

VDC-Agent: When Video Detailed Captioners Evolve Themselves via Agentic Self-Reflection

Qiang Wang¹, Xinyuan Gao², Yuhang He^{1*}, Jizhou Han¹, Jiangyang Li¹,
Songlin Dong^{3,4*}, Zhiheng Ma^{3,4,5*}, and Yihong Gong^{1,3*}

¹ Xi'an Jiaotong University

² Kuaishou Technology

³ Shenzhen University of Advanced Technology

⁴ Guangdong Provincial Key Laboratory of Computility Microelectronics

⁵ Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
{qwang,gxy010317}@stu.xjtu.edu.cn,ygong@mail.xjtu.edu.cn

Abstract. Existing Video Detailed Captioning (VDC) methods predominantly rely on costly human annotations or distillation from powerful proprietary models, creating a dependency on external supervision. In this paper, we propose **VDC-Agent**, an autonomous self-evolving framework that empowers a single Multimodal Large Language Model (MLLM) to generate and refine high-quality captions through principle-guided self-reflection. To overcome the inference latency inherent in iterative refinement, we further propose to *internalize* this reflective capability into the model. Specifically, we construct VDC-Agent-19K, a preference dataset derived from the agent’s self-scored trajectories, and introduce a Curriculum Direct Preference Optimization (DPO) strategy. This strategy leverages the quality gap between generated candidates to progressively align the model from easy to hard samples. Extensive experiments demonstrate that VDC-Agent achieves state-of-the-art performance on VDC and DREAM-1K benchmarks, generating captions with superior detail and faithfulness. Crucially, our internalization strategy retains the inference efficiency of the base model while significantly enhancing its generalization capabilities, as validated by both quantitative metrics and human evaluation.

Keywords: Video Captioning · Agent · Self-Reflection · Multimodal Large Language Models

Project Page: <https://vdcagent.github.io>

1 Introduction

With the explosive growth of online videos, understanding complex visual and temporal information has become a core capability for modern multimodal AI systems. Within this landscape, **Video Detailed Captioning (VDC)** [6, 40,

* Corresponding authors.

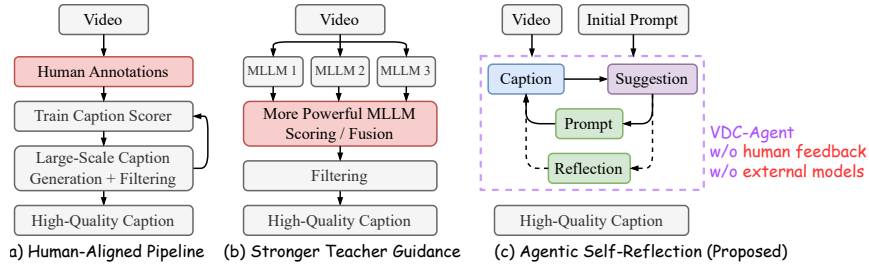


Fig. 1: Comparison of video captioning paradigms. (a) Human-aligned pipelines rely on manual annotations to train caption scorers. (b) Multi-MLLM-based pipelines depend on multiple or stronger MLLMs for scoring or fusion. (c) Our proposed **VDC-Agent** achieves self-improvement through agentic self-reflection, requiring neither human annotations nor larger models.

61, 78] serves as a fundamental task that aims to generate fine-grained and comprehensive descriptions capturing objects, actions, interactions, and scene transitions in videos. By enabling a more precise and detailed understanding of video content, VDC facilitates progress in a wide range of applications such as visual question answering [35, 86], open-vocabulary semantic segmentation [74–76], object detection [57, 73], text-to-video generation [29, 58], video retrieval [19], and temporal localization [45, 81].

Recent progress in VDC largely stems from fine-tuning Multimodal Large Language Models (MLLMs) on curated video-caption datasets [48, 53, 59]. While effective, these methods typically depend on external supervision, as shown in Fig. 1 (a)(b). Approaches like ShareGPT4Video [7], AVC-DPO [59], and VideoCap-R1 [48] rely on distilling capabilities from powerful proprietary or open-source models (*e.g.*, GPT-4V [1], Qwen-72B [3]), whereas Cockatiel [53] relies on extensive human annotations to align with preferences, and OwlCap [85] leverages more powerful models to fuse captions from different MLLMs.

However, existing VDC paradigms are trapped in an unsustainable resource dilemma: they either suffer from prohibitive human annotation costs or rely heavily on the massive computation of larger teacher models. To break this deadlock, we ask: *Can we enable a model to act as its own “critic” (akin to human introspection) to autonomously evaluate and correct its outputs without external intervention?* Achieving this would eliminate the dependency on expensive external MLLMs and extensive manual annotations. To this end, we propose transforming the captioner into an **autonomous agent**. By endowing it with intrinsic reflective capabilities, the agent establishes a closed loop to independently generate, evaluate, and refine its captions through iterative self-reflection.

To this end, we propose **VDC-Agent**, a self-evolving video captioning framework that enables MLLMs to improve themselves through iterative self-reflection without requiring stronger external supervision. As depicted in Fig. 1 (c), VDC-Agent forms a closed-loop system that continuously refines its captioning ability by alternating between caption generation, evaluation, and prompt refinement.

Given a collection of unlabeled videos, the model first generates captions using an initial prompt. It then conducts self-assessment based on a set of principles describing what constitutes a good caption (such as coverage of objects, actions, and temporal dynamics), assigns a quality score to the caption, and produces textual suggestions for improvement. These suggestions guide an internal *prompt refiner*, which updates the prompt in the next iteration. If the newly generated caption is even worse than the previous one, the model triggers a *self-reflection* mechanism that revisits the chain of thought used in the last prompt refinement, diagnosing why the previous update failed and avoiding the same mistake in subsequent steps. Through repeated cycles of caption–evaluation–refinement, our VDC-Agent can generate higher-quality video descriptions.

