IVT-GUARD: ALL-IN-ONE REASONING MODEL FOR AI-GENERATED CONTENT DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

The rapid proliferation of highly realistic AI-Generated Content (AIGC) necessitates robust and interpretable detection mechanisms. However, existing detectors are predominantly confined to single modalities and provide binary outputs without reasoning. While Multimodal Large Language Models (MLLMs) present a promising solution, their development is constrained by the scarcity of multimodal reasoning data and the reasoning–detection optimization dilemma, where explicit reasoning supervision can compromise detection accuracy. To this end, we introduce **IVT-Set**, a comprehensive dataset comprising over 152K diverse image, video, and text samples equipped with multi-granularity Chain-of-Thought (CoT) reasoning trajectories. Based on it, we propose **IVT-Guard**, a pioneering framework for unified and interpretable AIGC detection across image, video, and text modalities. Furthermore, to overcome the aforementioned optimization dilemma, we design a novel three-stage training paradigm: Artifact-Aware Pre-training, Artifact-to-Evidence Supervised Fine-Tuning via artifact-aware injection, and Evidence-Verdict Consistency Group Relative Policy Optimization. Extensive experiments demonstrate that IVT-Guard achieves state-of-the-art detection performance across in-domain, out-of-domain, and cross-dataset settings while delivering faithful reasoning. Code and data will be released.

1 INTRODUCTION

The rapid advancement of generative AI techniques has greatly enriched multimedia content creation. Modern image generation models (Labs, 2025; Zhao et al., 2026; Lee et al., 2026; OpenAI, 2026b), video generation models (OpenAI, 2025; Wu et al., 2025a; Wan et al., 2025; Seedance et al., 2026), multimodal large language models (Bai et al., 2025a; DeepMind, 2026; Anthropic, 2026; OpenAI, 2026a), and large language models (Yang et al., 2025a; Xu et al., 2026; Chen et al., 2026a; Zeng et al., 2026) can generate highly realistic images, temporally coherent videos, and semantically fluent text, substantially blurring the boundary between authentic and synthetic content. In response, substantial progress has been made in AIGC detection. However, most existing detectors (Yan et al., 2025; Internò et al., 2025; Advik Raj Basani, 2025; Zhou et al., 2026; Chen et al., 2026b; Liu et al., 2026) rely on modality-specific classifiers and typically provide only binary predictions, limiting both cross-modal generalization and prediction interpretability.

Recent advances in multimodal large language models (MLLMs) have demonstrated strong cross-modal understanding and reasoning capabilities, providing a promising foundation for unified and interpretable multimodal detection. Despite this progress, two key challenges remain: **1) Existing datasets remain siloed**, with none simultaneously providing broad modality coverage, large-scale data, and multi-granularity Chain-of-Thought (CoT) annotations. As summarized in table 1, benchmarks such as FakeClue (Wen et al., 2025c), ViF-CoT-4K (Li et al., 2026a), and DetectRL-X (Wu et al., 2026) are limited to individual modalities. Although CommGen15 (Zhou et al., 2026), GenBuster-Bench++ (Wen et al., 2025b), and LOKI (Ye et al., 2025) cover multiple modalities, they remain limited in scale and generator diversity, while lacking multi-granularity CoT annotations. These limitations hinder the development of unified and interpretable AIGC detectors. **2) Existing MLLMs struggle to maintain both reasoning capabilities and detection performance.** To analyze this phenomenon, we evaluate six SFT strategies on three MLLMs (GLM-4.6V-Flash, InternVL3.5-8B, and Qwen3-VL-8B-Instruct) across image, video, and text modalities under out-of-distribution (OOD) settings. As shown in fig. 2, answer-only supervision consistently outperforms

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

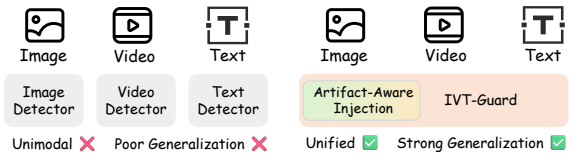


Figure 1: Comparison of modality-specific methods with our unified detector, IVT-Guard.

Table 1: Comparison of existing datasets.

Dataset	Mod.	Scale / Gen.	Annotation	Curation
FakeClue	I	100K / ≥5	CoT	MLLMs
ViF-CoT-4K	V	4.4K / 7	CoT	MLLM+Human
DetectRL-X	T	3.45M / 4	–	LLM+Human
CommGen15	I/V	66.8K / 15	–	–
GenBuster-Bench++	I/V	4K / 20	–	Human
LOKI	I/V/T	18K / 26	CoT	Human
IVT-Set	**I/V/T**	**152K / 44**	**M-CoT**	**MLLMs+Human**

standard CoT supervision across all models and modalities. However, answer-only supervision provides no explicit training signal for structured reasoning. Several common remedies, such as placing the answer first, reweighting the answer-token loss, or shortening the CoT, still result in suboptimal detection performance. We attribute this performance trade-off to an optimization dilemma arising from supervised fine-tuning. Under token-level supervision, lengthy reasoning sequences introduce substantially more language-modeling targets, diluting the supervision signal of the final verdict and biasing optimization toward language generation at the expense of modality-specific forensic cues.

To address the data limitation, we introduce **IVT-Set**, a multimodal dataset comprising 152K samples across image, video, and text modalities, constructed from diverse real-world sources and a wide range of modern generative models. Through well-crafted prompts and strict filtering mechanisms, we curate closely matched real-fake pairs to simulate challenging real-world detection scenarios. Furthermore, we develop an automated annotation pipeline with quality-gated human verification to produce multi-granularity Chain-of-Thought (CoT) trajectories for each sample. Based on this dataset, we develop **IVT-Guard**, a unified framework for interpretable AIGC detection. To resolve the reasoning-detection dilemma in MLLMs and enable reasoning to benefit detection, we propose a three-stage training paradigm: (1) *Artifact-Aware Pre-training (AAP):* We train modality-specific visual encoders with binary classification supervision to capture spatial and temporal artifacts, yielding artifact-aware forensic features. (2) *Artifact-to-Evidence Supervised Fine-Tuning (A2E-SFT):* We introduce an Artifact-Aware Injection mechanism that injects forensic features into the MLLM's visual stream. CoT-supervised fine-tuning then encourages the model to leverage these features for evidence-grounded reasoning. (3) *Evidence-Verdict Consistency Group Relative Policy Optimization (EVC-GRPO):* To further improve detection accuracy and reasoning consistency, we introduce an evidence-verdict consistency reward that rewards correct and logically consistent reasoning while penalizing contradictions. As a result, IVT-Guard consistently achieves strong detection performance across in-domain (ID), out-of-domain (OOD), and cross-dataset (CD) settings, while providing reliable and interpretable evidence for its predictions.

2 Related Work

AIGC Detection Methods. Existing AIGC detection methods are predominantly modality-specific. For images, methods exploit pretrained vision-language representations (Ojha et al., 2023; Koutlis & Papadopoulos, 2024; Liu et al., 2024b), pixel-level artifacts (Tan et al., 2024), semantic-forensic features (Yan et al., 2025; Cheng et al., 2025), and local or distributed patch cues (Zhou et al., 2026; Yang et al., 2026). Video detectors model temporal inconsistencies and dynamics (Chen et al., 2026b; Zheng et al., 2025; Zhang et al., 2025; Song et al., 2024; Li et al., 2026b; Corvi et al., 2025). Text detectors leverage likelihood statistics (Mitchell et al., 2023; Bao et al., 2024; Hans et al., 2024), contrastive or diversity-based features (Guo et al., 2024b; Advik Raj Basani, 2025), and generalization or routing strategies (Fu et al., 2025; Sun et al., 2026).

Multimodal Large Language Models for AIGC Detection. Recent MLLMs (Bai et al., 2025b;a; Team, 2026; Wang et al., 2025; Hong et al., 2026; Yu et al., 2026) provide a promising foundation for interpretable AIGC detection through multimodal reasoning. FakeVLM (Wen et al., 2025c) provides artifact-level explanations, while LEGION (Kang et al., 2025) and FakeXplainer (Ji et al., 2026) incorporate artifact localization and visual grounding. AIGI-Holmes (Zhou et al., 2025) further improves generalization and explainability through multi-expert annotation and preference optimization. BusterX++ (Wen et al., 2025b) and Skyra (Li et al., 2026a) extend multimodal explanations and grounded artifact reasoning to video detection, while VideoVeritas (Tan et al., 2026a) introduces perception-oriented reinforcement learning and Ivy-Fake (Jiang et al., 2026) unifies ex-

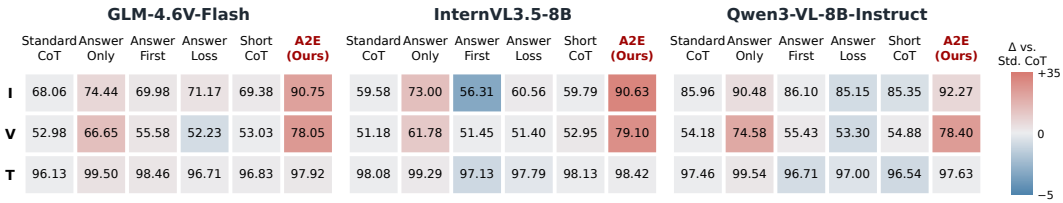


Figure 2: Comparison of OOD accuracy (%) under different SFT strategies for three MLLMs on image (I), video (V), and text (T). Colors indicate accuracy changes relative to the standard CoT.

plainable image-video detection. However, unified modality coverage and the joint optimization of reasoning and detection remain underexplored.

3 Dataset

3.1 Data Collection

Image. We first collect 70K real images from Open Images V7 (Kuznetsova et al., 2020), AnimeDL (Zhu et al., 2025), CelebA-HQ (Karras et al., 2017), CelebA (Liu et al., 2015), and ADE20K (Zhou et al., 2017). Constructing challenging real–fake pairs requires visual diversity while preserving semantic and structural similarity. To this end, we use Q-Insight (Li et al., 2025) for sampling across diverse quality levels, while Qwen2.5-VL-72B provides category labels and captions for category-balanced sampling and fake-image generation, respectively. We retain only image pairs with a CLIP (Radford et al., 2021) semantic similarity above 0.8 and a DINOv2 (Oquab et al., 2024) structural similarity above 0.85. The retained samples are then balanced across resolution, category, and quality score, resulting in approximately 50.5K images. Finally, we incorporate 3.5K GAN-generated images from WildFake (Hong et al., 2025b) to diversify the generation sources.

Video. We sample approximately 60K real videos from OpenVid-1M (Nan et al., 2025) and ActivityNet Captions (Krishna et al., 2017), while balancing resolution, frame rate, and frame count. We filter out low-quality videos using aesthetic scores from the LAION Aesthetic Predictor and remove temporally inconsistent samples based on the CLIP cosine similarity between adjacent frames. For fake-video generation, we feed the first frame and caption of each selected real video into diverse open-source and commercial image-to-video models as joint visual and textual prompts. The generated videos span a wide range of frame rates, durations, and resolutions. To reduce distribution discrepancies between real and fake videos, we proportionally rescale them for spatial alignment and encode all videos in YUV420p using x265 HEVC. We further incorporate 5K fake videos from GenBuster-200K (Wen et al., 2025a) and GenVideo (Chen et al., 2026b), resulting in approximately 49.4K videos in total.

Text. We sample 76K real texts from HC3 (Guo et al., 2023), Newsroom (Grusky et al., 2018), SQuAD (Rajpurkar et al., 2016), WritingPrompts (Fan et al., 2018), Arxiv-10 (Farhangi et al., 2022), and Yelp (Zhang et al., 2015), while balancing word-frequency and domain-label distributions. The corpus spans question answering, news, encyclopedic content, stories, scientific articles, and reviews. We first filter out overly short or invalid samples and then perform embedding-based deduplication and text normalization. For fake-text generation, we employ ten LLMs for question answering, summarization, and rewriting. Finally, we compute the cosine similarity between real and fake texts using INSTRUCTOR (Su et al., 2023) embeddings and retain pairs with high similarity and closely matched word counts, yielding approximately 48.6K samples.

3.2 Data Annotation

To equip each sample with high-quality CoT annotations at multiple granularities, we design a three-stage annotation pipeline, as illustrated in fig. 3. The pipeline consists of the following stages:

CoT Generation. For each sample, we provide its ground-truth label as a guidance signal and prompt Qwen3-VL-235B-A22B-Instruct (Bai et al., 2025a) to generate an initial CoT annotation. Each annotation is structured into four components: *thinking* (step-by-step reasoning), *key features*

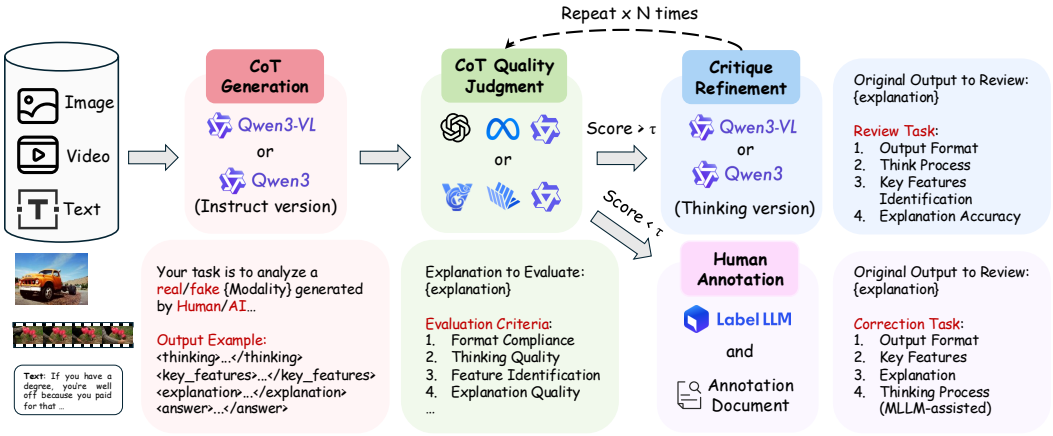


Figure 3: Data annotation pipeline for IVT-Set, consisting of CoT generation, multi-judge quality judgment, and iterative critique–refinement to produce high-quality multi-granularity CoT.

(salient forensic cues), *explanation* (evidence interpretation), and *answer* (final prediction). This multi-granularity CoT design establishes a traceable evidence chain from forensic cues to the final verdict, supporting interpretable and evidence-grounded reasoning.

CoT Quality Judgment. We employ multiple MLLMs or LLMs as independent judges to evaluate annotations across several quality dimensions and compute the average score. Annotations below the review threshold are assigned to human annotators for revision following standardized guidelines, while the others proceed to the Critique–Refinement Loop.

Critique–Refinement Loop. In this stage, we use Qwen3-VL-235B-A22B-Thinking to critique and refine annotations following the quality rubric from the previous stage, and re-evaluate them with the same MLLM/LLM judges. The critique–refinement–evaluation cycle continues until the average score meets the acceptance threshold or three refinement rounds are completed.

4 METHOD

To jointly optimize detection accuracy and reasoning capabilities, we propose a three-stage training paradigm for IVT-Guard, consisting of Artifact-Aware Pre-training (AAP), Artifact-to-Evidence Supervised Fine-Tuning (A2E-SFT), and Evidence-Verdict Consistency Group Relative Policy Optimization (EVC-GRPO), as illustrated in fig. 4.

4.1 ARTIFACT-AWARE PRE-TRAINING

The visual encoders of MLLMs are primarily optimized for high-level semantic understanding and underrepresent subtle generation artifacts. We therefore introduce Artifact-Aware Pre-training with two modality-specific perceptual amplifiers for spatial and temporal artifact perception.

Spatial Perceptual Amplifier. For an image input x_i, we employ the CLIP (Radford et al., 2021) image encoder and adapt its pretrained parameters $\boldsymbol{\theta}_0$ with LoRA modules $\Delta\boldsymbol{\theta}$. The encoder produces a sequence of visual tokens $\mathbf{F}_i$, from which we extract the `[CLS]` token as the spatially aggregated artifact feature $\mathbf{f}_i$:

$$\mathbf{F}_i = \mathcal{E}_{\text{CLIP}}(x_i;\, \boldsymbol{\theta}_0 + \Delta\boldsymbol{\theta}) \in \mathbb{R}^{(N_c+1)\times d_c}, \qquad \mathbf{f}_i = [\mathbf{F}_i]_{0,:}\,. \tag{1}$$

The Spatial Perceptual Amplifier captures both low-level texture irregularities and high-level semantic inconsistencies in images.

