real. Target ratios are modified for C4 (Misinformation) and C6 (Extreme); the remaining categories are included to examine changes beyond the targeted pair. P and R denote precision and recall.

Model	r_4^*	r_6^*	C1				C2			
			P	R	P/R	ETR	P	R	P/R	ETR
InternVL	–	–	88.46	84.66	1.04	1.39	56.52	18.57	3.04	5.70
Qwen3-VL	–	–	92.31	77.01	1.20	3.58	79.25	44.68	1.77	4.73
ATPO-C	1.0	1.0	84.69	94.15	0.90	0.34	63.30	73.40	0.86	0.63
ATPO-C	5.0	0.2	87.95	77.67	1.13	2.10	62.50	69.15	0.90	0.74
Model	r_4^*	r_6^*	C3				C4 (targeted)			
			P	R	P/R	ETR	P	R	P/R	ETR
InternVL	–	–	56.56	88.46	0.64	0.17	94.44	19.54	4.83	70.00
Qwen3-VL	–	–	55.47	69.34	0.80	0.56	95.83	22.55	4.25	79.00
ATPO-C	1.0	1.0	72.44	76.53	0.95	0.81	55.28	66.67	0.83	0.62
ATPO-C	5.0	0.2	63.91	79.81	0.80	0.45	70.41	67.65	1.04	1.14
Model	r_4^*	r_6^*	C5				C6 (targeted)			
			P	R	P/R	ETR	P	R	P/R	ETR
InternVL	–	–	71.28	87.01	0.82	0.37	94.74	15.79	6.00	96.00
Qwen3-VL	–	–	80.49	57.89	1.39	3.00	96.43	20.93	4.61	102.00
ATPO-C	1.0	1.0	88.68	82.46	1.08	1.67	80.31	79.07	1.02	1.08
ATPO-C	5.0	0.2	84.76	78.07	1.09	1.56	31.77	94.57	0.36	0.03

to break the display glass while pointing a gun at an employee. The GRPO (STR) model focuses on the theft aspect and predicts only the *Illegal* category while overlooking the violence (shooting) involved. ATPO (G-ATR), on the other hand, recognizes both the illegal activity and the violent threat, resulting in correct predictions.

In the second case, the video shows a dog walker abusing a dog in an elevator, which is labeled as *Abuse*. The GRPO (STR) model produces uncertain reasoning and incorrectly predicts *Violence*, missing the correct *Abuse* category. In contrast, ATPO (G-ATR) correctly identifies the behavior as abuse toward the animal and predicts the appropriate category.

In the third case, the video depicts a military knife training scenario labeled as *Violence* and *Extreme* (with a subcategory of *incitement to violence*). The GRPO model interprets the scene as routine training and therefore dismisses the possibility that the content may reflect or promote harmful or extremist activity. In contrast, the ATPO (C-ATR) model produces reasoning that acknowledges the potentially violent nature of the activity and its potential to glorify violence.

These examples suggest that ATPO can encourage more comprehensive reasoning about harmful content within videos, enabling the model to capture multiple unsafe aspects or correctly interpret the nature of the behavior, thereby reducing missed labels in Multi-VSD.

J Complete Category-Level Control Results

Table 9 extends the main text category-level control results in Table 4 to all six categories of SafeWatch-Bench-Real. We report the two ATPO-C configurations with $(r_4^*, r_6^*) = (1.0, 1.0)$ and $(5.0, 0.2)$, alongside the same zero-shot baselines. The modified configuration targets higher precision for C4 (Misinformation) and higher recall for C6 (Extreme).

The targeted categories shift in the intended directions: C4 precision increases from 55.28 to 70.41, while C6 recall increases from 79.07 to 94.57, accompanied by a decrease in C6 precision from 80.31 to 31.77. For the remaining four categories, the target ratios are held fixed, and their achieved precision-recall ratios remain broadly balanced under both configurations. In particular, P/R changes only slightly for C2 (0.86 to 0.90) and C5 (1.08 to 1.09). C1 and C3 show larger fluctuations, from 0.90 to 1.13 and from 0.95 to 0.80, respectively, but remain relatively close to one. These results indicate that C-ATR can steer the targeted categories while broadly maintaining the precision-recall balance of categories whose targets are unchanged.