However, the iterative nature of VDC-Agent inevitably increases inference latency. To reconcile high performance with efficiency, we propose to *internalize* the self-reflection capability into the MLLM, enabling it to achieve the effect of multiple rounds of reflection within a single inference step. Specifically, we apply VDC-Agent to a high-resolution video corpus (Cockatiel-4K [53]) to generate scored caption trajectories. By selecting the best and worst candidates, we construct a preference dataset, **VDC-Agent-19K**, comprising 18,886 pairs with quantified quality gaps. To fully exploit these pairs, we introduce a *Curriculum Direct Preference Optimization (DPO)* strategy that progressively learns from easy (large score gap) to hard (small score gap) samples. We fine-tune Qwen2.5-VL-7B-Instruct with this strategy to obtain VDC-Agent-7B.

Extensive experiments validate the effectiveness of our framework from multiple perspectives. On the VDC benchmark, our model achieves state-of-the-art results (49.08% accuracy, 2.50 score), demonstrating its capacity to generate richer and more fine-grained details. On the DREAM-1K benchmark, the significant improvements in Precision and Recall (37.4% F1) suggest that our method produces *faithful descriptions rather than merely longer text*. Furthermore, human evaluation verifies that the positive-negative pairs in VDC-Agent-19K are *highly consistent with human preferences*. Finally, ablation studies demonstrate that while the self-reflection mechanism is effective, internalizing it yields the best of both worlds: it *further improves accuracy and generalization* compared to test-time iteration while drastically *reducing inference time* to the level of the base model.

Our contributions can be summarized as follows:

- We propose VDC-Agent, an autonomous framework that enables a single MLLM to generate and refine detailed video captions via self-reflection, eliminating the need for human annotation or external teacher models.
- We construct a preference dataset VDC-Agent-19K and introduce curriculum DPO that exploits the score gap between preferred and dispreferred captions as a difficulty signal to enable easy-to-hard preference alignment.
- We achieve state-of-the-art performance on two VDC benchmarks. Human evaluation and ablation studies validate the effectiveness and efficiency of our self-reflection and internalization strategies.

2 Related Work

2.1 Video Detailed Captioning

Video detailed captioning (VDC) [6, 8, 33, 40, 43, 52, 68, 69] is a fundamental task in video understanding that aims to generate precise and comprehensive descriptions of video content. Early non-LLM approaches [22, 34] typically produced short, fragmentary captions with limited temporal grounding, which made them hard to use in downstream reasoning or retrieval systems. With the rapid progress of Multimodal Large Language Models (MLLMs) [3, 9, 13, 15], recent works have substantially improved the fluency and coherence of video captions. However, generic MLLMs are not explicitly optimized for fine-grained and temporally grounded descriptions, and they often miss subtle events, camera operations, or background cues that are critical for VDC.

Recent methods construct high-quality video-caption pairs and then fine-tune via LoRA [84] or Adapter [39, 62] on these curated datasets. Cockatiel [53] and Vripter [70] build dense, human-annotated resources or human-preferred scorers to obtain detailed captions, while ShareGPT4Video [7], Shot2Story [28], and LLaVA-Video [82] leverage GPT-4V or GPT-4o to synthesize large-scale annotations that are further refined by filtering or human checks. More recently, AVC-DPO [59] and VideoCap-R1 [48] employ powerful open-source models (*e.g.*, Qwen-72B) as teachers or reward providers to train smaller captioners with preference optimization or reinforcement learning. Despite their success, these approaches still hinge on stronger MLLMs or extensive human supervision for data construction, which leads to considerable annotation and computation costs and limits scalability and reproducibility in practice.

2.2 Multimodal Large Language Model Agent

Recent research has shifted from viewing LLMs as passive predictors to agents that interleave reasoning with actions [4, 10, 26, 27, 32, 47, 55, 66, 67, 72, 77, 79]. These frameworks enable robust decision-making through iterative “think-then-act” cycles. In the multimodal domain, agents extend this capability by integrating vision experts [71], navigating GUIs [30, 38, 80], or querying long videos [17, 65]. Distinct from these approaches that interact with external environments or tools, our VDC-Agent internalizes the action space. It operates directly on its own prompts and captions, establishing a closed loop of generation, principle-guided evaluation, and self-reflection to autonomously enhance video detailed captioning without relying on external teacher models.

3 Method

To obtain a stronger video detailed captioner, we first construct high-quality / low-quality caption pairs with an agentic data generator, **VDC-Agent**, and then fine-tune a multimodal large language model (MLLM) using direct preference optimization (DPO). Sec. 3.1 details how VDC-Agent iteratively improves caption quality. Sec. 3.2 shows how the resulting sequence of captions

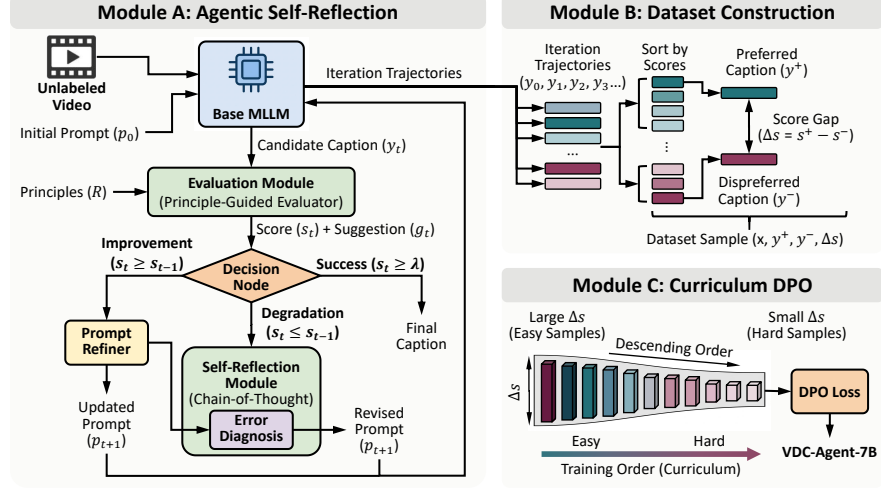


Fig. 2: Overview of the VDC-Agent Framework. (A) Agentic Self-Reflection: The agent iteratively refines captions through a closed-loop system of generation, evaluation, and self-reflection, dynamically handling both improvements and degradations. **(B) Dataset Construction:** We extract preference pairs (y^+, y^-) from the iteration trajectories based on quality scores to form the dataset. **(C) Curriculum DPO:** The model is fine-tuned by organizing these pairs from easy (large score gap Δs) to hard (small Δs), effectively internalizing the reflective capability.

is transformed into a training dataset. Finally, Sec. 3.3 explains how the positive-negative pairs are leveraged to train our model via curriculum DPO.