Temporal Perceptual Amplifier. Given a video input v_i consisting of T frames $\{v_{i,t}\}_{t=1}^{T}$, we employ the token encoder of the Perception Encoder (PE) (Bolya et al., 2026) to extract frame-level visual tokens. The resulting tokens $\mathbf{H}_{i,t}$ are aggregated by the PE attention pooling module into a frame-level artifact representation for each frame. We then perform mean pooling over these

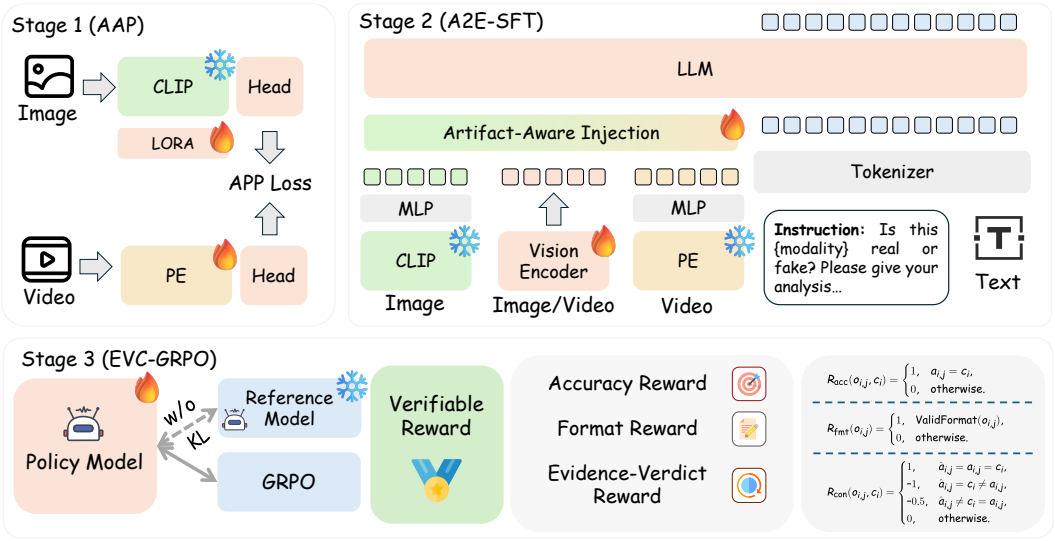


Figure 4: Overview of the three-stage training pipeline for IVT-Guard. Stage 1: AAP enhances sensitivity to modality-specific generation artifacts; Stage 2: A2E-SFT integrates artifact-aware representations into the MLLM for evidence-grounded reasoning; and Stage 3: EVC-GRPO jointly optimizes detection accuracy and evidence–verdict consistency.

representations to obtain the temporally aggregated artifact feature $\mathbf{h}_i$:

$$\mathbf{H}_{i,t} = \mathcal{T}_{\mathrm{PE}}^{\mathrm{tok}}(v_{i,t}) \in \mathbb{R}^{(N_p+1)\times d_p}, \qquad \mathbf{h}_i = \frac{1}{T}\sum_{t=1}^{T} \mathrm{AttnPool}_{\mathrm{PE}}(\mathbf{H}_{i,t})\,. \tag{2}$$

The Temporal Perceptual Amplifier captures frame-level visual irregularities and aggregates generation artifacts across video frames.

We separately introduce binary authenticity supervision for the two modality-specific perceptual amplifiers. Specifically, the spatially aggregated artifact feature $\mathbf{f}_i$ and temporally aggregated artifact feature $\mathbf{h}_i$ are fed into their respective classification heads g_f and g_h to predict authenticity. Both perceptual amplifiers are optimized using the same binary cross-entropy objective:

$$\mathcal{L}_{\mathrm{AAP}} = -\frac{1}{B}\sum_{i=1}^{B}\left[y_i \log \sigma(g(\mathbf{u}_i)) + (1-y_i)\log(1-\sigma(g(\mathbf{u}_i)))\right],\ (\mathbf{u}_i, g) \in \{(\mathbf{f}_i, g_f), (\mathbf{h}_i, g_h)\}, \tag{3}$$

where $\sigma(\cdot)$ denotes the sigmoid function and $y_i \in \{0,1\}$ denotes the ground-truth label. This supervision encourages both perceptual amplifiers to learn representations sensitive to authenticity-related generation artifacts.

4.2 Artifact-to-Evidence Supervised Fine-Tuning

To integrate the learned artifact representations into the MLLM's visual stream, we introduce an *Artifact-Aware Injection* mechanism. It fuses artifact representations with the native visual tokens at complementary granularities, supporting evidence-grounded reasoning over modality-specific generation artifacts.

Image Modality: Global-Local Injection. For an image x_i, the Spatial Perceptual Amplifier produces the spatially aggregated artifact feature $\mathbf{f}_i$ and the local artifact features $\mathbf{F}_i[1:,:]$. We inject these artifact features into the native MLLM visual tokens $V_i^{\mathrm{mllm}} \in \mathbb{R}^{N_i\times d}$ through two complementary pathways. Specifically, the projected global feature is broadcast to all visual tokens to provide holistic image-level artifact context, while the projected local features are selectively incorporated through cross-attention to capture fine-grained local irregularities:

$$\widetilde{\mathbf{V}}_i^{\mathrm{img}} = \mathbf{V}_i^{\mathrm{mllm}} + \phi_{\mathrm{g}}^{\mathrm{img}}(\mathbf{f}_i) + \mathrm{CrossAttn}\left(\mathbf{V}_i^{\mathrm{mllm}}, \phi_{\mathrm{l}}^{\mathrm{img}}(\mathbf{F}_i[1:,:])\right), \tag{4}$$

where $\phi_{\mathrm{g}}^{\mathrm{img}}$ and $\phi_{\mathrm{l}}^{\mathrm{img}}$ denote the global and local MLP projectors, respectively. This global-local injection combines holistic image-level cues with fine-grained local irregularities, providing complementary artifact evidence for subsequent reasoning.

Video Modality: Temporal-Spatial Injection. For a video v_i, the Temporal Perceptual Amplifier yields the temporally aggregated artifact feature $\mathbf{h}_i$ together with frame-specific local artifact features $\mathbf{H}_{i,t}[1:,:]$. The former captures video-level artifact information aggregated across frames, whereas the latter retains spatial irregularities within individual frames. For the t-th frame, we inject both into the native MLLM visual tokens $\mathbf{V}_{i,t}^{\mathrm{mllm}}$:

$$\widetilde{\mathbf{V}}_{i,t}^{\mathrm{vid}} = \mathbf{V}_{i,t}^{\mathrm{mllm}} + \phi_{\mathrm{t}}^{\mathrm{vid}}(\mathbf{h}_i) + \mathrm{CrossAttn}\left(\mathbf{V}_{i,t}^{\mathrm{mllm}}, \phi_{\mathrm{s}}^{\mathrm{vid}}\left(\mathbf{H}_{i,t}[1:,:]\right)\right), \tag{5}$$

where $\phi_{\mathrm{t}}^{\mathrm{vid}}$ and $\phi_{\mathrm{s}}^{\mathrm{vid}}$ denote the temporal and spatial MLP projectors, respectively. The temporal feature is shared across frames to provide consistent video-level artifact context, while cross-attention incorporates frame-specific spatial artifact cues. The enhanced frame-level visual tokens are concatenated to form the final video representation:

$$\widetilde{\mathbf{V}}_{i}^{\mathrm{vid}} = \mathrm{Concat}\left(\widetilde{\mathbf{V}}_{i,1}^{\mathrm{vid}}, \ldots, \widetilde{\mathbf{V}}_{i,T}^{\mathrm{vid}}\right). \tag{6}$$

For text inputs, we use the MLLM's native textual representations. Finally, we jointly fine-tune the MLLM across image, video, and text modalities with CoT supervision. Given a batch of target CoT response sequences $\{Z_i\}_{i=1}^{B}$, where $Z_i = \{z_{i,1}, \ldots, z_{i,L_i}\}$ and L_i denotes the length of the i-th response sequence, we optimize the A2E-SFT objective using autoregressive language modeling:

$$\mathcal{L}_{\text{A2E-SFT}} = -\frac{1}{\sum_{i=1}^{B} L_i} \sum_{i=1}^{B} \sum_{\ell=1}^{L_i} \log P_\theta(z_{i,\ell} \mid z_{i,<\ell}, x_i^{m_i}), \qquad m_i \in \{\mathrm{img}, \mathrm{vid}, \mathrm{txt}\}, \tag{7}$$

where $x_i^{m_i}$ denotes the modality-specific input of sample i. Through supervised fine-tuning, the MLLM learns to translate artifact-aware representations into forensic evidence and perform evidence-grounded reasoning.

4.3 Evidence-Verdict Consistency Group Relative Policy Optimization

In this stage, to reinforce the model's reasoning behavior through outcome-based rewards while improving detection accuracy, we further optimize the A2E-SFT-initialized policy π_θ using Group Relative Policy Optimization (GRPO) (Shao et al., 2024; Guo et al., 2025). For each input-question pair (x_i, Q_i), the current policy samples a group of G responses $\mathcal{O}_i = \{o_{i,j}\}_{j=1}^{G}$, where i indexes the input and j indexes the response within the group. Each response contains a reasoning trace and an explicit verdict $a_{i,j} \in \{\mathrm{real}, \mathrm{fake}\}$. To jointly encourage accurate predictions, valid response structures, and consistency between the generated reasoning and final verdict, we design a composite reward consisting of an accuracy reward, a format reward, and an evidence-verdict consistency reward. Specifically, we first define the accuracy reward and the format reward as follows:

$$R_{\mathrm{acc}}(o_{i,j}, c_i) = \begin{cases} 1, & a_{i,j} = c_i, \\ 0, & \text{otherwise}, \end{cases} \qquad R_{\mathrm{fmt}}(o_{i,j}) = \begin{cases} 1, & \mathrm{ValidFormat}(o_{i,j}), \\ 0, & \text{otherwise}. \end{cases} \tag{8}$$

where $\mathrm{ValidFormat}(o_{i,j})$ indicates whether the response follows the predefined output format, requiring the tags `<thinking>`, `<key_features>`, `<explanation>`, and `<answer>` to appear in the prescribed order.

Evidence-Verdict Consistency Reward. To assess whether the generated reasoning supports the final verdict, we remove the explicit answer and provide the preceding reasoning content, together with the verification prompt p_{v}, to Qwen3-4B-Instruct-2507 (Yang et al., 2025a), which serves as the external LLM judge:

$$\hat{a}_{i,j} = \mathrm{Judge}(p_{\mathrm{v}}, \mathrm{Reason}(o_{i,j})), \qquad \hat{a}_{i,j} \in \{\mathrm{real}, \mathrm{fake}, \mathrm{uncertain}\}. \tag{9}$$

Using the ground-truth label c_i, we define the consistency reward as

$$R_{\mathrm{con}}(o_{i,j}, c_i) = \begin{cases} 1, & \hat{a}_{i,j} = a_{i,j} = c_i, \\ -1, & \hat{a}_{i,j} = c_i \neq a_{i,j}, \\ -0.5, & \hat{a}_{i,j} \neq c_i = a_{i,j},\ \hat{a}_{i,j} \in \{\mathrm{real}, \mathrm{fake}\}, \\ 0, & \text{otherwise}. \end{cases} \tag{10}$$

Table 2: Accuracy comparison (%) on the image modality of IVT-Set and four cross-dataset benchmarks. *Avg.* denotes the mean of ID accuracy, average CD accuracy, and average OOD accuracy. Best and second-best results are shown in **bold** and underlined, respectively.

Method	ID	CD				OOD							Avg.
		Chameleon	LOKI	GenImage	GenBuster++	Real	FLUX.1 Krea	Qwen-Image	Midjourney v6	Infinity	Janus-Pro-7B	LlamaGen	
Image-specific Detectors													
UnivFD (Ojha et al., 2023)	77.75	52.42	67.70	72.97	54.70	82.75	79.88	68.00	73.38	73.62	74.12	87.38	72.24
DIRE (Wang et al., 2023)	89.00	70.92	71.23	75.95	**72.35**	89.75	72.88	55.75	74.75	81.50	71.00	87.25	79.25
NPR (Tan et al., 2024)	84.62	60.15	66.07	75.70	52.90	91.25	61.12	54.62	70.12	81.00	62.62	65.25	72.58
AIDE (Yan et al., 2025)	88.62	61.81	77.74	87.76	60.15	89.25	79.75	74.12	74.25	84.12	80.62	90.00	80.74
DGS-Net (Yan et al., 2026)	84.75	54.64	75.14	84.73	49.35	76.75	78.38	88.00	72.50	86.25	87.88	87.88	77.75
PGC (Zhou et al., 2026)	93.12	67.60	80.78	92.54	54.70	90.25	61.88	83.12	66.00	87.12	92.62	94.75	83.09
Generic MLLMs													
Qwen2.5-VL-7B-Instruct	54.62	63.56	50.60	53.57	51.02	95.50	52.25	50.62	51.62	52.00	52.38	65.75	56.44
Qwen3-VL-8B-Instruct	65.25	78.33	67.92	70.95	66.85	89.50	55.50	53.50	55.25	72.00	66.88	91.12	68.46
InternVL3.5-8B	61.00	62.93	55.70	63.65	52.05	92.75	52.12	54.00	52.25	55.12	59.88	70.00	60.63
MiniCPM-o 4.5	59.50	69.69	58.47	64.60	52.70	88.75	52.50	53.50	52.62	60.50	65.00	80.25	61.87
GLM-4.6V	57.38	63.09	51.95	57.03	49.80	97.25	52.12	50.88	51.38	52.25	53.25	69.25	57.92
Qwen3-VL-235B-A22B-Instruct	67.63	79.17	69.77	72.15	65.90	97.00	57.88	54.75	56.38	69.00	66.88	93.63	70.06
MLLM-based Image Detectors													
FakeVLM (Wen et al., 2025c)	73.50	62.52	**88.00**	**99.39**	55.10	82.50	67.50	73.62	83.38	84.25	77.12	87.75	76.40
UniGenDet (Zhang et al., 2026)	80.25	63.74	81.05	95.36	44.65	86.50	70.50	68.62	73.25	93.12	93.25	93.25	78.03
Veritas (Tan et al., 2026b)	51.00	54.43	44.90	53.24	47.30	93.00	48.75	50.12	49.38	50.38	51.00	51.75	52.44
IVT-Guard (Ours)	**98.75**	**93.32**	87.02	93.37	66.30	**98.75**	**90.63**	**89.25**	**89.63**	**99.13**	**92.88**	**99.38**	**92.66**

The consistency reward is positive only when the judge verdict, the model's final verdict, and the ground-truth label coincide. A larger penalty is assigned when the reasoning supports the ground-truth verdict but the final verdict is incorrect, whereas a smaller penalty is applied when the final verdict is correct but inconsistent with the reasoning. Uncertain judge outputs receive zero consistency reward. The final reward is defined as

$$R_{i,j} = 0.9\, R_{\text{acc}}(o_{i,j}, c_i) + 0.1\, R_{\text{fmt}}(o_{i,j}) + 0.2\, R_{\text{con}}(o_{i,j}, c_i). \tag{11}$$

The policy is optimized using the following EVC-GRPO objective:

$$\mathcal{L}_{\text{EVC-GRPO}} = -\mathbb{E}\left[\frac{1}{G}\sum_{j=1}^{G}\frac{1}{|o_{i,j}|}\sum_{\ell=1}^{|o_{i,j}|}\min\Big(\rho_{i,j,\ell}(\theta)\widehat{A}_{i,j}, \text{clip}(\rho_{i,j,\ell}(\theta), 1-\epsilon, 1+\epsilon)\,\widehat{A}_{i,j}\Big)\right], \tag{12}$$

where $\rho_{i,j,\ell}(\theta)$ denotes the importance ratio for the j-th response of the i-th input between the updated and rollout policies, and $\widehat{A}_{i,j} = \dfrac{R_{i,j} - \text{mean}\big(\{R_{i,k}\}_{k=1}^{G}\big)}{\text{std}\big(\{R_{i,k}\}_{k=1}^{G}\big)}$ denotes its group-relative advantage. We set $\epsilon = 0.2$ and omit the KL-divergence regularization term.

5 Experiments

5.1 Experimental Setup

Implementation details. We train IVT-Guard based on Qwen3-VL-8B-Instruct (Bai et al., 2025a). During training, we uniformly sample eight frames from each video. In AAP, we adapt CLIP ViT-L/14@336px with rank-8 LoRA and fine-tune the last six blocks of PE-Core-L14-336. In A2E-SFT, we freeze the CLIP and PE encoders and fine-tune the MLLM with LoRA, setting the LoRA rank to 128. We use a learning rate of 1×10^{-5} and a global batch size of 128. For EVC-GRPO, we use AdamW (Loshchilov & Hutter, 2019) with a learning rate of 1×10^{-6}, a global batch size of 128, and 8 rollouts per prompt. More training and inference details are provided in Appendix A.1.