Table 10: Post-hoc threshold tuning of fixed SFT-1ep and GRPO (STR) checkpoints (left), compared with ATPO-G checkpoints trained using different target ratios r^* (right), on SafeWatch-Bench-Real.

Post-hoc threshold tuning					Training-time control		
τ	SFT-1ep		GRPO (STR)		r^*	ATPO-G	
	Precision	Recall	Precision	Recall		Precision	Recall
0.1	75.76	36.36	72.26	73.58	0.2	66.33	86.55
0.3	77.31	33.45	72.35	73.58	0.5	76.18	80.71
0.5	78.82	32.48	72.35	73.58	1.0	75.74	78.81
0.7	79.06	30.67	72.35	73.58	2.0	75.72	69.05
0.9	80.27	28.61	72.43	73.58	5.0	78.56	67.62

K Comparison with Post-hoc Threshold Tuning

We compare post-hoc threshold tuning of the Qwen2.5-VL-7B-based SFT-1ep and GRPO (STR) checkpoints with ATPO-G checkpoints trained using different global target ratios r^*. The STR checkpoint uses $\alpha = \beta = 1$.

Confidence extraction and thresholding. Given the input $\mathbf{x} = (x, v)$, each baseline first generates a complete response y using greedy decoding. Let t_c denote the token position of the Boolean value for category c. We replay $\mathbf{x}$ and the generated prefix $y_{<t_c}$ to obtain the next-token probabilities of `true` and `false`, and compute

$$s_c = \frac{\pi_\theta(\texttt{true} \mid \mathbf{x}, y_{<t_c})}{\pi_\theta(\texttt{true} \mid \mathbf{x}, y_{<t_c}) + \pi_\theta(\texttt{false} \mid \mathbf{x}, y_{<t_c})}. \tag{14}$$

The prefix includes the model's original reasoning and preceding category decisions. Scores are extracted for all six categories, including those originally predicted as false. Samples are included only when all six category keys appear exactly once and all six scores are valid. Missing scores are not imputed.

Using a single threshold τ shared across categories, we construct the predicted label set

$$P_\tau = \{c : s_c \geq \tau\}.$$

The implementation checks that $P_{0.5} = P$, where P is the label set extracted from the original response. Lower thresholds can add positive labels, while higher thresholds can remove them. The original prefixes $y_{<t_c}$ and scores s_c remain fixed throughout the sweep. Changed predictions are not propagated into subsequent category scores.

We sweep $\tau \in \{0.1, 0.2, \ldots, 0.9\}$ and report five thresholds in Table 10. Precision and recall are micro-averaged over video-category pairs. The sweep characterizes operating points directly on the test split, without validation-set threshold selection.

Comparison of operating points. Across the reported thresholds, SFT-1ep remains in a high-precision, low-recall regime, while GRPO (STR) exhibits little change. ATPO-G obtains operating points not recovered by these threshold settings. At nearly matched precision, ATPO-G with $r^* = 5$ achieves 67.62 recall, compared with 32.48 for SFT-1ep at $\tau = 0.5$ (precision 78.56 versus 78.82). Furthermore, ATPO-G with $r^* = 0.5$ achieves 76.18 precision and 80.71 recall, exceeding all five reported GRPO (STR) threshold points in both metrics. These results support the benefit of training-time adaptive reward optimization beyond the operating-point adjustments obtained by the reported global-threshold settings.

L Cross-Taxonomy Transfer and Low-Resource Adaptation

We examine whether safety knowledge learned on SafeWatch-Bench transfers to XD-Violence, which uses different category definitions. Using Qwen2.5-VL-7B as the backbone, we first evaluate zero-shot transfer of the SafeWatch-Bench-trained ATPO-G-7B checkpoint. We replace the SafeWatch category definitions in the inference prompt with those of XD-Violence, without updating the model parameters. We then adapt this checkpoint for one epoch using G-ATR on a category-stratified

Table 11: Cross-taxonomy transfer and low-resource adaptation on XD-Violence. Example counts refer only to XD-Violence training data. The final row is a separately trained full-data reference.