3.1 VDC-Agent

Fig. 2 illustrates the overall framework of our VDC-Agent. Given a collection of videos, VDC-Agent automatically generates multiple captions for each video, scores their quality, and finally produces preference pairs used for DPO training. The key idea is to let an MLLM iteratively refine the task prompts via self-reflection, so that later iterations yield more detailed captions.

We set a maximum number of iterations T and use $t \in \{0, 1, \dots, T\}$ to index the interaction steps. At step t , the prompt, caption, score, suggestion are denoted by p_t , y_t , s_t , and g_t , respectively. As preparation, the user only needs to specify an initial prompt p_0 (e.g., “Please provide a detailed description of this video.”) and a set of textual principles R that describe what a good caption should look like (e.g., covering camera motion, background, main objects, etc.). The details of principles will be provided in the Appendix. After that, VDC-Agent can run fully automatically over a video collection.

Caption generation. For a video x and a given prompt p_t , a base MLLM produces a candidate caption:

$$y_t = f(x; \theta, p_t), \quad 0 \leq t \leq T, \quad (1)$$

where $f(\cdot; \cdot)$ denotes the base MLLM and Θ is its parameter set. When $t = 0$, the caption y_0 is generated using the initial prompt p_0 .

Scoring and suggestion generation. The caption y_t is then fed into the MLLM (which shares the same backbone but is used with different instructions) together with the video x and the principle R . This *principle-guided* MLLM outputs both a scalar quality score and natural-language suggestions for improving the prompt:

$$(s_t, g_t) = f(x, y_t; \Theta, R), \quad 0 \leq t \leq T, \quad (2)$$

where s_t is the quality score from 0 to 100, and g_t is a textual suggestion describing how the next prompt should be revised (*e.g.*, asking for more spatial details or emphasizing the main object).

Iterative prompt refinement with reflection. VDC-Agent decides whether and how to update the prompt p_t based on the score s_t . Given a score threshold λ , if the caption already satisfies the requirement, *i.e.*, $s_t \geq \lambda$, the agent stops and outputs the current caption. Otherwise, it refines the prompt and continues.

We distinguish two cases for unsuccessful captions. If the score increases or remains unchanged compared with the previous step, the agent simply performs *prompt refinement*, using the suggestion g_t as guidance. If the score drops below the previous score, this indicates that the last refinement was harmful. In this case VDC-Agent triggers a *self-reflection* stage: it asks the MLLM to reason about why the previous refinement failed, using the current prompt and caption, with the previous round’s prompt–caption pair and the chain-of-thought (CoT) produced by the previous prompt refiner, and then proposes a more reliable prompt update. Formally, the prompt update rule can be written as:

$$p_{t+1} = \begin{cases} p_t, & s_t \geq \lambda, \\ f(x, y_t, s_t; \Theta, p_{\text{refine}}), & s_t \in [s_{t-1}, \lambda), \\ f(x, y_t, s_t, p_t; \Theta, p_{\text{reflect}}), & s_t < s_{t-1}, \end{cases} \quad (3)$$

where p_{refine} is the system-level instruction used by the prompt refiner and p_{reflect} is the instruction for the self-reflection module. The system prompts of the (self-reflection) prompt refiner will be provided in the Appendix. The self-reflection trigger rate and examples for generation will be provided in the Appendix. For clarity, we omit the explicit CoT variables in Eq. (3); in practice, the CoT generated in the previous step is provided as additional context so that the agent can diagnose errors and avoid repeating them. The detailed pseudo-code and algorithm are presented in the Appendix. During the above process, each iteration produces a pair (y_t, s_t) . After the loop terminates due to either reaching the threshold or hitting the maximum iteration T , we keep all caption-score pairs generated for this video and pass them to the data construction stage.

3.2 Dataset Construction by VDC-Agent

High-quality video data is crucial for training reliable detailed captioners. We adopt the Cockatiel-4K [53] corpus, which contains 4,008 high-resolution videos

sampled from OpenVid-1M. To expose the model to complementary aspects of video understanding, we follow the taxonomy in VDC [6] and generate captions along five task dimensions: camera (shot type and camera motion), short (a concise summary), background (scene layout and context), main object (key objects and their attributes), and detailed (fine-grained temporal and spatial events). For each video x and each task dimension d , VDC-Agent executes the iterative procedure of Sec. 3.1 and produces a trajectory of caption-score pairs:

$$\mathcal{P}(x, d) = \{(y_t, s_t)\}_{t=0}^{T_v(x, d)}, \quad 1 \leq T_v(x, d) \leq T, \quad (4)$$

where y_t is the caption at iteration t , s_t is its quality score, and $T_v(x, d)$ is the number of iterations before termination by threshold or by the cap T . Across 4,008 videos and 5 dimensions, we obtain a total of 20,040 raw caption sets.

Rule-based filtering. If a video already attains a score no smaller than the threshold λ in the first iteration (*i.e.*, $|\mathcal{P}(x, d)| = 1$), the existing MLLM provides a sufficiently good caption without any refinement. Such cases offer little learning signal for preference modeling and are therefore removed. This filter discards 1,078 (x, d) pairs. In addition, we remove 76 pairs due to JSON formatting / parsing errors during automatic generation. All filtering is *fully automatic* and rule-based, *requiring no human annotation*. After filtering, we retain 18,886 caption sets for the next stage. The details of the dataset and the filtered data will be provided in the Appendix.