Evaluation Protocols. For modality-specific detectors, we retrain each method on IVT-Set following its original training configuration. For generic off-the-shelf MLLMs and IVT-Guard, we use the same prompt. For specialized MLLM-based detectors, we follow their original prompting protocols. We evaluate all methods under in-domain (ID), cross-dataset (CD), and out-of-domain (OOD) settings, reporting accuracy (%). More details are provided in Appendix A.1.1.

5.2 Main Results

Comparison with State-of-the-Art Methods. As shown in tables 2 and 3, IVT-Guard achieves the highest average accuracy on both image and video detection, reaching 92.66% and 86.82%, respectively. It delivers improvements of 9.57% and 13.32% over the strongest modality-specific baselines, PGC and DeMamba. Even the substantially larger Qwen3-VL-235B-A22B-Instruct falls

Table 3: Accuracy comparison (%) on the video modality of IVT-Set and three cross-dataset benchmarks. † denotes samples from four closed-source generators in AIGVDBench (Ma et al., 2026). *Avg.* denotes the mean of ID accuracy, average CD accuracy, and average OOD accuracy. Best and second-best results are shown in **bold** and underlined, respectively.

Method	ID	CD			OOD						*Avg.*
		LOKI	AIGVDBench†	GenBuster++	Real	EasyAnimateV5.1	HunyuanVideo-I2V	Kling 2.1	Veo 3	Vidu Q2	
Video-specific Detectors											
ReStraV (Internò et al., 2025)	69.25	36.64	44.79	52.00	77.25	48.62	67.62	58.63	50.62	54.37	57.75
NSG-VD (Zhang et al., 2025)	50.38	57.46	78.73	50.40	18.75	46.00	45.25	50.12	48.00	47.88	51.75
DeMamba (Chen et al., 2026b)	87.62	79.25	44.22	62.45	88.25	67.00	76.00	63.88	55.88	74.38	73.50
Generic MLLMs											
Qwen2.5-VL-7B-Instruct	50.88	49.21	48.64	58.50	84.25	48.50	47.50	49.25	47.00	48.88	52.41
Qwen3-VL-8B-Instruct	52.12	55.03	51.43	62.85	86.75	49.12	47.88	49.88	48.62	48.00	54.53
InternVL3.5-8B	52.75	50.00	48.78	54.85	78.75	49.12	48.50	48.50	47.38	46.38	52.36
MiniCPM-o 4.5	50.12	60.06	60.91	55.85	74.50	50.88	47.88	49.00	46.75	47.50	53.94
GLM-4.6V	50.25	48.11	43.49	55.10	85.75	49.00	47.62	48.38	47.88	47.25	51.15
Qwen3-VL-235B-A22B-Instruct	52.75	55.30	52.18	61.95	87.25	50.38	48.63	50.75	48.38	49.38	55.01
MLLM-based Video Detectors											
BusterX++ (Wen et al., 2025b)	52.12	66.77	83.81	**83.20**	82.75	46.88	45.62	48.88	47.62	45.75	60.99
Skyra-RL (Li et al., 2026a)	60.00	56.66	80.76	51.40	**97.25**	49.12	57.75	**86.75**	49.12	51.38	62.72
IVT-Guard (Ours)	**92.00**	**89.25**	**88.64**	77.60	85.75	**87.00**	**85.25**	79.63	**73.88**	**88.25**	**86.82**

behind IVT-Guard, suggesting that simply increasing the model size of generic MLLMs is insufficient to improve AIGC detection accuracy. IVT-Guard also surpasses MLLM-based detectors such as UniGenDet and Skyra-RL by 14.63% and 24.10%, respectively, while maintaining strong cross-dataset and OOD generalization across both modalities. These results demonstrate the effectiveness of our three-stage training paradigm in jointly strengthening modality-specific artifact perception, evidence-grounded reasoning, and detection performance. Text detection results are reported in Appendix table 6.

5.3 Ablation Study

Training-stage contributions. As shown in table 4, standard SFT and GRPO yield limited improvements in detection performance, particularly for video. Introducing AAP+A2E-SFT substantially alleviates this bottleneck, improving image and video ID/OOD accuracy by 8.00%/6.31% and 34.38%/24.22% over standard SFT, respectively. These results suggest that generic instruction tuning and reward optimization alone are insufficient to fully exploit forensic evidence, demonstrating the benefit of integrating effective artifact representations into evidence-based reasoning. Notably, vanilla GRPO after A2E-SFT yields only a slight gain in video OOD accuracy. In contrast, EVC-GRPO consistently improves OOD performance across all three modalities, suggesting that when artifact-aware perceptual representations are already sufficiently strong, maintaining evidence–verdict consistency becomes crucial for further gains.

Complementarity of perceptual amplifiers. As shown in table 5, the Spatial Perceptual Amplifier (SPA) and Temporal Perceptual Amplifier (TPA) exhibit clear modality-specific strengths. SPA alone improves image ID/OOD accuracy by 7.00%/5.83%, whereas TPA alone improves video ID/OOD accuracy by 35.38%/23.25%. Combining both amplifiers consistently improves OOD accuracy over either amplifier alone across all three modalities, demonstrating the complementarity of spatial and temporal forensic representations.

Design of Artifact-Aware Injection. In table 5, "Global" denotes aggregated image and video features, while "Patch" denotes local image and frame-level video features. "Probe-attention" instead uses learnable probe queries to fuse artifact representations before injection. The results show that Global-only already achieves strong performance, whereas Patch-only leads to a noticeable drop, particularly for images, where Global+Patch improves ID/OOD accuracy by 5.87%/6.29% over Patch-only. This indicates that aggregated artifact representations provide the primary holistic forensic context, while local or spatial features serve more effectively as complementary fine-grained evidence. Compared with Probe-attention, Global+Patch further improves image and video OOD accuracy by 0.98% and 0.22%, respectively, showing that preserving both holistic and fine-grained artifact cues is more effective than probe-based fusion.

Visual encoder combinations. As shown in fig. 5, regardless of the visual encoder pairing adopted in the AAP stage, the subsequent A2E-SFT consistently achieves higher OOD accuracy than its counterpart without AAP, with gains of 2.62%–6.77% on images and 22.80%–23.53% on videos. These results indicate that AAP provides a backbone-agnostic mechanism for enhancing artifact-

Table 4: Ablation study of the training stages. Results are reported as ID/OOD accuracy (%).

Method	Image	Video	Text
Base	65.25/65.71	52.12/48.70	43.25/45.75
w/o Artifact-Aware Pre-training			
+ SFT	91.00/85.96	57.50/54.18	98.38/97.46
+ SFT + GRPO	89.75/87.15	69.13/63.00	**98.88**/97.71
w/ Artifact-Aware Pre-training			
+ A2E-SFT	**99.00**/92.27	91.88/78.40	98.75/97.63
+ A2E-SFT + GRPO	98.63/92.00	91.25/81.88	98.75/97.87
+ A2E-SFT + EVC-GRPO	98.75/**93.48**	**92.00/82.80**	98.38/**98.00**
Δ vs Base (pp)	**+33.50/+27.77**	**+39.88/+34.10**	**+55.13/+52.25**

Table 5: Ablation study of Perceptual Amplifiers and Artifact-Aware Injection in A2E-SFT.

SPA	TPA	Fusion Setting	Image	Video	Text
✗	✗	None	91.00/85.96	57.50/54.18	98.38/97.46
Perceptual Amplifiers Ablation					
✓	✗	Global+Patch	98.00/91.79	59.13/53.25	97.25/97.21
✗	✓	Global+Patch	91.38/85.75	**92.88**/77.43	98.13/97.42
Artifact-Aware Injection Ablation					
✓	✓	Global-only	98.50/91.71	92.38/78.38	98.13/**97.83**
✓	✓	Patch-only	93.13/85.98	91.38/77.40	98.25/97.71
✓	✓	Probe-attention	98.63/91.29	92.25/78.18	98.25/97.67
✓	✓	Global+Patch	**99.00/92.27**	91.88/**78.40**	**98.75**/97.63

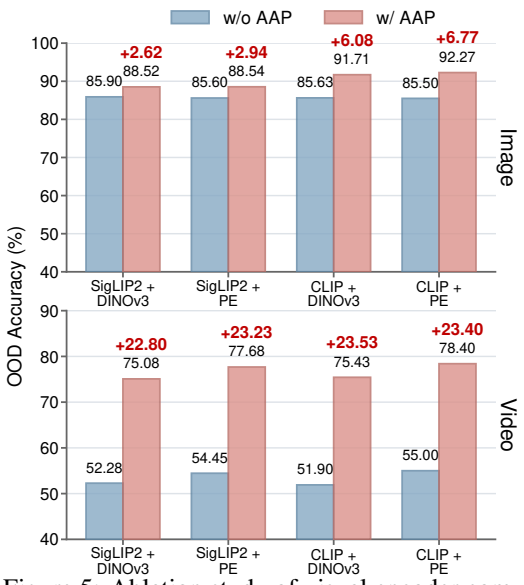


Figure 5: Ablation study of visual encoder combinations in Artifact-Aware Pre-training.

aware visual representations, while preserving the modality-specific strengths of different encoders. Among the combinations, CLIP+PE achieves the best overall detection performance. More experimental results and analyses are provided in Appendix A.2.

6 Conclusion

In this work, we introduced IVT-Set and IVT-Guard for unified and interpretable AIGC detection across image, video, and text. IVT-Set provides large-scale, multi-granularity Chain-of-Thought supervision for evidence-grounded detection reasoning. Building on IVT-Set, IVT-Guard employs a three-stage training paradigm comprising Artifact-Aware Pre-training, Artifact-to-Evidence Supervised Fine-Tuning, and Evidence-Verdict Consistency Group Relative Policy Optimization. These stages respectively enhance modality-specific artifact perception, integrate artifact representations into evidence-grounded reasoning, and enforce consistency between generated evidence and final verdicts, alleviating the reasoning–detection optimization dilemma. This training paradigm mitigates the optimization dilemma between detection accuracy and reasoning quality. Extensive experiments demonstrate that IVT-Guard consistently outperforms modality-specific detectors, generic MLLMs, and specialized MLLM-based detectors across the evaluated modalities and generalization settings, while producing reliable and interpretable explanations.

AI use statement

In this work, we used generative AI tools for synthetic data generation and AI-assisted annotation. Specifically, generative image, video, and text models were employed to construct synthetic samples as part of our dataset construction pipeline. Large language and multimodal language models were further used in the CoT annotation pipeline, including CoT generation, quality judgment, and the critique–refinement loop, as described in the paper. We did not use generative AI tools to formulate research hypotheses or mathematical claims, develop the proposed methodology, design experiments, implement the proposed method, or interpret experimental results; the remaining tasks requiring disclosure are not applicable to this work. The generated samples and annotations were subject to the filtering, quality-control, and human-review procedures specified in our dataset construction pipeline.

Additionally, we used generative AI tools to assist with and polish the writing of this paper, primarily to improve grammar, clarity, conciseness, and academic presentation. All AI-assisted writing was reviewed and revised by the authors to ensure that the final manuscript accurately reflects the

intended methodology, experimental results, and conclusions. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

Reproducibility statement

We provide comprehensive information to facilitate the reproducibility of this work. The construction of IVT-Set, including data sources, synthetic data generation, filtering, annotation, and quality-control procedures, is detailed in the dataset construction section and the appendix. The architecture and three-stage training procedure of IVT-Guard, together with model configurations, optimization settings, and inference details, are described in the method, experimental setup, and appendix. We further specify the dataset splits, evaluation protocols, benchmark compositions, and metric definitions used for the in-domain, out-of-domain, and cross-dataset evaluations. Additional implementation details and ablation studies are provided in the appendix, and anonymized source code for reproducing the proposed method and experiments is included in the supplementary materials. Code, trained checkpoints, data annotations, and dataset construction resources will be publicly released upon publication.

References

Pin-Yu Chen Advik Raj Basani. Diversity boosts AI-generated text detection. In *Data in Generative Models - The Bad, the Ugly, and the Greats*, 2025.

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, et al. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025.

Hailuo AI. Hailuo. `https://hailuoai.video/`, 2025.

Stability AI. Stable diffusion 3.5 large. `https://huggingface.co/stabilityai/stable-diffusion-3.5-large`, 2024.

AI Detector Benchmark. 2026 ai detector benchmark. `https://github.com/mattc95/2026-AI-DETECTOR-BENCHMARK`, 2026.

Anthropic. Claude opus 4.1. `https://www.anthropic.com/news/claude-opus-4-1`, 2025.

Anthropic. Claude opus 5 system card. Technical report, Anthropic, 2026. URL `https://www-cdn.anthropic.com/b514064af1408018e64b1ad24e7d5e75850b4ffd/Claude%20Opus%205%20System%20Card.pdf`.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, Linyi Yang, and Yue Zhang. Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature. In *ICLR*, 2024.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

Daniel Bolya, Po-Yao Huang, Peize Sun, Jang Hyun Cho, Andrea Madotto, Chen Wei, Tengyu Ma, Jiale Zhi, Jathushan Rajasegaran, Hanoona Bangalath, et al. Perception encoder: The best visual embeddings are not at the output of the network. In *NeurIPS*, 2026.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. In *ICLR*, 2019.

Aili Chen, Aonian Li, Baichuan Zhou, Bangwei Gong, Binyang Jiang, Boji Dan, Changqing Yu, Chao Wang, Cheng Ma, Cheng Zhong, et al. The minimax-m2 series: Mini activations unleashing max real-world intelligence. *arXiv preprint arXiv:2605.26494*, 2026a.

Haoxing Chen, Yan Hong, Zizheng Huang, Zhuoer Xu, Zhangxuan Gu, Yaohui Li, Jun Lan, Huijia Zhu, Jianfu Zhang, Weiqiang Wang, et al. Demamba: Ai-generated video detection on million-scale genvideo benchmark. *Science China Information Sciences*, 2026b.

Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *ICLR*, 2024.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025.

Siyuan Cheng, Lingjuan Lyu, Zhenting Wang, Xiangyu Zhang, and Vikash Sehwag. Co-spy: Combining semantic and pixel features to detect synthetic images by ai. In *CVPR*, 2025.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

CompVis. Stable diffusion v1-4 model card. `https://huggingface.co/CompVis/stable-diffusion-v1-4`, 2022.

Riccardo Corvi, Davide Cozzolino, Ekta Prashnani, Shalini De Mello, Koki Nagano, and Luisa Verdoliva. Seeing what matters: Generalizable ai-generated video detection with forensic-oriented augmentation. In *NeurIPS*, 2025.

Junbo Cui, Bokai Xu, Chongyi Wang, Tianyu Yu, Weiyue Sun, Yingjing Xu, Tianran Wang, Zhihui He, Wenshuo Ma, Tianchi Cai, et al. Minicpm-o 4.5: Towards real-time full-duplex omni-modal interaction. *arXiv preprint arXiv:2604.27393*, 2026.

Google DeepMind. Veo 3. `https://deepmind.google/models/veo/`, 2025.

Google DeepMind. Gemini 3.1 pro model card. Technical report, Google DeepMind, 2026. URL `https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-3-1-Pro-Model-Card.pdf`.

Desklib. Desklib ai text detector v1.01. `https://huggingface.co/desklib/ai-text-detector-v1.01`, 2025.

George Drayson, Emine Yilmaz, and Vasileios Lampos. Machine-generated text detection prevents language model collapse. In *EMNLP*, 2025.

Angela Fan, Mike Lewis, and Yann Dauphin. Hierarchical neural story generation. In *ACL*, 2018.

Ashkan Farhangi, Ning Sui, Nan Hua, Haiyan Bai, Arthur Huang, and Zhishan Guo. Protoformer: Embedding prototypes for transformers. In *PAKDD*, 2022.

Jiachen Fu, Chun-Le Guo, and Chongyi Li. Detectanyllm: Towards generalizable and robust detection of machine-generated text across domains and models. In *ACM MM*, 2025.

Yu Gao, Haoyuan Guo, Tuyen Hoang, Weilin Huang, Lu Jiang, Fangyuan Kong, Huixia Li, Jiashi Li, Liang Li, Xiaojie Li, et al. Seedance 1.0: Exploring the boundaries of video generation models. *arXiv preprint arXiv:2506.09113*, 2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Max Grusky, Mor Naaman, and Yoav Artzi. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *NAACL-HLT*, 2018.

Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597*, 2023.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Hanxi Guo, Siyuan Cheng, Xiaolong Jin, Zhuo Zhang, Kaiyuan Zhang, Guanhong Tao, Guangyu Shen, and Xiangyu Zhang. Biscope: Ai-generated text detection by checking memorization of preceding tokens. In *NeurIPS*, 2024a.