Initialization	XD-Violence training		XD-Violence test			
	Method	Examples	Jaccard	Micro	Macro	Binary
Backbone	None	0 (0%)	76.54	72.29	62.78	90.21
SafeWatch ATPO-G-7B	None	0 (0%)	80.51	76.83	64.76	94.04
SafeWatch ATPO-G-7B	ATPO	395 (10%)	84.58	81.00	68.00	95.10
Backbone	SFT + ATPO	3,950 (100%)	88.17	85.24	71.95	96.69

random sample of 395 XD-Violence training examples (10%). For reference, we include the XD-Violence-trained ATPO-G-7B from Table 1, obtained through a separate SFT-ATPO pipeline using all 3,950 target training examples.

As shown in Table 11, zero-shot transfer improves all four metrics over the backbone, with Jaccard increasing from 76.54 to 80.51. Adaptation using only 10% of the target training data further improves all four metrics, reaching 84.58 Jaccard, within 3.59 points of the full-data reference. These results indicate that safety knowledge learned on SafeWatch-Bench remains useful under the different category definitions of XD-Violence and supports adaptation with limited target-dataset supervision.

M Selecting Error-Ratio Targets for Deployment

Error-ratio targets specify the desired balance between missed harmful content and unnecessary intervention, complementing the definitions of the safety categories. They are deployment-specific parameters rather than universal constants. Practitioners may specify desired precision and recall, minimum-precision or minimum-recall constraints, or an FN/FP ratio directly. For category c, precision P_c and recall R_c, expressed as proportions, determine the achieved error ratio:

$$r_c = \frac{\mathrm{FN}_c}{\mathrm{FP}_c} = \frac{P_c(1 - R_c)}{R_c(1 - P_c)}. \tag{15}$$

For illustration, a child-facing video service that routes flagged content to human review might require at least 80% recall for violence while maintaining at least 60% precision. This prioritizes reducing missed harmful content while limiting unnecessary review. The boundary operating point $(P_c, R_c) = (0.6, 0.8)$ corresponds to an achieved error ratio of $r_c = 0.375$. These numerical requirements are illustrative rather than industry standards.

The controller target r_c^* need not equal the desired achieved ratio, because its relationship with the resulting precision and recall depends on the model and deployment distribution. Practitioners should therefore train a small set of candidate target configurations and evaluate the resulting checkpoints on held-out labeled data representative of deployment. Selection should check the actual precision and recall constraints, rather than error-ratio matching alone. This procedure applies to both the global target r^* in G-ATR and the category-specific targets r_c^* in C-ATR. The targets guide training; they are not inference-time adjustments to a fixed checkpoint.

N Societal Impacts

The proposed ATPO framework is designed to improve automated detection of unsafe content in videos, thereby assisting human moderators in identifying potentially harmful material across multiple categories, such as violence, illegal activities, misinformation, and extremist content. The system is intended to support research on video safety and contribute to the development of more reliable moderation tools for online platforms. In particular, its controllable precision-recall behavior may help adapt detection systems to different deployment requirements, where some scenarios require higher recall to reduce harmful exposure, while others require higher precision to avoid unnecessary intervention.

However, ATPO is not intended to replace human judgment in high-stakes moderation decisions. Its outputs should be used as supporting signals within broader moderation pipelines rather than as final

decisions. We believe that improving the reliability and controllability of multi-label video safety detection can contribute positively to online safety by helping moderation systems identify unsafe content more effectively and efficiently.

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: Please refer to the abstract, introduction, and conclusion.

   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: Please see Appendix A.

   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: The paper does not include formal theoretical results.

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 a detailed description of our ATPO algorithm in Section 3 and Appendix B, and details of the experimental setup in Section 4.1 and Appendix C.

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: We provide details of the environment in Appendix C and will provide code and scripts upon publication.

   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: We describe datasets, splits, models, evaluation metrics, training procedure, and key hyperparameters in Section 4.1 and Appendix C.

   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: [No]

   Justification: We note that conducting multiple training runs would require substantially greater computational cost. We found that results are consistent across models and datasets (Section 4.2, 4.3, and 4.4).

   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: We provide the information in Appendix C.2.

   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 has been conducted with strict adherence to the NeurIPS Code of Ethics.

   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: Please see Appendix N.

    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 paper does not release new datasets or high-risk models. Experiments are conducted on existing public benchmarks.

    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 use publicly available datasets and models and cite their original sources. We follow their respective licenses and terms of use.

    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 does not introduce or release new assets.

    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 paper does not involve crowdsourcing nor 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: This paper does not involve crowdsourcing nor research with human subjects.

    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: We did not use LLMs in developing the core method in this research.

    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.