Constructing positive-negative pairs. For every remaining set $\mathcal{P}(x, d)$, we sort captions by their scores and select the highest-scoring caption as the *best* caption y^+ with score s^+ , and the lowest-scoring caption as the *worst* caption y^- with score s^- . To quantify the preference strength (and later drive the curriculum), we compute the score gap $\Delta s = s^+ - s^-$. Each video and task dimension then contributes one training tuple $(x, y^+, y^-, \Delta s)$, forming the dataset used by the training in Sec. 3.3.

3.3 VDC Learning with Curriculum DPO

We fine-tune the base MLLM with DPO, which updates a policy π_θ from paired preferences without training a reward model or running on-policy RL.

Vanilla DPO. Given a video x , a preferred caption y^+ and a dispreferred caption y^- , DPO [54] maximizes the probability that π_θ prefers y^+ over y^- while keeping it close to a reference policy π_{ref} :

$$\mathcal{L}_{\text{DPO}}(\theta; x, y^+, y^-) = -\log \sigma \left(\beta \left[\log \frac{\pi_\theta(y^+ | x)}{\pi_\theta(y^- | x)} - \log \frac{\pi_{\text{ref}}(y^+ | x)}{\pi_{\text{ref}}(y^- | x)} \right] \right), \quad (5)$$

where $\beta > 0$ controls the implicit KL strength.

Why curriculum? Vanilla DPO treats every pair equally, which is suboptimal for our data. In our setting, pair difficulty varies: some preferred-dispreferred pairs differ markedly (large semantic gap), which are ideal for early-stage coarse adaptation [5, 24, 63]; others are much closer (small gap), which are valuable for

late-stage fine-grained alignment [12, 25, 37]. Fortunately, the dataset produced by VDC-Agent naturally provides a difficulty signal, *i.e.*, the score gap Δs , quantifying how much better the preferred caption is than the dispreferred one. **Curriculum DPO.** To let the model learn *from easy to hard* in an optimization-friendly manner, we adopt a simple yet effective curriculum DPO [5, 51]. We exploit the score gap Δs as a built-in difficulty signal and only modify the sampling order. Specifically, we sort all tuples $\mathcal{D} = \{(x_i, y_i^+, y_i^-, \Delta s_i)\}_{i=1}^N$ by Δs_i in descending order ($\Delta s_1 \geq \dots \geq \Delta s_N$), and feed mini-batches sequentially along this list (large $\Delta s \rightarrow$ small Δs). The optimization objective is

$$\min_{\theta} \sum_{i=1}^N \mathcal{L}_{\text{DPO}}(\theta; x_i, y_i^+, y_i^-), \text{ with order } i = 1 \rightarrow N. \quad (6)$$

Intuitively, large-gap pairs yield strong, low-variance gradients that quickly steer the policy toward the correct global behavior; small-gap pairs then refine subtle distinctions without delicate weight tuning. Moreover, a cosine-decay learning-rate schedule naturally complements this curriculum: early confident pairs are learned with larger step sizes, while later ambiguous pairs are absorbed with smaller steps. Empirically, we find that this simple curriculum accelerates convergence and yields better final alignment for video detailed captioning.

4 Experiments

4.1 Experimental Setup

Training data. We construct our training dataset based on Cockatiel-4K [53] due to its high visual quality, which contains 4,008 high-resolution videos sampled from OpenVid-1M [49]. We only use the videos from Cockatiel-4K without captions. Importantly, there is no overlap between the training videos and those used in the VDC evaluation benchmarks. Following our VDC-Agent pipeline, we generate raw caption trajectories on five task dimensions (camera, short, background, main object, detailed), producing 20,040 raw sets in total. Applying the rule-based filtering in Sec. 3.2 to remove trivial one-step cases and malformed JSON outputs, we obtain 18,886 preference pairs for training, denoted as VDC-Agent-19K.

Implementation details. We use Qwen2.5-VL-7B-Instruct [3] as our base MLLM. We fine-tune with curriculum DPO using VDC-Agent-19K. For efficiency, only the LLM backbone is adapted with LoRA [20, 31, 64] while all other parameters are frozen. We set the LoRA rank and alpha to 16 and 32, respectively, with dropout 0.1. Training runs for 3 epochs with a cosine-decay learning-rate schedule (initial LR 5×10^{-5} , warmup 10% of total steps). We train on $4 \times$ NVIDIA A800 GPUs with a global batch size of 16. The maximum number of iterations T is set to 4. λ is set to 90. Following AVC-DPO [59], we set the maximum number of sampled frames to 32 and tokens per frame to 128. For decoding, we use a beam size of 1, a temperature of 0.1, and a maximum caption length of 1024.

Table 1: VDCscore comparison on the VDC benchmark. “Human” indicates whether human annotations are used. “External” indicates whether larger external teacher models are involved. We report Accuracy/Score for five dimensions (higher is better) and the average across dimensions. Qwen2.5-VL-7B-Instruct is the baseline model of our VDC-Agent-7B. **Bold** denotes the best results, and underline denotes the second-best results.