Xun Guo, Shan Zhang, Yongxin He, Ting Zhang, Wanquan Feng, Haibin Huang, and Chongyang Ma. Detective: Detecting ai-generated text via multi-level contrastive learning. In *NeurIPS*, 2024b.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, et al. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

Jian Han, Jinlai Liu, Yi Jiang, Bin Yan, Yuqi Zhang, Zehuan Yuan, Bingyue Peng, and Xiaobing Liu. Infinity: Scaling bitwise autoregressive modeling for high-resolution image synthesis. In *CVPR*, 2025.

Abhimanyu Hans, Avi Schwarzschild, Valeriia Cherepanova, Hamid Kazemi, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Spotting llms with binoculars: Zero-shot detection of machine-generated text. *arXiv preprint arXiv:2401.12070*, 2024.

Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, et al. Glm-4.5 v and glm-4.1 v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. *arXiv preprint arXiv:2507.01006*, 2025a.

Wenyi Hong, Xiaotao Gu, Ziyang Pan, Zhen Yang, Yuting Wang, Yue Wang, Yuanchang Yue, Yu Wang, Yanling Wang, Yan Wang, et al. Glm-5v-turbo: Toward a native foundation model for multimodal agents. *arXiv preprint arXiv:2604.26752*, 2026.

Yan Hong, Jianming Feng, Haoxing Chen, Jun Lan, Huijia Zhu, Weiqiang Wang, and Jianfu Zhang. Wildfake: A large-scale and hierarchical dataset for ai-generated images detection. In *AAAI*, 2025b.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Christian Internò, Robert Geirhos, Markus Olhofer, Sunny Liu, Barbara Hammer, and David Klindt. Ai-generated video detection via perceptual straightening. In *NeurIPS*, 2025.

Yikun Ji, Yan Hong, Qi Fan, jun lan, Huijia Zhu, Weiqiang Wang, Liqing Zhang, and Jianfu Zhang. Fakexplain: Ai-generated image detection via human-aligned grounded reasoning. In *ICLR*, 2026.

Changjiang Jiang, Wenhui Dong, Zhonghao Zhang, Fengchang Yu, Wei Peng, Xinbin Yuan, Yifei Bi, Ming Zhao, Zian Zhou, Chenyang Si, et al. Ivy-fake: A unified explainable framework and benchmark for image and video aigc detection. In *ICMR*, 2026.

Hengrui Kang, Siwei Wen, Zichen Wen, Junyan Ye, Weijia Li, Peilin Feng, Baichuan Zhou, Bin Wang, Dahua Lin, Linfeng Zhang, et al. Legion: Learning to ground and explain for synthetic image detection. In *ICCV*, 2025.

Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up gans for text-to-image synthesis. In *CVPR*, 2023.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Christos Koutlis and Symeon Papadopoulos. Leveraging representations from intermediate encoder-blocks for synthetic image detection. In *ECCV*, 2024.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, 2017.

Kuaishou. Kling ai. `https://klingai.kuaishou.com/`, 2025.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, 2023.

Cagliostro Research Lab. Animagine xl 4.0. `https://huggingface.co/cagliostrolab/animagine-xl-4.0`, 2025.

Black Forest Labs. FLUX.2: Frontier Visual Intelligence. `https://bfl.ai/blog/flux-2`, 2025.

Pika labs. `https://pika.art/`, 2022.

Sangwu Lee, Titus Ebbecke, Erwann Millon, Will Beddow, Le Zhuo, Iker García-Ferrero, Liam Esparraguera, Mihai Petrescu, Gian Saß, Gabriel Menezes, and Victor Perez. Flux.1 krea [dev]. `https://github.com/krea-ai/flux-krea`, 2025.

Sangwu Lee, Erwann Millon, Le Zhuo, Matthew Newton, Andrei Filatov, Abhinay Devarinti, Elea Zhong, Avram Djordjevic, Gabriel Menezes, Will Beddow, Titus Ebbecke, Mihai Petrescu, Owen Fahey, Gian Saß, Felix Gil, and Victor Perez. Krea 2. `https://www.krea.ai/blog/krea-2-technical-report`, 2026.

Daiqing Li, Aleks Kamko, Ehsan Akhgari, Ali Sabet, Linmiao Xu, and Suhail Doshi. Playground v2.5: Three insights towards enhancing aesthetic quality in text-to-image generation. *arXiv preprint arXiv:2402.17245*, 2024.

Weiqi Li, Xuanyu Zhang, Shijie Zhao, Yabin Zhang, Junlin Li, Li Zhang, and Jian Zhang. Q-insight: Understanding image quality via visual reinforcement learning. In *NeurIPS*, 2025.

Yifei Li, Wenzhao Zheng, Yanran Zhang, Runze Sun, Yu Zheng, Lei Chen, Jie Zhou, and Jiwen Lu. Skyra: Ai-generated video detection via grounded artifact reasoning. In *CVPR*, 2026a.

Zhengcen Li, Chenyang Jiang, Hang Zhao, Shiyang Zhou, Yunyang Mo, Feng Gao, Fan Yang, Qiben Shan, Shaocong Wu, and Jingyong Su. Preserving forgery artifacts: Ai-generated video detection at native scale. In *ICLR*, 2026b.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024a.

Huan Liu, Zichang Tan, Chuangchuang Tan, Yunchao Wei, Jingdong Wang, and Yao Zhao. Forgery-aware adaptive transformer for generalizable synthetic image detection. In *CVPR*, 2024b.

Zhichen Liu, Kaitong Qin, Linhan He, and Yang Xu. Wavedetect: Robust framework for machine-generated text detection via wavelet transform. In *Findings of ACL*, 2026.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

Long Ma, Zihao Xue, Yan Wang, Zhiyuan Yan, Jin Xu, Xiaorui Jiang, Haiyang Yu, Yong Liao, and Zhen Bi. Your one-stop solution for ai-generated video detection. In *CVPR*, 2026.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *ICML*, 2023.

Kepan Nan, Rui Xie, Penghao Zhou, Tiehan Fan, Zhenheng Yang, Zhijie Chen, Xiang Li, Jian Yang, and Ying Tai. Openvid-1m: A large-scale high-quality dataset for text-to-video generation. In *ICLR*, 2025.

Utkarsh Ojha, Yuheng Li, and Yong Jae Lee. Towards universal fake image detectors that generalize across generative models. In *CVPR*, 2023.

OpenAI. Sora 2 is here. `https://openai.com/index/sora-2/`, 2025.

OpenAI. Gpt-5.6 system card. Technical report, OpenAI, 2026a. URL `https://deploymentsafety.openai.com/gpt-5-6/gpt-5-6.pdf`.

OpenAI. Introducing chatgpt images 2.0. `https://openai.com/index/introducing-chatgpt-images-2-0/`, 2026b.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research Journal*, 2024.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *ICLR*, 2024.

Qi Qin, Le Zhuo, Yi Xin, Ruoyi Du, Zhen Li, Bin Fu, Yiting Lu, Xinyue Li, Dongyang Liu, Xiangyang Zhu, et al. Lumina-image 2.0: A unified and efficient image generative framework. In *ICCV*, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. In *EMNLP*, 2016.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

Team Seedance, De Chen, Liyang Chen, Xin Chen, Ying Chen, Zhuo Chen, Zhuowei Chen, Feng Cheng, Tianheng Cheng, Yufeng Cheng, et al. Seedance 2.0: Advancing video generation for world complexity. *arXiv preprint arXiv:2604.14148*, 2026.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Xiufeng Song, Xiao Guo, Jiache Zhang, Qirui Li, Lei Bai, Xiaoming Liu, Guangtao Zhai, and Xiaohong Liu. On learning multi-modal forgery representation for diffusion generated video detection. In *NeurIPS*, 2024.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A Smith, Luke Zettlemoyer, and Tao Yu. One embedder, any task: Instruction-finetuned text embeddings. In *Findings of ACL*, 2023.

Ke Sun, Guangsheng Bao, Han Cui, and Yue Zhang. Minimizing mismatch risk: A prototype-based routing framework for zero-shot llm-generated text detection. *arXiv preprint arXiv:2602.01240*, 2026.

Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024.

Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, Ping Liu, and Yunchao Wei. Rethinking the up-sampling operations in cnn-based generative network for generalizable deepfake detection. In *CVPR*, 2024.

Hao Tan, Jun Lan, Senyuan Shi, Zichang Tan, Zijian Yu, Huijia Zhu, Weiqiang Wang, Jun Wan, and Zhen Lei. Videoveritas: Ai-generated video detection via perception pretext reinforcement learning. In *ICML*, 2026a.

Hao Tan, Jun Lan, Zichang Tan, Ajian Liu, Chuanbiao Song, Senyuan Shi, Huijia Zhu, Weiqiang Wang, Jun Wan, and Zhen Lei. Veritas: Generalizable deepfake detection via pattern-aware reasoning. In *ICLR*, 2026b.

Ming Tao, Hao Tang, Fei Wu, Xiao-Yuan Jing, Bing-Kun Bao, and Changsheng Xu. Df-gan: A simple and effective baseline for text-to-image synthesis. In *CVPR*, 2022.

Ming Tao, Bing-Kun Bao, Hao Tang, and Changsheng Xu. Galip: Generative adversarial clips for text-to-image synthesis. In *CVPR*, 2023.

Kimi Team, Yifan Bai, Yiping Bao, Y Charles, Cheng Chen, Guanduo Chen, Haiting Chen, Huarong Chen, Jiahao Chen, Ningxin Chen, et al. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*, 2025.

Midjourney Team. Midjourney v6.1. `https://www.midjourney.com/home`, 2024a.

Qwen Team. Qwen3.5: Accelerating productivity with native multimodal agents, 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

Vidu Team. Vidu: Ai video generator. `https://www.vidu.cn/`, 2024b.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. Dire for diffusion-generated image detection. In *ICCV*, 2023.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, et al. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. In *ACL*, 2025.

Haiquan Wen, Yiwei He, Zhenglin Huang, Tianxiao Li, Zihan Yu, Xingru Huang, Lu Qi, Baoyuan Wu, Xiangtai Li, and Guangliang Cheng. Busterx: Mllm-powered ai-generated video forgery detection and explanation. *arXiv preprint arXiv:2505.12620*, 2025a.

Haiquan Wen, Tianxiao Li, Zhenglin Huang, Yiwei He, and Guangliang Cheng. Busterx++: Towards unified cross-modal ai-generated content detection and explanation with mllm. *arXiv preprint arXiv:2507.14632*, 2025b.

Siwei Wen, Peilin Feng, Hengrui Kang, Zichen Wen, Yize Chen, Jiang Wu, Conghui He, Weijia Li, et al. Spot the fake: Large multimodal model-based synthetic image detection with artifact explanation. In *NeurIPS*, 2025c.

Bing Wu, Chang Zou, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, Jack Peng, Jianbing Wu, Jiangfeng Xiong, Jie Jiang, et al. Hunyuanvideo 1.5 technical report. *arXiv preprint arXiv:2511.18870*, 2025a.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025b.

Junchao Wu, Yefeng Liu, Chenyu Zhu, Hao Zhang, Zeyu Wu, Tianqi Shi, Yichao Du, Longyue Wang, Weihua Luo, Jinsong Su, et al. Detectrl-x: Towards reliable multilingual and real-world llm-generated text detection. In *ACL*, 2026.

Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Gongye Liu, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Dynamicrafter: Animating open-domain images with video diffusion priors. In *ECCV*, 2024.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

Jiaqi Xu, Kunzhe Huang, Xinyi Zou, Yunkuo Chen, Bo Liu, Mengli Cheng, Jun Huang, and Xing Shi. Easyanimate: High-performance video generation framework with hybrid windows attention and reward backpropagation. In *ACM MM*, 2025.

Yang Xu, Yu Wang, Hao An, Zhichen Liu, and Yongyuan Li. Detecting subtle differences between human and model languages using spectrum of relative likelihood. In *EMNLP*, 2024.

Jiazhen Yan, Ziqiang Li, Fan Wang, Boyu Wang, Ziwen He, and Zhangjie Fu. Dgs-net: Distillation-guided gradient surgery for clip fine-tuning in ai-generated image detection. In *ICML*, 2026.

Shilin Yan, Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Weidi Xie. A sanity check for ai-generated image detection. In *ICLR*, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Zheng Yang, Ruoxin Chen, Zhiyuan Yan, Ke-Yue Zhang, Xinghe Fu, Shuang Wu, Xiujun Shu, Taiping Yao, Shouhong Ding, Zequn Qin, and Xi Li. All patches matter, more patches better: Enhance AI-generated image detection via panoptic patch learning. In *ICLR*, 2026.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *ICLR*, 2025b.

Junyan Ye, Baichuan Zhou, Zilong Huang, Junan Zhang, Tianyi Bai, Hengrui Kang, Jun He, Honglin Lin, Zihao Wang, Tong Wu, et al. Loki: A comprehensive synthetic data detection benchmark using large multimodal models. In *ICLR*, 2025.

Tianyu Yu, Zefan Wang, Chongyi Wang, Fuwei Huang, Wenshuo Ma, Zhihui He, Tianchi Cai, Weize Chen, Yuxiang Huang, Ranchi Zhao, et al. Minicpm-v 4.5: Cooking efficient mllms via architecture, data, and training recipe. In *CVPR*, 2026.

Aohan Zeng, Xin Lv, Qinkai Zheng, Zhenyu Hou, Bin Chen, Chengxing Xie, Cunxiang Wang, Da Yin, Hao Zeng, Jiajie Zhang, et al. Glm-4.5: Agentic, reasoning, and coding (arc) foundation models. *arXiv preprint arXiv:2508.06471*, 2025.

Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, et al. Glm-5: from vibe coding to agentic engineering. *arXiv preprint arXiv:2602.15763*, 2026.

Shuhai Zhang, ZiHao Lian, Jiahao Yang, Daiyuan Li, Guoxuan Pang, Feng Liu, Bo Han, Shutao Li, and Mingkui Tan. Physics-driven spatiotemporal modeling for ai-generated video detection. In *NeurIPS*, 2025.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *NeurIPS*, 2015.

Yanran Zhang, Wenzhao Zheng, Yifei Li, Bingyao Yu, Yu Zheng, Lei Chen, Jie Zhou, and Jiwen Lu. Unigendet: A unified generative-discriminative framework for co-evolutionary image generation and generated image detection. In *CVPR*, 2026.

Bing Zhao, Chenfei Wu, Deqing Li, Hao Meng, Jiahao Li, Jie Zhang, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kuan Cao, et al. Qwen-image-2.0 technical report. *arXiv preprint arXiv:2605.10730*, 2026.

Chende Zheng, Ruiqi Suo, Chenhao Lin, Zhengyu Zhao, Le Yang, Shuai Liu, Minghui Yang, Cong Wang, and Chao Shen. D3: Training-free ai-generated video detection using second-order features. In *ICCV*, 2025.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *arXiv preprint arXiv:2412.20404*, 2024.

ZhipuAI. Cogview4-6b. `https://huggingface.co/zai-org/CogView4-6B`, 2025.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017.

Xiaoyu Zhou, Jianwei Fei, Peipeng Yu, Jingchang Xie, Chong Cheng, and Zhihua Xia. Pgc: Peak-guided calibration for generalizable ai-generated image detection. In *ICML*, 2026.

Ziyin Zhou, Yunpeng Luo, Yuanchen Wu, Ke Sun, Jiayi Ji, Ke Yan, Shouhong Ding, Xiaoshuai Sun, Yunsheng Wu, and Rongrong Ji. Aigi-holmes: Towards explainable and generalizable ai-generated image detection via multimodal large language models. In *ICCV*, 2025.

Chenyang Zhu, Xing Zhang, Yuyang Sun, Ching-Chun Chang, and Isao Echizen. Animedl-2m: Million-scale ai-generated anime image detection and localization in diffusion era. *arXiv preprint arXiv:2504.11015*, 2025.

Mingjian Zhu, Hanting Chen, Qiangyu Yan, Xudong Huang, Guanyu Lin, Wei Li, Zhijun Tu, Hailin Hu, Jie Hu, and Yunhe Wang. Genimage: A million-scale benchmark for detecting ai-generated image. In *NeurIPS*, 2023.