Model	H E	Camera Acc./Score	Short Acc./Score	Background Acc./Score	Main Object Acc./Score	Detailed Acc./Score	Average Acc./Score
General MLLMs							
Llama 3.1-8B [16]	- -	17.83/1.00	17.90/1.02	19.52/1.00	19.57/1.10	20.10/1.22	18.98/1.07
Gemini 1.5 Pro [60]	- -	38.68/2.05	35.71/1.85	43.84/2.23	47.32/2.41	43.11/2.22	41.73/2.15
GPT-5.4 [50]	- -	46.35/2.38	35.56/1.82	47.04/2.40	44.65/2.28	45.09/2.30	43.74/2.24
Gemini 3.1 Pro [21]	- -	49.25/2.51	37.63/1.91	48.10/2.45	48.71/2.49	47.31/2.40	46.20/2.35
LLaMA-VID-7B [42]	- -	39.47/2.10	29.92/1.56	28.01/1.45	31.24/1.59	25.67/1.38	30.86/1.62
Video-ChatGPT-7B [46]	- -	37.46/2.00	29.36/1.56	33.68/1.70	30.47/1.60	24.61/1.26	31.12/1.62
Video-LLaVA-7B [44]	- -	37.48/1.97	30.67/1.63	32.50/1.70	36.01/1.85	27.36/1.43	32.80/1.72
VideoChat-Flash-7B [41]	- -	43.70/2.30	33.70/1.70	45.10/2.30	47.60/2.40	44.50/2.30	42.92/2.20
Video-R1-7B [18]	- -	42.70/2.20	44.50/2.30	40.60/2.10	45.90/2.30	45.60/2.40	43.86/2.26
Qwen3-VL-8B [2]	- -	48.44/2.45	<u>36.58/1.86</u>	47.69/2.42	46.94/2.41	46.75/2.39	45.28/2.31
Video Caption MLLMs							
ShareGPT4Video-8B [7]	✗ ✓	33.28/1.76	39.08/1.94	35.77/1.81	37.12/1.89	35.62/1.84	36.17/1.85
Vriptor [70]	✓ ✗	37.64/1.96	38.35/2.00	37.11/1.94	37.02/1.93	38.49/2.00	37.72/1.97
AuroraCap-7B [6]	✗ ✗	43.50/2.27	32.07/1.68	35.92/1.84	39.02/1.97	41.30/2.15	38.36/1.98
Cockatiel-8B [53]	✓ ✗	42.25/2.19	44.01/2.27	43.89/2.26	43.85/2.26	44.00/2.27	43.60/2.25
VideoCap-R1-7B [48]	✗ ✓	41.70/2.30	35.20/1.90	47.20/2.50	47.00/2.50	43.80/2.40	42.98/2.32
SynPO [11]	✗ ✗	-/1.78	-/1.94	-/1.91	-/1.87	-/2.04	-/1.91
AVC-DPO-7B [59]	✗ ✓	50.40/2.66	39.00/2.03	49.90/2.57	50.50/2.58	48.90/2.54	47.70/2.47
OwlCap-7B [85]	✗ ✓	46.90/2.50	47.00/2.40	46.50/2.40	47.10/2.50	46.80/2.40	46.90/2.40
LLaVA-OneVision-7B [36]	- -	37.82/2.02	32.58/1.70	37.43/1.92	38.21/1.96	41.20/2.13	37.45/1.95
w/ VDC-Agent (Ours)	✗ ✗	44.15/2.31	35.22/1.84	45.02/2.26	43.55/2.25	47.96/2.37	43.18/2.21
Improvement	✗ ✗	+6.33/+0.29	+2.64/+0.14	+7.59/+0.34	+5.34/+0.29	+6.76/+0.24	+5.73/+0.26
InternVL2.5-8B [9]	- -	40.22/2.16	35.59/1.86	38.93/2.03	45.70/2.36	36.75/2.01	39.44/2.08
w/ VDC-Agent (Ours)	✗ ✗	48.08/2.53	38.17/1.96	46.59/2.32	49.13/2.48	42.43/2.23	44.88/2.30
Improvement	✗ ✗	+7.86/+0.37	+2.58/+0.10	+7.66/+0.29	+3.43/+0.12	+5.68/+0.22	+5.44/+0.22
Qwen2.5-VL-7B [3]	- -	42.61/2.18	37.76/1.91	44.10/2.22	49.00/2.47	46.25/2.35	43.95/2.23
w/ VDC-Agent (Ours)	✗ ✗	50.52/2.67	39.49/1.99	51.93/2.62	53.23/2.65	50.21/2.55	49.08/2.50
Improvement	✗ ✗	+7.91/+0.49	+1.73/+0.08	+7.83/+0.40	+4.23/+0.18	+3.96/+0.20	+5.13/+0.27

Evaluation benchmarks. We evaluate our model on the VDC and DREAM-1K benchmarks to assess the models in video captioning tasks. VDC is a detailed video captioning benchmark containing 1,027 videos that span a wide range of categories, and we report the VDCscore following AuroraCap [6]. DREAM-1K is a collection of 1,000 annotated video clips with diverse complexities, for which we report the event-level precision, recall, and F1 score following Tarsier [61].

4.2 Main Results

Compared methods. We compare our method with two categories of models: General MLLMs, such as Llama 3.1-8B, and specialized Video Caption MLLMs with approximately 7B parameters, such as VideoCap-R1-7B, AVC-DPO-7B, and OwlCap-7B. Notably, while several top-performing baselines heavily rely on human annotations or distillation from larger teacher models (e.g., 72B parameters), our VDC-Agent operates entirely autonomously. To demonstrate the method-agnostic generalization of our framework, we apply it to three distinct base models: LLaVA-OneVision-7B, InternVL2.5-8B, and Qwen2.5-VL-7B.

Table 2: Comparison on the DREAM-1K benchmark. “Human” indicates whether human annotations are used. “External” indicates whether larger external teacher models are involved. **Bold**: the best results. Underline: the second-best results.

Model	Human	External	F1	Precision	Recall
General MLLMs					
Gemini 1.5 Pro [60]	-	-	36.2	37.6	34.8
GPT-5.4 [50]	-	-	34.9	36.2	33.7
Gemini 3.1 Pro [21]	-	-	36.4	38.1	<u>34.9</u>
LLaVA-Video-7B [82]	-	-	32.5	37.9	28.4
Video-R1-7B [18]	-	-	33.3	32.2	34.5
Qwen3-VL-8B [2]	-	-	32.5	33.6	31.5
Video Caption MLLMs					
Tarsier-7B [61]	✓	✓	34.6	<u>40.3</u>	30.2
ShareGPT4Video-8B [7]	✗	✓	20.4	27.6	16.1
Vriptor [70]	✓	✗	24.4	23.6	25.1
AuroraCap-7B [6]	✗	✗	20.8	24.4	18.1
VideoCap-R1-7B [48]	✗	✓	34.2	33.6	34.7
OwlCap-7B [85]	✗	✓	34.7	34.1	35.3
LLaVA-OneVision-7B [36]	-	-	31.7	34.3	29.4
w/ VDC-Agent (Ours)	✗	✗	<u>36.9</u> (+5.2)	39.6 (+5.3)	34.5 (+5.1)
InternVL2.5-8B [9]	-	-	27.6	34.7	22.9
w/ VDC-Agent (Ours)	✗	✗	33.9 (+6.3)	40.5 (+5.8)	29.1 (+6.2)
Qwen2.5-VL-7B [3]	-	-	30.1	30.5	29.7
w/ VDC-Agent (Ours)	✗	✗	37.4 (+7.3)	38.5 (+8.0)	36.3 (+6.6)

Evaluation on VDC benchmark. As shown in Tab. 1, VDC-Agent establishes a new state-of-the-art on the VDC benchmark. Built on Qwen2.5-VL-7B, our model achieves the highest average accuracy (49.08%) and score (2.50), outperforming strong teacher-dependent models like AVC-DPO-7B and OwlCap-7B. Crucially, our framework demonstrates robust cross-model transferability: applying VDC-Agent yields substantial and consistent improvements across all three base models (e.g., +5.73% average accuracy for LLaVA-OneVision and +5.44% for InternVL2.5). Dimension-wise, the gains are particularly pronounced in detail-oriented aspects such as camera (+7.91%), background (+7.83%), and main object (+4.23%). In contrast, the improvement on the ‘Short’ dimension is relatively modest (+1.73%), because base MLLMs already possess strong global summarization capabilities, whereas our method primarily targets the refinement of visual details. This proves that agentic self-reflection effectively enhances fine-grained spatial and temporal video understanding without requiring external supervision.

Evaluation on DREAM-1K benchmark. To verify that our performance gains reflect genuine improvements in visual grounding rather than overfitting to the VDCscore metric, we further evaluate on the DREAM-1K benchmark in Tab. 2. VDC-Agent consistently boosts performance across all evaluated backbones. Specifically, our Qwen2.5-VL-7B variant achieves the highest F1 score (37.4%) and Recall (36.3%) among all models, surpassing strong baselines like Tarsier-7B and OwlCap-7B. Furthermore, the InternVL2.5-8B variant attains the best Precision (40.5%). These consistent and significant gains (+5.2% to

Table 3: Preference evaluation on 500 randomly sampled caption pairs from VDC-Agent-19K. y^+ denotes the caption with a higher score, and y^- denotes the caption with a lower score. For GPT-5.1, we report results for both input orders to analyze position bias, along with their average.

Evaluator	Input Order	Preference Distribution (%)		
		y^+ Better	Tie	y^- Better
Human Evaluator 1	-	75.8	17.0	7.2
Human Evaluator 2	-	69.0	24.6	6.4
Human Evaluator 3	-	67.6	22.2	10.2
GPT-5.1	y^- then y^+	84.4	4.2	11.4
	y^+ then y^-	95.2	3.6	1.2
	Average	89.8	3.9	6.3

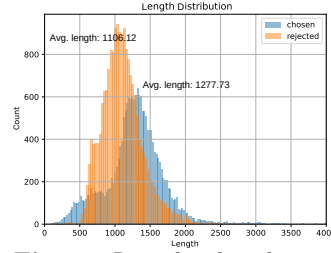


Fig. 3: Length distribution comparison between y^+ and y^- . The considerable overlap between the two distributions suggests that the preference for y^+ is not primarily driven by length.

+7.3% in F1) confirm that our self-evolving framework successfully reduces hallucinations and improves the overall faithfulness of detailed video captions.

Alignment with human judgments. To strictly validate whether the scores assigned by our method accurately reflect caption quality, we conducted a pairwise preference evaluation using 500 randomly sampled caption pairs from the VDC-Agent-19K dataset. Each pair consists of a high-scoring caption (y^+) and a low-scoring caption (y^-). We invited three independent human evaluators to perform a blind review, alongside GPT-5.1 as an AI judge.

As presented in Tab. 3, the results demonstrate a *strong consistency supporting the effectiveness of our method*. All three human evaluators consistently preferred y^+ over y^- , with preference rates ranging from 67.6% to 75.8%, significantly surpassing the cases where y^- was favored. Furthermore, GPT-5.1 exhibits a stronger preference for our high-scoring captions, with an average of 89.8%. Notably, we tested both input orders for the AI evaluator, and the consistent results confirm that the evaluation *is robust to position bias*.

In addition to preference alignment, we further investigate *potential length bias*. Fig. 3 illustrates the length distributions of the chosen captions (y^+) and the rejected captions (y^-). Although y^+ tends to be moderately longer on average, the two distributions overlap considerably. Moreover, the human evaluation in Tab. 3 consistently favors y^+ in terms of semantic richness and visual grounding, further indicating that the observed preference is largely attributable to content quality rather than length alone.

4.3 Ablation and Discussion

Why is self-reflection effective? To better understand why our agentic self-reflection design is necessary and how each component contributes, we conduct ablation experiments on the VDC benchmark. We compare four variants:

Table 4: Ablation on self-reflection and principle components. Baseline is Qwen2.5-VL-7B-Instruct, **P** denotes Principle, and **R** denotes self-Reflection. We compare the baseline, naive principle-augmented input, test-time self-reflection, and our VDC-Agent. While adding principles helps, it increases inference cost and fails to generalize to unseen videos. Self-reflection further improves caption quality but is impractical at inference time. Our model achieves the best balance of accuracy and efficiency by internalizing self-reflection through training-time agentic data generation.

Method	Camera Acc. Score		Short Acc. Score		Background Acc. Score		Main Object Acc. Score		Detailed Acc. Score		Average Acc. Score		Time
Baseline	42.61	2.18	37.76	1.91	44.10	2.22	49.00	2.47	46.25	2.35	43.95	2.23	15.5s
Baseline+P	47.76	2.43	38.81	1.96	47.82	2.41	50.07	2.53	47.87	2.42	46.47	2.35	22.3s
Baseline+P+R	45.15	2.29	39.13	1.98	50.02	2.53	52.13	2.63	49.09	2.49	47.10	2.38	164.9s
VDC-Agent	50.52	2.67	39.49	1.99	51.93	2.62	53.23	2.65	50.21	2.55	49.08	2.50	15.5s

Table 5: Ablation on curriculum strategies. We compare fine-tuning strategies on the VDC benchmark. **Random DPO** shuffles data. **Anti-Curr. DPO** sorts by ascending Δs (hard-to-easy). **Curr. DPO** sorts by descending Δs (easy-to-hard, ours).