A APPENDIX

This appendix provides additional details on the implementation and training configurations, supplementary quantitative experiments, the construction of IVT-Set, and Chain-of-Thought annotation prompts. It is organized as follows:

- **A.1. More Implementation Details**
  - A.1.1. Training and Evaluation Datasets
  - A.1.2. Comparison Baselines
  - A.1.3. Training and Inference Details
- **A.2. Additional Quantitative Experiments**
  - A.2.1. Text Modality Detection Performance
  - A.2.2. Reliability of Evidence-Grounded Explanations
  - A.2.3. Cross-Modal Joint Training and Complementarity
  - A.2.4. Robustness under Modality-Specific Perturbations
  - A.2.5. Effectiveness of the Three-Stage Training Paradigm Across MLLMs
- **A.3. Details of the IVT-Set**
  - A.3.1. Generative Methods
  - A.3.2. Detailed Statistics of IVT-Set
- **A.4. Chain-of-Thought Annotation Prompts**

A.1 MORE IMPLEMENTATION DETAILS

A.1.1 TRAINING AND EVALUATION DATASETS

All training and evaluation data cover three modalities: image, video, and text. Training and in-domain (ID) evaluation sets use a balanced 1:1 ratio between real and fake samples. For out-of-domain (OOD) evaluation, each unseen generator is evaluated using 400 fake samples together with a shared set of 400 real samples. OOD metrics are computed separately for each generator and then macro-averaged across unseen generators. Detailed in-domain (ID) and out-of-domain (OOD) splits for each modality are provided in tables 10 to 12.

Training Data. We construct the training set with broad coverage of semantic categories and generator types. The training set contains 45K samples across the three modalities, with 15K samples per modality, including 10K for A2E-SFT and 5K for EVC-GRPO.

Evaluation Data. For each modality, we construct an in-domain (ID) test set containing 400 real and 400 fake samples, where the fake samples are generated by the same generator pool used for training, to evaluate within-distribution performance. To further assess out-of-distribution (OOD) generalization, we construct an additional OOD test set for each modality, consisting of a shared set of 400 real samples and 400 fake samples from each generator unseen during training. We compute OOD metrics for each unseen generator separately and report their macro-average. For cross-dataset (CD) evaluation, we use four image benchmarks (Chameleon (Yan et al., 2025), LOKI (Ye et al., 2025), GenImage (Zhu et al., 2023), and GenBuster++ (Wen et al., 2025b)), three video benchmarks (LOKI, AIGVDBench (Ma et al., 2026), and GenBuster++), and four text benchmarks (LOKI, BiScope (Guo et al., 2024a), DetectRL-X (Wu et al., 2026), and AI Detector Bench (AI Detector Benchmark)).

A.1.2 COMPARISON BASELINES

We compare IVT-Guard with modality-specific detectors, generic MLLMs/LLMs, and MLLM/LLM-based detectors.

For image detection, modality-specific baselines include UnivFD (Ojha et al., 2023), DIRE (Wang et al., 2023), NPR (Tan et al., 2024), AIDE (Yan et al., 2025), DGS-Net (Yan et al., 2026), and PGC (Zhou et al., 2026). MLLM-based detectors include FakeVLM (Wen et al., 2025c), UniGenDet (Zhang et al., 2026), and Veritas (Tan et al., 2026b). Generic MLLM baselines include Qwen2.5-VL-7B-Instruct (Bai et al., 2025b), Qwen3-VL-8B-Instruct (Bai et al., 2025a), InternVL3.5-8B (Wang et al., 2025), MiniCPM-o 4.5 (Cui et al., 2026), GLM-4.6V (Hong et al., 2025a), and Qwen3-VL-235B-A22B-Instruct (Bai et al., 2025a).

For video detection, modality-specific baselines include ReStraV (Internò et al., 2025), NSG-VD (Zhang et al., 2025), and DeMamba (Chen et al., 2026b). MLLM-based detectors include BusterX++ (Wen et al., 2025b) and Skyra-RL (Li et al., 2026a). We use the same generic MLLM baselines as in image detection.

For text detection, modality-specific baselines include ModernBERT (Warner et al., 2025; Drayson et al., 2025) and Desklib AI (Desklib, 2025). Generic LLM/MLLM baselines include Qwen3-VL-8B-Instruct, InternVL3.5-8B, gpt-oss-20b (Agarwal et al., 2025), Qwen3-30B-A3B (Yang et al., 2025a), and DeepSeek-V4-Flash (Xu et al., 2026). LLM-based detectors include DivEye (Advik Raj Basani, 2025), FourierGPT (Xu et al., 2024), DetectAnyLLM (Fu et al., 2025), and WaveDetect (Liu et al., 2026).

A.1.3 Training and Inference Details

For AAP, the Spatial Perceptual Amplifier (SPA) is initialized from CLIP ViT-L/14@336px and adapted with rank-8 LoRA ($\alpha = 16$, dropout 0.05), while the Temporal Perceptual Amplifier (TPA) is initialized from PE-Core-L14-336 and fine-tunes its last six blocks together with the output pooler and binary classification head. We uniformly sample eight frames from each video. SPA and TPA are trained for 20 and 3 epochs with global batch sizes of 32 and 16, respectively. The learning rate is set to 1×10^{-4} for SPA. For TPA, we use 5×10^{-6} for the PE encoder and 5×10^{-5} for the classification head. Both branches use cosine learning-rate decay.

For A2E-SFT, we initialize the MLLM from Qwen3-VL-8B-Instruct and freeze the CLIP and PE encoders. We train the modality-specific MLP projectors and cross-attention modules used for artifact-aware injection, together with all-linear LoRA adapters of the MLLM. Specifically, the image branch optimizes the global and local projectors, while the video branch optimizes the temporal and spatial projectors. The LoRA rank and scaling factor are set to 128 and 256, respectively. We train for 3 epochs with a learning rate of 1×10^{-5} and a global batch size of 128.

For EVC-GRPO, we initialize the policy from the merged A2E-SFT checkpoint and keep the auxiliary CLIP and PE encoders frozen. We jointly optimize the Qwen3-VL policy together with the modality-specific MLP projectors and cross-attention modules used for artifact-aware injection. We train for 1 epoch with a learning rate of 1×10^{-6} and a global batch size of 128. The PPO mini-batch size is 32, and eight rollouts are sampled per prompt. The reward combines accuracy, format compliance, and evidence–verdict consistency with weights 0.9, 0.1, and 0.2, respectively, without KL regularization.

For inference, we deploy IVT-Guard with vLLM (Kwon et al., 2023). For image and video inputs, the frozen auxiliary encoders extract artifact representations, which are incorporated into the MLLM visual tokens through the same artifact-aware injection mechanism used during training. We use greedy decoding with temperature 0, top-$k = 1$, top-$p = 1.0$, a repetition penalty of 1.0, and a maximum generation length of 4,096 tokens.

A.2 Additional Quantitative Experiments

A.2.1 Text Modality Detection Performance

As shown in table 6, IVT-Guard achieves the highest overall average accuracy of 97.29% on text detection, outperforming the strongest text-specific baseline, WaveDetect, by 3.50 percentage points. Generic MLLMs and LLMs achieve overall averages below 58%, suggesting that general-purpose language modeling capability alone is insufficient for reliable text authenticity detection. IVT-Guard also surpasses LLM-based text detectors such as DetectAnyLLM by 13.45 percentage points, while achieving 95.67% average CD accuracy and 97.82% average OOD accuracy. These results demonstrate the effectiveness of IVT-Guard in jointly achieving accurate text authenticity detection and strong generalization across datasets and unseen generators.

A.2.2 Reliability of Evidence-Grounded Explanations

As shown in table 7, IVT-Guard achieves the highest explanation scores and Elo ratings across all three modalities on the OOD test set. Compared with the strongest baseline in each modality, it improves the automatically evaluated explanation score by 0.60–1.28 points and the human-evaluated

Table 6: Performance comparison (Acc, %) on the text modality of IVT-Set and three cross-dataset benchmarks. Results are reported under in-domain (ID), cross-dataset (CD), and out-of-domain (OOD) settings. *Avg.* denotes the average of the ID accuracy, the mean CD accuracy, and the mean OOD accuracy. Best results are in **bold** and the second-best are underlined.

Method	ID	CD				OOD				*Avg.*
		LOKI	BiScope	DetectRL-X	AI Detector Bench	Real	Claude Opus 4.1	Gemini 2.5 Pro	GPT-4o	
Text-specific Detectors										
ModernBERT (Warner et al., 2025; Drayson et al., 2025)	51.75	78.16	63.00	54.62	62.10	89.00	64.38	62.88	66.25	62.28
Desklib AI (Desklib, 2025)	79.75	93.75	**96.61**	84.28	83.40	91.00	93.50	91.62	91.88	87.09
Generic MLLMs & LLMs										
Qwen3-VL-8B	43.25	51.81	42.14	50.93	50.70	69.50	47.62	45.38	44.25	47.94
InternVL3.5-8B	42.50	47.82	42.36	60.78	59.10	76.50	48.25	47.75	47.50	50.00
gpt-oss-20b	44.38	66.46	68.94	49.64	55.90	44.75	48.75	45.00	46.12	50.26
Qwen3-30B-A3B	36.88	63.76	61.51	52.75	51.60	48.25	41.75	40.75	40.25	45.68
DeepSeek-V4-Flash	48.75	50.83	57.63	58.90	60.00	76.00	63.60	61.16	61.80	57.08
LLM-based Text Detectors										
DivEye (Advik Raj Basani, 2025)	71.25	54.38	57.33	60.53	58.80	69.00	67.38	61.88	64.75	64.92
FourierGPT (Xu et al., 2024)	54.00	62.54	55.62	56.07	54.80	47.75	48.88	52.00	52.88	53.88
DetectAnyLLM (Fu et al., 2025)	83.75	63.46	62.53	93.82	89.40	**99.75**	87.75	84.62	89.75	83.84
WaveDetect (Liu et al., 2026)	94.25	93.07	95.18	**95.26**	87.00	92.25	96.00	94.12	95.62	93.79
IVT-Guard (Ours)	**98.38**	**94.97**	94.42	94.39	**98.90**	97.25	**97.88**	**97.75**	**98.38**	**97.29**

Table 7: Explanation reliability evaluation on the OOD test set. We report the absolute score evaluated by Qwen3.5-122B-A10B and the pairwise Elo rating obtained from human evaluation for each modality, with all methods initialized at an Elo rating of 1,000.

Method	**Image**		**Video**		**Text**	
	Score	Elo	Score	Elo	Score	Elo
GLM-4.6V	3.09	1036	2.91	964	3.33	1132
Qwen3-VL-235B-A22B	3.88	1352	3.06	1024	3.68	1294
Qwen3-VL-8B-Instruct	3.91	1364	2.96	984	3.41	1164
IVT-Guard (Ours)	4.70	1676	4.34	1564	4.28	1712

Table 8: Cross-modal evaluation and modality ablation at the A2E-SFT stage. Results are reported as ID/OOD accuracy and F1 (%). Best results are in **bold**.

Training Data	**ID**			**OOD**		
	Image (Acc/F1)	Video (Acc/F1)	Text (Acc/F1)	Image (Acc/F1)	Video (Acc/F1)	Text (Acc/F1)
Image only	98.00/98.31	51.38/42.62	47.00/36.45	91.13/91.03	48.45/39.05	48.21/36.95
Video only	68.00/67.54	91.25/91.25	45.38/39.43	66.73/63.79	75.73/75.14	46.54/39.80
Text only	56.25/51.55	49.88/46.38	88.63/88.62	55.56/50.97	49.05/46.77	88.33/88.30
Image + Video	98.75/98.75	**92.63/92.62**	46.00/39.73	91.58/91.52	77.23/76.63	46.88/39.25
Video + Text	71.63/71.11	91.75/91.75	97.75/97.75	70.90/68.34	**78.65/78.16**	96.04/96.12
Image + Text	98.63/98.75	50.50/38.80	98.13/98.12	92.13/92.09	48.98/37.02	96.96/96.96
All	**99.00/99.06**	91.88/91.87	**98.75/98.75**	**92.27/92.23**	78.40/77.86	**97.63/97.62**

Elo rating by 312, 540, and 418 points for image, video, and text, respectively. These results demonstrate that IVT-Guard provides higher-quality and more reliable evidence-grounded explanations across modalities, beyond producing accurate authenticity predictions.

A.2.3 Cross-Modal Joint Training and Complementarity

As shown in table 8, single-modality training exhibits a clear diagonal pattern: each model performs strongly on its training modality but transfers poorly to the other two modalities. In contrast, joint training consistently benefits from complementary forensic cues. Training on all three modalities achieves 99.00%/92.27%, 91.88%/78.40%, and 98.75%/97.63% ID/OOD accuracy on image, video, and text, respectively, improving OOD accuracy over the corresponding single-modality models by 1.14, 2.67, and 9.30 percentage points. The gains are also asymmetric across modality combinations: adding image data to text training improves text OOD accuracy from 88.33% to 96.96%, whereas adding text data to image training increases image OOD accuracy only from 91.13% to 92.13%. Overall, these results show that image, video, and text data can be jointly trained within a unified model with little cross-modal interference, while benefiting from complementary forensic information across modalities.

Table 9: Robustness evaluation of IVT-Guard under modality-specific input perturbations. Results are reported as accuracy/F1 (%) under in-domain (ID) and out-of-domain (OOD) settings for image, video, and text modalities.

Method	Perturbation	ID			OOD		
		Image (Acc/F1)	Video (Acc/F1)	Text (Acc/F1)	Image (Acc/F1)	Video (Acc/F1)	Text (Acc/F1)
All	None	98.75/98.75	92.00/91.99	98.38/98.37	93.48/93.43	82.80/82.71	98.00/98.00
Image	JPEG Compression (QF=90)	98.38/98.37	92.00/91.99	98.38/98.37	93.69/93.64	82.80/82.71	98.00/98.00
	Gaussian Blur ($\sigma = 1.0$)	96.63/96.62	92.00/91.99	98.38/98.37	92.69/92.65	82.80/82.71	98.00/98.00
Video	Shuffle frame	98.75/98.75	92.25/92.24	98.38/98.37	93.48/93.43	82.55/82.45	98.00/98.00
	Resize $\times 0.7$	98.75/98.75	89.13/89.06	98.38/98.37	93.48/93.43	81.93/81.87	98.00/98.00
Text	Human-like AI text	98.75/98.75	92.00/91.99	98.38/98.37	93.48/93.43	82.80/82.71	98.08/98.08
	Adversarial prompt	98.75/98.75	92.00/91.99	98.25/98.25	93.48/93.43	82.80/82.71	97.38/97.37

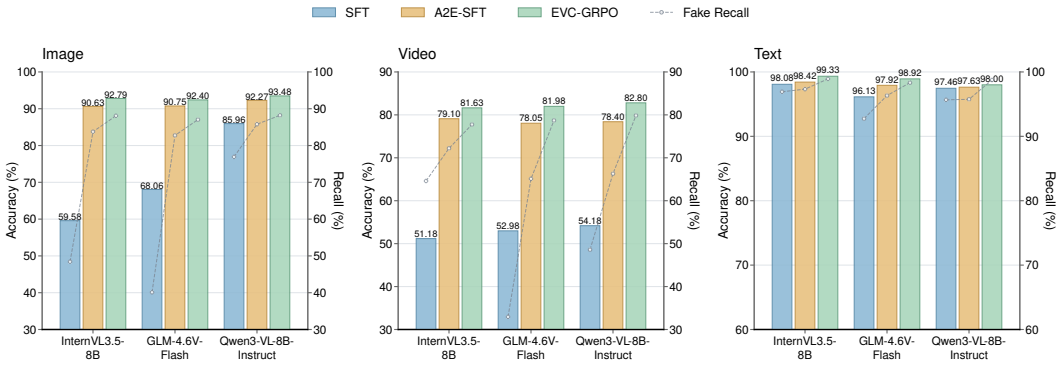


Figure 6: OOD accuracy across image, video, and text modalities for three model families at different training stages. Bars denote OOD accuracy (%), while lines denote mean recall (%) across OOD generators, with "fake" treated as the positive category.

A.2.4 Robustness under Modality-Specific Perturbations

As shown in table 9, IVT-Guard maintains strong performance under a range of modality-specific input perturbations. Most perturbations cause only minor changes in both ID and OOD accuracy, with the largest degradation remaining below 3 percentage points. F1 scores follow trends consistent with accuracy across all settings. These results demonstrate that IVT-Guard is robust to common perturbations across image, video, and text modalities.

A.2.5 Effectiveness of the Three-Stage Training Paradigm Across MLLMs

As shown in fig. 6, the improvements introduced by our training paradigm generalize consistently across different MLLM backbones. A2E-SFT yields the largest gains, substantially improving OOD accuracy on both image and video detection. For example, InternVL3.5-8B improves from 59.58% to 90.63% on images and from 51.18% to 79.10% on videos, while Qwen3-VL-8B-Instruct gains 6.31 and 24.22 percentage points, respectively. EVC-GRPO further improves all three backbones, contributing an additional 1.21–2.16 points on images and 2.53–4.40 points on videos. Text detection starts from a substantially stronger SFT baseline, yet still benefits consistently from the subsequent stages. Notably, after A2E-SFT and EVC-GRPO, the final OOD accuracies across the three backbones differ by only 1.08 points on images and 1.17 points on videos, despite their much larger gaps under standard SFT. Together with the consistently increasing fake recall across training stages, these results demonstrate that our three-stage paradigm provides robust gains across different MLLM backbones and substantially reduces sensitivity to the choice of base model when generalizing to unseen synthetic content.