Method	Camera Acc. Score		Short Acc. Score		Background Acc. Score		Main Object Acc. Score		Detailed Acc. Score		Average Acc. Score	
VDC-Agent (SFT)	45.43	2.36	39.30	1.99	51.35	2.56	51.60	2.58	50.01	2.54	47.54	2.41
VDC-Agent (Random DPO)	47.95	2.44	38.79	1.97	51.66	2.58	52.38	2.62	49.36	2.49	48.03	2.42
VDC-Agent (Anti-Curr. DPO)	47.31	2.42	39.20	1.98	51.69	2.58	51.19	2.56	48.86	2.47	47.65	2.40
VDC-Agent (Curr. DPO)	50.52	2.67	39.49	1.99	51.93	2.62	53.23	2.65	50.21	2.55	49.08	2.50

- **Baseline**: the original Qwen2.5-VL-7B-Instruct model without any additional mechanism;
- **Baseline + Principle**: a naive variant that directly concatenates the textual principles into the prompt without agentic iteration;
- **Baseline + Principle + Reflection**: which performs self-reflective refinement directly on the test set, but this leads to longer inference time.
- **VDC-Agent**: the proposed method.

As shown in Tab. 4, simply injecting the principle text (Baseline+P) improves over the baseline, demonstrating that explicit quality criteria provide useful prior knowledge. However, this naive strategy not only increases inference time but also fails to adapt to diverse video inputs due to its fixed principle, resulting in only a modest +2.52 average accuracy gain. The third variant (Baseline+P+R) further enhances performance, validating that iterative diagnosis and correction indeed refine caption quality. Nevertheless, it assumes test-time adaptation [14] and significantly prolongs inference time, which is impractical for deployment. In contrast, our VDC-Agent achieves comparable or better performance without repeatedly accessing the test data. This demonstrates that VDC-Agent successfully distills the benefits of reflective refinement into a single forward model, providing both robustness and efficiency at inference time.

Ablation on curriculum strategies. To validate the effectiveness of our easy-to-hard scheduling, we compare four strategies in Tab. 5. Notably, Anti-Curriculum DPO results in performance degradation compared to Random DPO,

Table 6: Ablation on the robustness to principles. Three contributors independently write principle sets (P1–P3), and P4 was derived from Shutterstock’s industrial guidelines. R denotes self-reflection (please see Tab. 4). The consistent performance across diverse principle sources demonstrates the robustness of VDC-Agent.

Contributor	Method	VDCScore		DREAM-1K		
		Acc.	Score	F1	Precision	Recall
-	Baseline	43.95	2.23	30.1	30.5	29.7
1	Baseline + P1	46.47	2.35	34.2	34.6	33.8
	Baseline + P1 + R	47.10	2.38	35.1	35.4	34.9
	VDC-Agent (with P1)	49.08	2.50	37.4	38.5	36.3
2	Baseline + P2	45.43	2.30	33.5	33.2	33.9
	Baseline + P2 + R	46.15	2.34	35.0	34.8	35.2
	VDC-Agent (with P2)	48.84	2.48	36.9	37.1	36.8
3	Baseline + P3	46.85	2.37	33.7	34.1	33.4
	Baseline + P3 + R	47.25	2.39	34.5	34.9	34.1
	VDC-Agent (with P3)	<u>49.02</u>	2.50	<u>37.1</u>	<u>38.2</u>	36.0
4 (Industry)	Baseline + P4	46.10	2.33	33.8	34.0	33.6
	Baseline + P4 + R	46.90	2.37	34.8	35.1	34.5
	VDC-Agent (with P4)	48.95	<u>2.49</u>	36.6	37.3	35.9

Table 7: Ablation study of the maximum iteration T . The generation time refers to the runtime for generating a sample on one GPU or the whole dataset on 8×A800 GPUs. We set $T = 4$ by default.

Max Iterations T	VDCScore		DREAM-1K			Dataset Construction Time
	Acc.	Score	F1	Precision	Recall	
2	46.98	2.38	34.0	34.8	33.2	41.6s / 29.0h
3	48.15	2.43	36.7	<u>38.0</u>	35.5	58.3s / 40.6h
4	49.08	2.50	37.4	38.5	36.3	70.7s / 49.2h
5	<u>49.29</u>	<u>2.52</u>	<u>37.3</u>	37.9	<u>36.8</u>	85.5s / 59.5h
6	49.55	2.53	36.8	36.5	37.0	103.8s / 72.2h

suggesting that exposing the model to the most difficult samples prematurely hinders the alignment process. In contrast, our Curriculum DPO achieves the best performance across all metrics, confirming that the proposed easy-to-hard curriculum is essential for effective optimization.

Robustness to principles. In our VDC-Agent, principles act as textual guidelines that steer automatic prompt optimization. A natural question is whether our performance depends on the exact wording or content of these principles. To verify the robustness, we conducted an ablation study to examine the *sensitivity* to different principle sets. We invited three independent contributors to write their own versions of the principles, denoted as P1, P2, and P3, without mutual discussion. Additionally, to evaluate the applicability of our method to real-world industrial standards, we curated P4 derived from the official content review guidelines of Shutterstock [56]. These principles are detailed in the Appendix. Each set covers five task dimensions (camera, short, background, main object, and detailed) but differs in linguistic style, granularity, and emphasis. For each set, we re-ran the complete pipeline described in §3. As shown

in Tab. 6, despite minor fluctuations across different sources, the final results remain highly consistent (around 49.0 accuracy and 2.5 score). This indicates that our improvement mainly stems from the agent’s iterative self-reflection and prompt refinement mechanism rather than the particular phrasing of principles, demonstrating strong robustness to principle design.