A.3 Details of the IVT-Set

The overall construction pipeline of IVT-Set is illustrated in fig. 7. For each modality, we collect diverse real-world data across multiple sources and semantic categories, construct corresponding

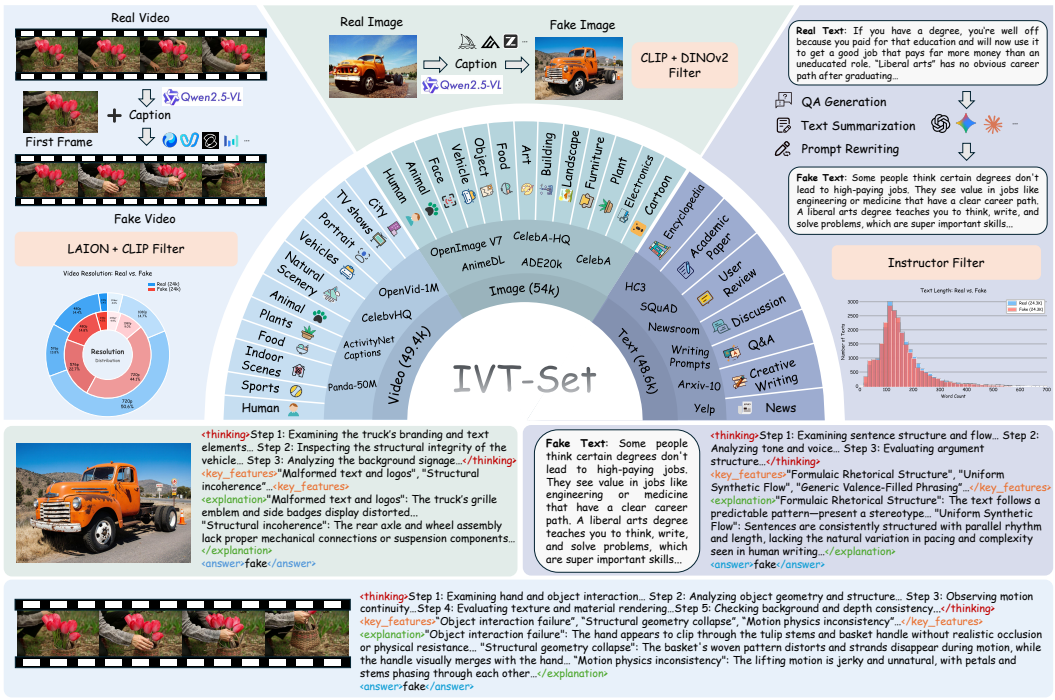


Figure 7: Overview of the IVT-Set construction pipeline, including modality-specific data pairing and filtering, diverse source and category coverage, and multi-granularity CoT annotations.

synthetic samples using a broad range of generative models, and apply modality-specific filtering to ensure data quality and reduce trivial real–fake discrepancies. This process results in a large-scale multimodal benchmark comprising 54K images, 49.4K videos, and 48.6K texts. We further annotate each sample with a unified multi-granularity reasoning structure, consisting of `thinking`, `key_features`, `explanation`, and `answer`, to connect modality-specific forensic cues with the final authenticity verdict. We next describe the construction and statistics of each modality in detail.

A.3.1 Generative Methods

To construct a comprehensive and challenging benchmark, we collect AI-generated samples from a diverse set of contemporary generators across image, video, and text modalities. The selected generators span multiple architectural paradigms, release periods, and access types (open-source vs. commercial) to reduce bias toward specific generator families. Generators with available split annotations are divided into in-domain (ID) and out-of-domain (OOD) splits to support the evaluation of cross-generator generalization.

Image. As shown in table 10, the image subset contains 28,820 generated images from 20 generators spanning GAN, diffusion/flow-based, and autoregressive paradigms, including a closed-source system whose architecture is not publicly disclosed. (1) GAN-based models include early large-scale class-conditional generation with BigGAN (Brock et al., 2019), style-disentangled synthesis with StyleGAN (Karras et al., 2019), and text-guided GANs such as DF-GAN (Tao et al., 2022), GALIP (Tao et al., 2023), and the scalable GigaGAN (Kang et al., 2023). (2) Diffusion/flow-based models cover the evolution of modern generators, from Stable Diffusion v1.4 (CompVis, 2022) to more recent models such as CogView4-6B (ZhipuAI, 2025), Lumina-Image 2.0 (Qin et al., 2025), Qwen-Image (Wu et al., 2025b), and FLUX.1 Krea [dev] (Lee et al., 2025). These models vary substantially in architectural design, model scale, training data, and supported resolutions. (3) Closed-source systems with architectures that are not publicly disclosed include Midjourney V6 (Team, 2024a). (4) Autoregressive models include LlamaGen (Sun et al., 2024), which extends LLM-style next-token prediction to image synthesis, Infinity (Han et al., 2025), which adopts bitwise-parallel generation, and the unified vision-language model Janus-Pro-7B (Chen et al., 2025). All autoregres-

Table 10: Details of the image modality generation methods in IVT-Set. The dataset includes 20 generators spanning GAN, diffusion/flow-based, and autoregressive paradigms, including a closed-source system whose architecture is not publicly disclosed. Generators are split into in-domain (ID) and out-of-domain (OOD) to evaluate cross-generator generalization. Image sizes are reported as width × height in pixels. For generators with mixed-resolution samples, we report up to the four most common image sizes in descending order.

Method	Model Type	Resolution	Data Scale	Data Split
BigGAN (Brock et al., 2019)	GAN	200×200 / 128×128	742	ID
StyleGAN (Karras et al., 2019)	GAN	200×200	1,155	ID
DF-GAN (Tao et al., 2022)	GAN	256×256	198	ID
GALIP (Tao et al., 2023)	GAN	224×224	994	ID
GigaGAN (Kang et al., 2023)	GAN	512×512	497	ID
Stable Diffusion v1.4 (CompVis, 2022)	Diffusion	1024×704 / 1024×768 704×1024 / 768×1024	767	ID
SDXL 1.0 Base (Podell et al., 2024)	Diffusion	1024×704 / 1024×768 768×1024 / 704×1024	1,127	ID
PixArt-α (Chen et al., 2024)	Diffusion	1024×704 / 1024×768 704×1024 / 768×1024	881	ID
Playground v2.5 (Li et al., 2024)	Diffusion	1024×704 / 1024×768 704×1024 / 768×1024	1,202	ID
Stable Diffusion 3.5 Large (AI, 2024)	Diffusion	1024×704 / 1024×768 704×1024 / 768×1024	1,903	ID
Stable Diffusion 2.1 (Rombach et al., 2022)	Diffusion	1024×704 / 1024×768 704×1024 / 768×1024	1,038	ID
Animagine XL 4.0 (Lab, 2025)	Diffusion	704×1024 / 704×960 576×768 / 640×896	235	ID
CogView4-6B (ZhipuAI, 2025)	Diffusion	1024×704 / 1024×768 704×1024 / 768×1024	2,584	ID
Lumina-Image 2.0 (Qin et al., 2025)	Diffusion	1024×704 / 1024×768 704×1024 / 768×1024	2,139	ID
Midjourney V6 (Team, 2024a)	Closed-source	1024×771 / 771×1024 1024×1024 / 1024×573	2,565	OOD
FLUX.1 Krea [dev] (Lee et al., 2025)	Diffusion	1024×704 / 1024×768 704×1024 / 768×1024	2,368	OOD
Qwen-Image (Wu et al., 2025b)	Diffusion	1024×704 / 1024×768 704×1024 / 768×1024	2,525	OOD
LlamaGen (Sun et al., 2024)	Autoregressive	512×512	497	OOD
Infinity (Han et al., 2025)	Autoregressive	512×512	2,654	OOD
Janus-Pro-7B (Chen et al., 2025)	Autoregressive	384×384	2,749	OOD
Total: 20 generators			**28,820**	

sive generators are assigned to the OOD split, yielding a challenging cross-model generalization setting.

Video. As shown in table 11, the video subset comprises 24,735 generated videos from 14 generators, with resolutions ranging from 240p to 1080p and frame rates from 6 to 60 FPS. The collection covers early diffusion-based video models such as SVD (Blattmann et al., 2023) and DynamiCrafter (Xing et al., 2024), intermediate open-source systems including OpenSora (Zheng et al., 2024) and CogVideoX (Yang et al., 2025b), and more recent high-fidelity generators such as HunyuanVideo-I2V (Kong et al., 2024), Wan 2.2 (Wan et al., 2025), Seedance 1.0 (Gao et al., 2025), and Veo 3 (DeepMind, 2025). We also include commercial platforms (Pika (labs, 2022), Hailuo 02 (AI, 2025), Kling 2.1 (Kuaishou, 2025), and Vidu Q2 (Team, 2024b)) that are widely used in real-world content creation, further enhancing the diversity and practical relevance of the benchmark. For generators inherited from existing public benchmarks, we did not perform any split annotations. For newly collected models, we assign ID/OOD labels to support systematic cross-source evaluation.

Text. As shown in table 12, the text subset comprises 24,336 machine-generated texts produced by 10 representative LLMs. The selected models cover both open-source systems (Llama-3.3-70B-Instruct (Grattafiori et al., 2024), gpt-oss-120b (Agarwal et al., 2025), GLM-4.5-Air (Zeng et al., 2025), DeepSeek-V3.1 (Liu et al., 2024a), Kimi K2-Instruct-0905 (Team et al., 2025), Qwen3-235B-A22B-Instruct-2507 (Yang et al., 2025a), and Qwen3-Next-80B-A3B-Instruct (Yang et al., 2025a)) and leading closed-source models (GPT-4o (Hurst et al., 2024), Claude Opus 4.1 (Anthropic, 2025),

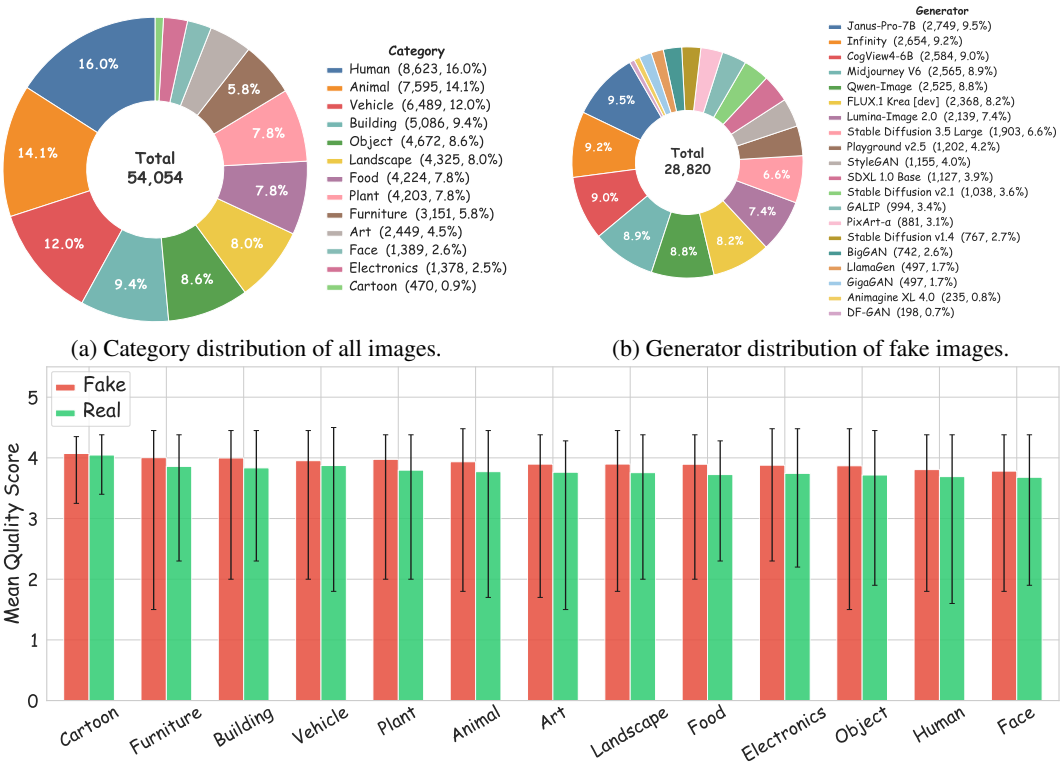
Table 11: Details of the video modality generation methods in IVT-Set. The dataset includes 14 generators with diverse resolutions and frame rates. Generators are split into in-domain (ID) and out-of-domain (OOD) categories when applicable to evaluate cross-generator generalization. Video resolutions are reported as output heights in pixels. For generators with mixed-resolution samples, we report up to the four most common output heights in descending order.

Method	Duration (s)	Resolution	FPS	Data Scale	Data Split
Pika (labs, 2022)	3.00–4.00	768p / 576p 640p / 1024p	12–24	1,000	–
SVD (Blattmann et al., 2023)	3.57	576p	7	1,000	–
DynamiCrafter (Xing et al., 2024)	2.00	576p	8	1,000	–
OpenSora (Zheng et al., 2024)	2.00	256p / 512p	8	1,000	–
CogVideoX (Yang et al., 2025b)	5.04	1024p	24	1,000	–
LTX-Video (HaCohen et al., 2024)	1.37–10.25	672p / 480p 512p / 240p	6–60	4,999	ID
Hailuo 02 (AI, 2025)	5.88–10.12	720p / 1080p 512p / 480p	24	988	ID
Wan 2.2 (Wan et al., 2025)	1.92–9.92	704p / 512p 480p / 928p	6–60	3,999	ID
Seedance 1.0 (Gao et al., 2025)	5.04	704p / 1080p 512p / 480p	24	990	ID
HunyuanVideo-I2V (Kong et al., 2024)	1.93–5.57	720p / 768p 448p / 416p	23–60	2,000	OOD
Vidu Q2 (Team, 2024b)	2.08–8.08	720p / 1080p 512p / 480p	24	990	OOD
EasyAnimateV5.1 (Xu et al., 2025)	0.80–6.00	576p / 512p 480p / 240p	7.5–60	3,998	OOD
Kling 2.1 (Kuaishou, 2025)	5.04–10.04	720p / 512p 480p / 960p	24	965	OOD
Veo 3 (DeepMind, 2025)	4.00	720p / 512p 480p / 240p	24	806	OOD
Total: 14 generators				**24,735**	

Table 12: Details of the text modality generation methods in IVT-Set. The dataset includes 10 representative LLMs covering both open-source and closed-source models. Models are split into in-domain (ID) and out-of-domain (OOD) categories to evaluate cross-model generalization. R./F. Ratio denotes the total real-text length divided by the total corresponding generated-text length. Length reports the minimum–maximum generated-text length in words.

Method	R./F. Ratio	Length	Data Scale	Data Split
Llama-3.3-70B-Instruct (Grattafiori et al., 2024)	1.09	20–1,190	1,486	ID
gpt-oss-120b (Agarwal et al., 2025)	0.99	27–1,056	2,635	ID
GLM-4.5-Air (Zeng et al., 2025)	1.04	20–1,116	2,469	ID
DeepSeek-V3.1 (Liu et al., 2024a)	1.09	20–1,516	2,473	ID
Kimi K2-Instruct-0905 (Team et al., 2025)	1.05	20–787	2,190	ID
Qwen3-235B-A22B-Instruct-2507 (Yang et al., 2025a)	0.99	21–1,206	2,680	ID
Qwen3-Next-80B-A3B-Instruct (Yang et al., 2025a)	1.04	20–1,270	2,451	ID
GPT-4o (Hurst et al., 2024)	0.99	20–1,098	2,500	OOD
Claude Opus 4.1 (Anthropic, 2025)	0.98	20–1,528	2,730	OOD
Gemini 2.5 Pro (Comanici et al., 2025)	1.03	24–1,033	2,722	OOD
Total: 10 LLMs			**24,336**	

and Gemini 2.5 Pro (Comanici et al., 2025)). These models vary substantially in parameter scale, training strategy, and generation style, with fake text lengths ranging from 20 to 1,528 words. The near-unity real-to-fake length ratios (0.98–1.09) limit overall length discrepancies between real and fake texts, reducing the risk that detectors exploit text length as a superficial cue. Each generator contributes 1,486–2,730 samples, ensuring broad coverage without the benchmark being dominated by a single model. We designate three closed-source models as OOD to evaluate whether detectors trained on open-source LLM outputs can generalize to unseen proprietary systems.

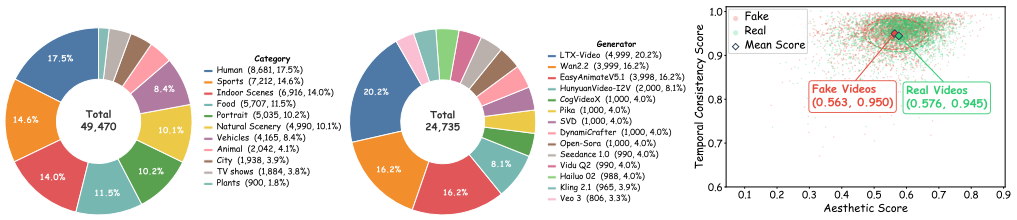


(a) Category distribution of all images.

(b) Generator distribution of fake images.

(c) Comparison of the mean quality scores of fake and real images in each category.

Figure 8: Data statistics of the image modality in IVT-Set. (a) Category distribution of all images across 13 semantic categories. (b) Generator distribution of fake images across 20 models spanning GAN, Diffusion, and Autoregressive paradigms. (c) Comparison of the mean quality scores of fake and real images in each category, showing that the quality scores of the two are very close across all categories.



(a) Category distribution of all videos.

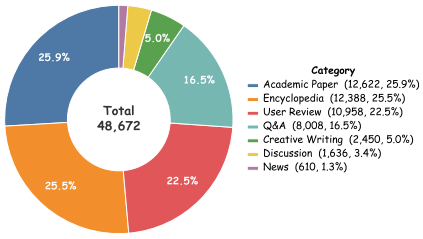
(b) Generator distribution of fake videos.

(c) Correlation between aesthetic score and temporal consistency score.

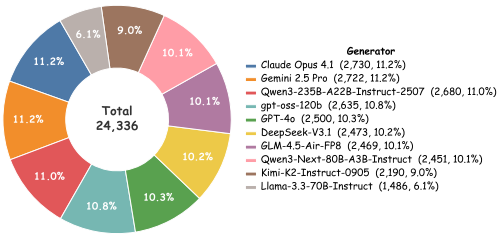
Figure 9: Data statistics of the video modality in IVT-Set. (a) Category distribution of all videos across 11 semantic categories. (b) Generator distribution of fake videos across 14 generation models, covering both open-source and commercial models. (c) Correlation between aesthetic score and temporal consistency score for fake and real videos, showing that the two distributions largely overlap.

A.3.2 Detailed Statistics of IVT-Set

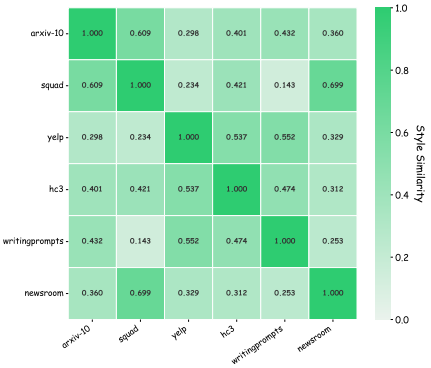
Image. The image subset contains 54,054 samples, including 25,234 real images and 28,820 fake images. It covers 13 broad semantic categories (fig. 8a), ranging from common object categories (e.g., animals and vehicle) to more challenging scenarios (e.g., landscape and art). The category distribution is centered on subjects commonly encountered in real-world images and also includes



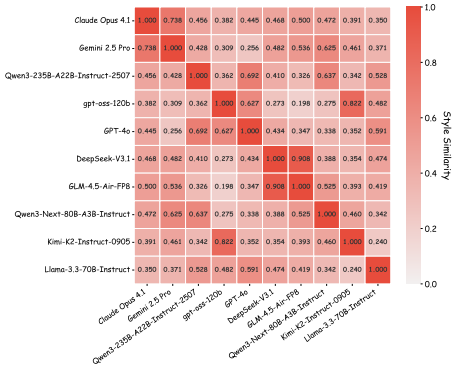
(a) Category distribution of all texts.



(b) Generator distribution of fake texts.



(c) Stylometric similarity among real sources.



(d) Stylometric similarity among generators.

Figure 10: Data statistics of the text modality in IVT-Set. (a) Category distribution across 7 domains. (b) Generator distribution of fake texts from 10 LLMs. (c) Stylometric similarity among real-text sources. (d) Stylometric similarity among fake-text generators. For (c) and (d), each source is represented by a category-averaged profile of 20 rule-based stylometric features. Pairwise similarity is computed as the Pearson correlation between standardized source profiles and rescaled to $[0, 1]$, with higher values indicating more similar profiles.

different scenes and visual styles. As shown in fig. 8b, fake images are broadly distributed across different generators, with the largest single source accounting for only 9.5% of the subset, thereby reducing potential generator-specific bias. Real and fake images also exhibit closely matched mean quality scores across all categories (fig. 8c), with a maximum difference of only 0.18 on a five-point scale. Meanwhile, both groups contain images spanning a wide range of quality levels. Overall, the image subset combines comparable quality between real and fake samples with substantial quality diversity within each group, encouraging detectors to learn forgery-related evidence rather than overall visual quality.

Video. The video subset contains 49,470 samples, evenly divided into 24,735 real videos and 24,735 fake videos. It spans 11 semantic categories (fig. 9a), covering human activities, indoor and outdoor scenes, and a variety of subjects commonly found in real-world video content. The fake videos are generated by 14 models, including both open-source models and commercial platforms, and cover a range of generation architectures, strategies, and quality levels (fig. 9b). Each generator contributes at least 806 videos, ensuring sufficient representation across different sources. We further characterize video quality from two complementary aspects: aesthetic quality and temporal consistency. As shown in fig. 9c, real and fake videos have very similar mean score pairs, with $(0.576, 0.945)$ for real videos and $(0.563, 0.950)$ for fake videos, respectively. Overall, the video subset combines diverse semantic content and generator sources with closely matched quality between real and fake videos, providing a balanced basis for evaluating video forgery detection.

Text. The text subset contains 48,672 samples, evenly divided into 24,336 real texts and 24,336 fake texts. It spans seven writing domains (fig. 10a), covering factual and academic writing, conversational exchanges, subjective reviews, creative writing, and news reporting. Within each domain, real and fake texts are represented by exactly the same number of samples, maintaining a consistent domain composition between the two groups. The fake texts are produced by 10 LLMs, with each

Prompt for CoT Generation (Real Image)

<image> **This is a real image captured by a camera or created by a human artist.**
Task:
1. Carefully inspect the given image and think about the visual cues that indicate it is captured by a camera or created by a human artist.
2. Based on your thinking, group these visual cues into specific categories of real-image authentic features.
3. For each category, provide a brief explanation describing how the features appears in the image.

Output Format:
1. The thinking process is enclosed within `<thinking> </thinking>` tags.
2. The key categories of real-image authentic features are enclosed within `<key_features> </key_features>` tags.
3. The brief explanation of each category is enclosed within `<explanation> </explanation>` tags.

Output Example:
`<thinking>`
Step 1: [What aspect is being examined]. [Detailed observations...]
Step 2: [What aspect is being examined]. [Detailed observations...]
...
`</thinking>`
`<key_features>`"[Authentic features 1]", "[Authentic features 2]", ...`</key_features>`
`<explanation>`
"[Authentic features 1]": [How this authentic features appears in the image]
"[Authentic features 2]": [How this authentic features appears in the image]
...
`</explanation>`

Requirements:
1. Strictly follow the specified format and do not output anything irrelevant.
2. Ensure the thinking process shows clear step-by-step analysis.
3. Avoid redundant or vague explanations. Focus on the most prominent authentic features.
4. Avoid including any AI-generation features related content in the key categories and explanations (e.g., artifacts, imperfections).
5. Key categories must use definitive terms indicating authenticity, NOT neutral descriptive terms.
6. Do not use bold or markdown formatting in the answer.

Figure 11: Prompt template for CoT Generation on **real images**.

model accounting for 6.1% to 11.2% of the fake subset (fig. 10b). This relatively even distribution provides broad coverage of different LLM output styles without allowing any single model to dominate the subset. The pairwise stylometric similarity matrices further reveal distinct cross-source patterns in both real and fake texts (figs. 10c and 10d), demonstrating the stylistic diversity of the subset. Overall, the text subset combines domain-level balance, broad coverage of LLM sources, and source-level stylistic diversity, supporting the evaluation of model-agnostic linguistic evidence across writing domains and generators.

A.4 Chain-of-Thought Annotation Prompts

To generate high-quality chain-of-thought (CoT) annotations, we use modality- and label-specific prompts at different stages. In the CoT Generation stage, the prompt templates (figs. 11 to 16) guide the MLLM to generate an initial CoT annotation according to the task requirements. In the CoT Quality Judgment stage, a quality-judgment prompt (fig. 17) is used with multiple MLLMs or LLMs to score each annotation along multiple dimensions. In the Critique–Refinement Loop, dedicated prompt templates (figs. 18 to 23) critique and refine the current CoT annotation. The revised annotation is then re-evaluated by the same MLLM/LLM judges. This loop continues until the average score reaches the acceptance threshold or three refinement rounds have been completed.

Prompt for CoT Generation (Fake Image)

<image> **This is a fake image generated by AI.**
Task:
1. Carefully inspect the given image and think about the visual cues that indicate it is AI-generated.
2. Based on your thinking, group these visual cues into specific categories of AI-generation features.
3. For each category, provide a brief explanation describing how the feature appears in the image.

Output Format:
1. The thinking process is enclosed within `<thinking> </thinking>` tags.
2. The key categories of AI-generation features are enclosed within `<key_features> </key_features>` tags.
3. The brief explanation of each category is enclosed within `<explanation> </explanation>` tags.

Output Example:
`<thinking>`
Step 1: [What aspect is being examined]. [Detailed observations...]
Step 2: [What aspect is being examined]. [Detailed observations...]
...
`</thinking>`
`<key_features>`"[AI-generation features 1]", "[AI-generation features 2]", ...`</key_features>`
<explanation>
"[AI-generation features 1]": [How this artifact manifests in the image]
"[AI-generation features 2]": [How this artifact manifests in the image]
...
`</explanation>`

Requirements:
1. Strictly follow the specified format and do not output anything irrelevant.
2. Ensure the thinking process shows clear step-by-step analysis.
3. Avoid redundant or vague explanations. Focus on the most prominent AI-generation features.
4. Do not include any authentic or real-world features in key categories and explanations.
5. Key categories must use definitive terms indicating AI-generation artifacts, NOT neutral descriptive terms.
6. Do not use bold or markdown formatting in the answer.

Figure 12: Prompt template for CoT Generation on **fake images**.

Prompt for CoT Generation (Real Video)

<video> **This is a real video captured by physical camera equipment.**
Task:
1. Carefully inspect the given video and think about the visual cues that indicate it is captured by physical camera equipment.
2. Based on your thinking, group these visual cues into specific categories of real-video authentic features.
3. For each category, provide a brief explanation describing how the features appears in the video.

Output Format:
1. The thinking process is enclosed within `<thinking> </thinking>` tags.
2. The key categories of real-video authentic features are enclosed within `<key_features> </key_features>` tags.
3. The brief explanation of each category is enclosed within `<explanation> </explanation>` tags.

Output Example:
`<thinking>`
Step 1: [What aspect is being examined]. [Detailed observations...]
Step 2: [What aspect is being examined]. [Detailed observations...]
...
`</thinking>`
`<key_features>`"[Authentic features 1]", "[Authentic features 2]", ...`</key_features>`
`<explanation>`
"[Authentic features 1]": [How this authentic features appears in the video]
"[Authentic features 2]": [How this authentic features appears in the video]
...
`</explanation>`

Requirements:
1. Strictly follow the specified format and do not output anything irrelevant.
2. Ensure the thinking process shows clear step-by-step analysis.
3. Avoid redundant or vague explanations. Focus on the most prominent authentic features.
4. Avoid including any AI-generation features related content in the key categories and explanations (e.g., artifacts, imperfections).
5. Key categories must use definitive terms indicating authenticity, NOT neutral descriptive terms.
6. Do not use bold or markdown formatting in the answer.

Figure 13: Prompt template for CoT Generation on real videos.

Prompt for CoT Generation (Fake Video)

<video> **This is a fake video generated by AI.**
Task:
1. Carefully inspect the given video and think about the visual cues that indicate it is AI-generated.
2. Based on your thinking, group these visual cues into specific categories of AI-generation features.
3. For each category, provide a brief explanation describing how the feature appears in the video.

Output Format:
1. The thinking process is enclosed within `<thinking> </thinking>` tags.
2. The key categories of AI-generation features are enclosed within `<key_features> </key_features>` tags.
3. The brief explanation of each category is enclosed within `<explanation> </explanation>` tags.

Output Example:
`<thinking>`
Step 1: [What aspect is being examined]. [Detailed observations...]
Step 2: [What aspect is being examined]. [Detailed observations...]
...
`</thinking>`
`<key_features>`"[AI-generation features 1]", "[AI-generation features 2]", ...`</key_features>`
`<explanation>`
"[AI-generation features 1]": [How this artifact manifests in the video]
"[AI-generation features 2]": [How this artifact manifests in the video]
...
`</explanation>`

Requirements:
1. Strictly follow the specified format and do not output anything irrelevant.
2. Ensure the thinking process shows clear step-by-step analysis.
3. Avoid redundant or vague explanations. Focus on the most prominent AI-generation features.
4. Do not include any authentic or real-world features in key categories and explanations.
5. Key categories must use definitive terms indicating AI-generation artifacts, NOT neutral descriptive terms.
6. Do not use bold or markdown formatting in the answer.

Figure 14: Prompt template for CoT Generation on **fake videos**.

Prompt for CoT Generation (Real Text)

This is a real text written by human: {real_text}
Task:
1. Carefully inspect the given text and think about the cues that indicate it is written by a human.
2. Based on your thinking, group these cues into specific categories of authentic features.
3. For each category, provide a brief explanation describing how the features appears in the text.

Output Format:
1. The thinking process is enclosed within `<thinking> </thinking>` tags.
2. The key categories of authentic features are enclosed within `<key_features> </key_features>` tags.
3. The brief explanation of each category is enclosed within `<explanation> </explanation>` tags.

Output Example:
`<thinking>`
Step 1: [What aspect is being examined]. [Detailed observations...]
Step 2: [What aspect is being examined]. [Detailed observations...]
...
`</thinking>`
`<key_features>`"[Authentic features 1]", "[Authentic features 2]", ...`</key_features>`
`<explanation>`
"[Authentic features 1]": [How this authentic features appears in the text]
"[Authentic features 2]": [How this authentic features appears in the text]
...
`</explanation>`

Requirements:
1. Strictly follow the specified format and do not output anything irrelevant.
2. Ensure the thinking process shows clear step-by-step analysis.
3. Avoid redundant or vague explanations. Focus on the most prominent authentic features.
4. Avoid including any AI-generation features related content in the key categories and explanations.
5. Key categories must use definitive terms indicating authenticity, NOT neutral descriptive terms.
6. Do not use bold or markdown formatting in the answer.

Figure 15: Prompt template for CoT Generation on **real texts**.

Prompt for CoT Generation (Fake Text)

This is a fake text generated by AI: {fake_text}
Task:
1. Carefully inspect the given text and think about the visual cues that indicate it is AI-generated.
2. Based on your thinking, group these visual cues into specific categories of AI-generation features.
3. For each category, provide a brief explanation describing how the feature appears in the text.

Output Format:
1. The thinking process is enclosed within `<thinking> </thinking>` tags.
2. The key categories of AI-generation features are enclosed within `<key_features> </key_features>` tags.
3. The brief explanation of each category is enclosed within `<explanation> </explanation>` tags.

Output Example:
`<thinking>`
Step 1: [What aspect is being examined]. [Detailed observations...]
Step 2: [What aspect is being examined]. [Detailed observations...]
...
`</thinking>`
`<key_features>`"[AI-generation features 1]", "[AI-generation features 2]", ...`</key_features>`
`<explanation>`
"[AI-generation features 1]": [How this artifact manifests in the text]
"[AI-generation features 2]": [How this artifact manifests in the text]
...
`</explanation>`

Requirements:
1. Strictly follow the specified format and do not output anything irrelevant.
2. Ensure the thinking process shows clear step-by-step analysis.
3. Avoid redundant or vague explanations. Focus on the most prominent AI-generation features.
4. Do not include any authentic or real-world features in key categories and explanations.
5. Key categories must use definitive terms indicating AI-generation artifacts, NOT neutral descriptive terms.
6. Do not use bold or markdown formatting in the answer.

Figure 16: Prompt template for CoT Generation on **fake texts**.

Prompt for CoT Quality Judgment

Modality Input:
{modality input}

Corresponding CoT Annotation:
{CoT annotation}

Evaluation Criteria:
Please evaluate the explanation based on the following criteria, scoring each from 1-5:

1. ****Format Compliance (1-5)**:** Does the explanation follow the required format: `<thinking>...</thinking><key_features>...</key_features><explanation>...</explanation>` (newlines between tags are allowed)?
- 5: Perfect format with all three tags (`<thinking>, <key_features>, <explanation>`) correctly used in order and well-structured
- 4: All three tags present with minor formatting inconsistencies
- 3: Minor format issues but readable
- 2: Multiple tags missing or incorrectly used
- 1: Completely wrong format or missing tags

2. ****Thinking Quality (1-5)**:** Is the reasoning process within `<thinking> </thinking>` tags logical, clear, and step-by-step?
- 5: Excellent logical flow with clear reasoning chain
- 4: Good reasoning with minor logical gaps
- 3: Adequate reasoning but some gaps
- 2: Weak reasoning with significant logical issues
- 1: Illogical or missing reasoning

3. ****Feature Identification (1-5)**:** Are the identified AI-generation features within `<key_features> </key_features>` tags accurate, comprehensive, and meaningful?
- 5: All major AI artifacts identified with precise categorization
- 4: Most features identified with good accuracy
- 3: Some features identified but missing important ones
- 2: Few features identified or some incorrect identifications
- 1: Features are wrong or irrelevant

4. ****Explanation Quality (1-5)**:** Does the content within `<explanation> </explanation>` tags provide specific, concrete evidence from the text?
- 5: Rich specific quotes and examples from the text
- 4: Good evidence with relevant quotes
- 3: Some evidence but could be more specific
- 2: Limited evidence with vague references
- 1: No evidence or generic statements only

5. ****Content Consistency (1-5)**:** Are the contents across `<thinking>...</thinking>`, `<key_features>...</key_features>`, and `<explanation>...</explanation>` tags logically consistent? (The thinking should identify textual cues, key_features should group these cues into categories, and explanation should describe how each category manifests in the text)
- 5: Perfect consistency - thinking observations directly support key_features categories, and explanations accurately describe each category
- 4: Good consistency with minor disconnects between sections
- 3: Moderate consistency but some categories lack support from thinking or explanations
- 2: Weak consistency - significant gaps between thinking, categories, and explanations
- 1: No consistency - sections appear unrelated or contradictory

6. ****Overall Quality (1-5)**:** Holistic assessment of the annotation quality.
- 5: Publication-ready, excellent quality
- 4: High quality with minor improvements needed
- 3: Acceptable but needs improvement
- 2: Below average, significant revision required
- 1: Poor quality, needs complete rewrite

Output Format:
Provide your evaluation in the following JSON format (no other text):

```
{{
    "format_compliance": <score 1-5>,
    "thinking_quality": <score 1-5>,
    "feature_identification": <score 1-5>,
    "explanation_quality": <score 1-5>,
    "content_consistency": <score 1-5>,
    "overall_quality": <score 1-5>
}}
```

Only output the JSON with the 6 dimension scores. Do not include any unrelated content.

Figure 17: Prompt template for CoT Quality Judgment.

Prompt for Critique Refinement (Real Image)

<image>
- Part 1: User Instructions
This is a real image captured by a camera or created by a human artist.
Task:
1. Carefully inspect the given image and think about the visual cues that indicate it is captured by a camera or created by a human artist.
2. Based on your thinking, group these visual cues into specific categories of real-image authentic features.
3. For each category, provide a brief explanation describing how the features appears in the image.

Output Format:
1. The thinking process is enclosed within <thinking> </thinking> tags.
2. The key categories of real-image authentic features are enclosed within <key_features> </key_features> tags.
3. The brief explanation of each category is enclosed within <explanation> </explanation> tags.

Output Example:
<thinking>
Step 1: [What aspect is being examined]. [Detailed observations...]
Step 2: [What aspect is being examined]. [Detailed observations...]
...
</thinking>
<key_features>"[Authentic features 1]", "[Authentic features 2]", ...</key_features>
<explanation>
"[Authentic features 1]": [How this authentic features appears in the image]
"[Authentic features 2]": [How this authentic features appears in the image]
...
</explanation>

Requirements:
1. Strictly follow the specified format and do not output anything irrelevant.
2. Ensure the thinking process shows clear step-by-step analysis.
3. Avoid redundant or vague explanations. Focus on the most prominent authentic features.
4. Avoid including any AI-generation features related content in the key categories and explanations (e.g., artifacts, imperfections).
5. Key categories must use definitive terms indicating authenticity, NOT neutral descriptive terms.
6. Do not use bold or markdown formatting in the answer.

- Part 2: Original Output to Review
{original_exp}

- Part 3: CoT Quality Evaluation Rubric
{Output of quality judgment}

- Part 4: Your Review Task
Review the Original Output based on the User Instructions and check:
1. Check if the original output strictly follows the Output Format specified in the User Instructions.
2. Check if the think process (enclosed within the <thinking></thinking> tags) is coherent, logically rigorous and comprehensive.
3. Check that the output accurately identifies the categories (enclosed within <key_features> </key_features> tag) of authentic features visible in the image and whether any important authentic features were omitted.
4. Check if the explanation (enclosed within <explanation> </explanation> tag) is accurate.

- Part 5: Output
1. If there are errors, omissions, or inaccuracies: Provide the corrected version following the **User Instructions**.
2. If the original is accurate and complete, Output "None".

Figure 18: Prompt template for Critique Refinement on **real images**.

Prompt for Critique Refinement (Fake Image)

<image>
- Part 1: User Instructions
This is a fake image generated by AI.
Task:
1. Carefully inspect the given image and think about the visual cues that indicate it is AI-generated.
2. Based on your thinking, group these visual cues into specific categories of AI-generation features.
3. For each category, provide a brief explanation describing how the feature appears in the image.

Output Format:
1. The thinking process is enclosed within <thinking> </thinking> tags.
2. The key categories of AI-generation features are enclosed within <key_features> </key_features> tags.
3. The brief explanation of each category is enclosed within <explanation> </explanation> tags.

Output Example:
<thinking>
Step 1: [What aspect is being examined]. [Detailed observations...]
Step 2: [What aspect is being examined]. [Detailed observations...]
...
</thinking>
<key_features>"[AI-generation features 1]", "[AI-generation features 2]", ...</key_features>
<explanation>
"[AI-generation features 1]": [How this artifact manifests in the image]
"[AI-generation features 2]": [How this artifact manifests in the image]
...
</explanation>

Requirements:
1. Strictly follow the specified format and do not output anything irrelevant.
2. Ensure the thinking process shows clear step-by-step analysis.
3. Avoid redundant or vague explanations. Focus on the most prominent AI-generation features.
4. Do not include any authentic or real-world features in key categories and explanations.
5. Key categories must use definitive terms indicating AI-generation artifacts, NOT neutral descriptive terms.
6. Do not use bold or markdown formatting in the answer.

- Part 2: Original Output to Review
{original_exp}

- Part 3: CoT Quality Evaluation Rubric
{Output of quality judgment}

- Part 4: Your Review Task
Review the Original Output based on the User Instructions and check:
1. Check if the original output strictly follows the Output Format specified in the User Instructions.
2. Check if the think process (enclosed within the <thinking></thinking> tags) is coherent, logically rigorous and comprehensive.
3. Check that the output accurately identifies the categories (enclosed within <key_features> </key_features> tag) of AI-generation features visible in the image and whether any important AI-generation features were omitted.
4. Check if the explanation (enclosed within <explanation> </explanation> tag) is accurate.

- Part 5: Output
1. If there are errors, omissions, or inaccuracies: Provide the corrected version following the **User Instructions**.
2. If the original is accurate and complete, Output "None".

Figure 19: Prompt template for Critique Refinement on **fake images**.

Prompt for Critique Refinement (Real Video)

<video>
- Part 1: User Instructions
This is a real video captured by physical camera equipment.
Task:
1. Carefully inspect the given video and think about the visual cues that indicate it is captured by physical camera equipment.
2. Based on your thinking, group these visual cues into specific categories of real-video authentic features.
3. For each category, provide a brief explanation describing how the features appears in the video.

Output Format:
1. The thinking process is enclosed within <thinking> </thinking> tags.
2. The key categories of real-video authentic features are enclosed within <key_features> </key_features> tags.
3. The brief explanation of each category is enclosed within <explanation> </explanation> tags.

Output Example:
<thinking>
Step 1: [What aspect is being examined]. [Detailed observations...]
Step 2: [What aspect is being examined]. [Detailed observations...]
...
</thinking>
<key_features>"[Authentic features 1]", "[Authentic features 2]", ...</key_features>
<explanation>
"[Authentic features 1]": [How this authentic features appears in the video]
"[Authentic features 2]": [How this authentic features appears in the video]
...
</explanation>

Requirements:
1. Strictly follow the specified format and do not output anything irrelevant.
2. Ensure the thinking process shows clear step-by-step analysis.
3. Avoid redundant or vague explanations. Focus on the most prominent authentic features.
4. Avoid including any AI-generation features related content in the key categories and explanations (e.g., artifacts, imperfections).
5. Key categories must use definitive terms indicating authenticity, NOT neutral descriptive terms.
6. Do not use bold or markdown formatting in the answer.

- Part 2: Original Output to Review
{original_exp}

- Part 3: CoT Quality Evaluation Rubric
{Output of quality judgment}

- Part 4: Your Review Task
Review the Original Output based on the User Instructions and check:
1. Check if the original output strictly follows the Output Format specified in the User Instructions.
2. Check if the think process (enclosed within the <thinking></thinking> tags) is coherent, logically rigorous and comprehensive.
3. Check that the output accurately identifies the categories (enclosed within <key_features> </key_features> tag) of authentic features visible in the video and whether any important authentic features were omitted.
4. Check if the explanation (enclosed within <explanation> </explanation> tag) is accurate.

- Part 5: Output
1. If there are errors, omissions, or inaccuracies: Provide the corrected version following the **User Instructions**.
2. If the original is accurate and complete, Output "None".

Figure 20: Prompt template for Critique Refinement on **real videos**.

Prompt for Critique Refinement (Fake Video)

<video>
- Part 1: User Instructions
This is a fake video generated by AI.
Task:
1. Carefully inspect the given video and think about the visual cues that indicate it is AI-generated.
2. Based on your thinking, group these visual cues into specific categories of AI-generation features.
3. For each category, provide a brief explanation describing how the feature appears in the video.

Output Format:
1. The thinking process is enclosed within `<thinking> </thinking>` tags.
2. The key categories of AI-generation features are enclosed within `<key_features> </key_features>` tags.
3. The brief explanation of each category is enclosed within `<explanation> </explanation>` tags.

Output Example:
`<thinking>`
Step 1: [What aspect is being examined]. [Detailed observations...]
Step 2: [What aspect is being examined]. [Detailed observations...]
...
`</thinking>`
`<key_features>`"[AI-generation features 1]", "[AI-generation features 2]", ...`</key_features>`
`<explanation>`
"[AI-generation features 1]": [How this artifact manifests in the video]
"[AI-generation features 2]": [How this artifact manifests in the video]
...
`</explanation>`

Requirements:
1. Strictly follow the specified format and do not output anything irrelevant.
2. Ensure the thinking process shows clear step-by-step analysis.
3. Avoid redundant or vague explanations. Focus on the most prominent AI-generation features.
4. Do not include any authentic or real-world features in key categories and explanations.
5. Key categories must use definitive terms indicating AI-generation artifacts, NOT neutral descriptive terms.
6. Do not use bold or markdown formatting in the answer.

- Part 2: Original Output to Review
{original_exp}

- Part 3: CoT Quality Evaluation Rubric
{Output of quality judgment}

- Part 4: Your Review Task
Review the Original Output based on the User Instructions and check:
1. Check if the original output strictly follows the Output Format specified in the User Instructions.
2. Check if the think process (enclosed within the `<thinking></thinking>` tags) is coherent, logically rigorous and comprehensive.
3. Check that the output accurately identifies the categories (enclosed within `<key_features> </key_features>` tag) of AI-generation features visible in the video and whether any important AI-generation features were omitted.
4. Check if the explanation (enclosed within `<explanation> </explanation>` tag) is accurate.

- Part 5: Output
1. If there are errors, omissions, or inaccuracies: Provide the corrected version following the **User Instructions**.
2. If the original is accurate and complete, Output "None".

Figure 21: Prompt template for Critique Refinement on **fake videos**.

Prompt for Critique Refinement (Real Text)

- Part 1: User Instructions
This is a real text written by human: {real_text}
Task:
1. Carefully inspect the given text and think about the cues that indicate it is written by a human.
2. Based on your thinking, group these cues into specific categories of authentic features.
3. For each category, provide a brief explanation describing how the features appears in the text.

Output Format:
1. The thinking process is enclosed within <thinking> </thinking> tags.
2. The key categories of authentic features are enclosed within <key_features> </key_features> tags.
3. The brief explanation of each category is enclosed within <explanation> </explanation> tags.

Output Example:
<thinking>
Step 1: [What aspect is being examined]. [Detailed observations...]
Step 2: [What aspect is being examined]. [Detailed observations...]
...
</thinking>
<key_features>"[Authentic features 1]", "[Authentic features 2]", ...</key_features>
<explanation>
"[Authentic features 1]": [How this authentic features appears in the text]
"[Authentic features 2]": [How this authentic features appears in the text]
...
</explanation>

Requirements:
1. Strictly follow the specified format and do not output anything irrelevant.
2. Ensure the thinking process shows clear step-by-step analysis.
3. Avoid redundant or vague explanations. Focus on the most prominent authentic features.
4. Avoid including any AI-generation features related content in the key categories and explanations.
5. Key categories must use definitive terms indicating authenticity, NOT neutral descriptive terms.
6. Do not use bold or markdown formatting in the answer.

- Part 2: Original Output to Review
{original_exp}

- Part 3: CoT Quality Evaluation Rubric
{Output of quality judgment}

- Part 4: Your Review Task
Review the Original Output based on the User Instructions and check:
1. Check if the original output strictly follows the Output Format specified in the User Instructions.
2. Check if the think process (enclosed within the <thinking></thinking> tags) is coherent, logically rigorous and comprehensive.
3. Check that the output accurately identifies the categories (enclosed within <key_features> </key_features> tag) of authentic features visible in the text and whether any important authentic features were omitted.
4. Check if the explanation (enclosed within <explanation> </explanation> tag) is accurate.

- Part 5: Output
1. If there are errors, omissions, or inaccuracies: Provide the corrected version following the **User Instructions**.
2. If the original is accurate and complete, Output "None".

Figure 22: Prompt template for Critique Refinement on **real texts**.

Prompt for Critique Refinement (Fake Text)

- Part 1: User Instructions
This is a fake text generated by AI: {fake_text}
Task:
1. Carefully analyze the given text and think about the textual cues that indicate it is AI-generated.
2. Based on your thinking, group these textual cues into specific categories of AI-generation features.
3. For each category, provide a brief explanation describing how the feature appears in the text.

Output Format:
1. The thinking process is enclosed within <thinking> </thinking> tags.
2. The key categories of AI-generation features are enclosed within <key_features> </key_features> tags.
3. The brief explanation of each category is enclosed within <explanation> </explanation> tags.

Output Example:
<thinking>
Step 1: [What aspect is being examined]. [Detailed observations...]
Step 2: [What aspect is being examined]. [Detailed observations...]
...
</thinking>
<key_features>"[AI-generation features 1]", "[AI-generation features 2]", ...</key_features>
<explanation>
"[AI-generation features 1]": [How this artifact manifests in the text]
"[AI-generation features 2]": [How this artifact manifests in the text]
...
</explanation>

Requirements:
1. Strictly follow the specified format and do not output anything irrelevant.
2. Ensure the thinking process shows clear step-by-step analysis.
3. Avoid redundant or vague explanations. Focus on the most prominent AI-generation features.
4. Do not include any authentic or real-world features in key categories and explanations.
5. Key categories must use definitive terms indicating AI-generation features, NOT neutral descriptive terms.
6. Do not use bold or markdown formatting in the answer.

- Part 2: Original Output to Review
{original_exp}

- Part 3: CoT Quality Evaluation Rubric
{Output of quality judgment}

- Part 4: Your Review Task
Review the Original Output based on the User Instructions and check:
1. Check if the original output strictly follows the Output Format specified in the User Instructions.
2. Check if the think process (enclosed within the <thinking></thinking> tags) is coherent, logically rigorous and comprehensive.
3. Check that the output accurately identifies the categories (enclosed within <key_features> </key_features> tag) of AI-generation features visible in the text and whether any important AI-generation features were omitted.
4. Check if the explanation (enclosed within <explanation> </explanation> tag) is accurate.

- Part 5: Output
1. If there are errors, omissions, or inaccuracies: Provide the corrected version following the **User Instructions**.
2. If the original is accurate and complete, Output "None".

Figure 23: Prompt template for Critique Refinement on **fake texts**.