Effect of the maximum iteration T . VDC-Agent’s refinement depth is controlled by the hyperparameter T , which caps the number of agent loops. As shown in Tab. 7, performance improves steadily from the baseline (no agent) to $T=4$, where accuracy and score largely saturate while keeping generation time reasonable. Increasing T beyond 4 yields marginal gains at the cost of additional latency, indicating a practical efficiency–performance trade-off. At the same time, this highlights our method’s compute flexibility: users with ample resources can obtain higher caption quality and more powerful captioner by increasing T , while $T=4$ remains a well-balanced default. Moreover, the further comparison of overhead will be provided in the Appendix.

Compute-normalized comparison

with Best-of-N. To validate that our self-reflection mechanism’s improvements derive from directed reasoning rather than simply scaling test-time compute [23, 83], we compare it against a Best-of-N (BoN) baseline under aligned computational budgets (average inference time per video). As shown in Tab. 8, scaling N in BoN yields marginal gains over the greedy baseline. Conversely, our Baseline+P+R ($T = 4$) achieves 47.10% accuracy and a 2.38 score in 164.9s, significantly outperforming BoN ($N = 12$), which reaches only 45.12% and 2.30 despite requiring more time (186.0s). This confirms that our agentic self-reflection provides a highly efficient, directed optimization path that is fundamentally superior to the undirected, trial-and-error approach of BoN.

Table 8: Compute-normalized comparison. Baseline: Qwen2.5-VL-7B. BoN: Best-of-N. P+R: Test-time Reflection.

Method	Acc.	Score	Time
Baseline (Greedy)	43.95	2.23	15.5s
Baseline (BoN, $N = 6$)	44.88	2.28	93.0s
Baseline (BoN, $N = 9$)	45.04	2.29	139.5s
Baseline (BoN, $N = 12$)	45.12	2.30	186.0s
Baseline+P+R ($T = 2$)	46.53	2.35	87.4s
Baseline+P+R ($T = 3$)	<u>46.86</u>	<u>2.36</u>	131.7s
Baseline+P+R ($T = 4$)	47.10	2.38	164.9s

4.4 Qualitative Results

Fig. 4 presents a qualitative comparison between VDC-Agent-7B and strong baselines on a greenhouse video. While Qwen2.5-VL-7B recognizes the plant-measuring activity, it hallucinates unsupported details (*e.g.*, “tomato plants”). Video-R1-7B and Cockatiel-13B provide more elaborate descriptions but omit crucial background elements (*e.g.*, tray arrangements) and camera motion (*e.g.*, the smooth pan along the aisle as the researchers walk and measure the plants).

In contrast, VDC-Agent-7B produces a more structured and comprehensive narrative. It better captures scene layout, entity interactions, and the activity’s intent, alongside depicting subtle attributes like spatial organization and camera motion. These observations align with our quantitative results, confirming that

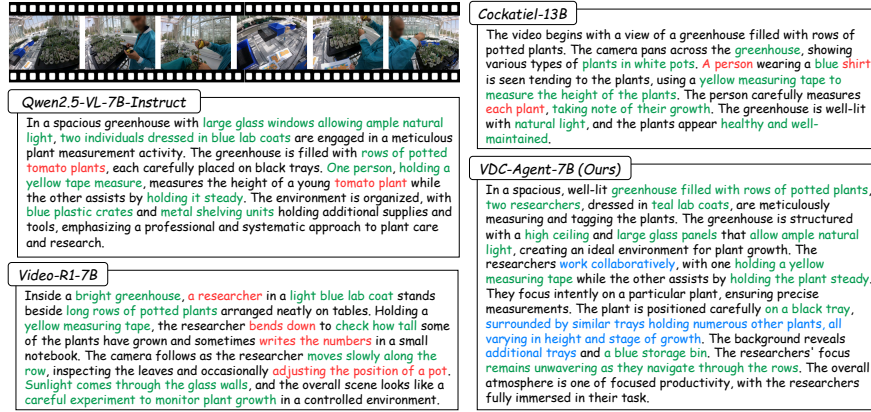


Fig. 4: Qualitative comparison on a video from the VDC benchmark. We generate captions with different captioners under the same prompt: “Please provide a detailed description of the given video.” Text in green denotes correct, video-grounded descriptions, while red indicates incorrect or hallucinated content. Blue highlights fine-grained details that are correctly identified only by our VDC-Agent-7B.

agentic self-reflection enables the generation of richer, more faithful, and better-grounded detailed captions.

5 Conclusion

We propose VDC-Agent, which upgrades off-the-shelf MLLMs for detailed video captioning via iterative self-reflection. By leveraging self-generated preference pairs ordered by difficulty, our approach effectively aligns models to produce precise descriptions. VDC-Agent-7B establishes a new state-of-the-art, outperforming prior baselines while maintaining efficient inference.

Future Work. We will extend our approach to larger backbones (e.g., Qwen-32B) and broader video understanding tasks (e.g., video question answering) to further investigate the performance ceilings, scalability, and universality of the proposed agentic framework.

Acknowledgements

This work is supported by the National Natural Science Foundation of China under Grant 62576224 and 62302382, the Guangdong Provincial Key Laboratory of Computility Microelectronics 2024B1212010007, the Guangdong Basic and Applied Basic Research Foundation Project 2026A1515011733, the Shenzhen Key Technical Projects under Grant CJGJZD20220517141605013, JCYJ20220818101406014 and JSGG20220831105801004, China Postdoctoral Science Foundation 2024M752584, and Natural Science Foundation of Shaanxi Province 2024JC-YBQN-0637.

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
2. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., et al.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)
3. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al.: Qwen2.5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
4. Bai, Y., Kadavath, S., Kundu, S., Askell, A., Kernion, J., Jones, A., Chen, A., Goldie, A., Mirhoseini, A., McKinnon, C., et al.: Constitutional ai: Harmlessness from ai feedback. arXiv preprint arXiv:2212.08073 (2022)
5. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: Proceedings of the 26th annual international conference on machine learning. pp. 41–48 (2009)
6. Chai, W., Song, E., Du, Y., Meng, C., Madhavan, V., Bar-Tal, O., Hwang, J.N., Xie, S., Manning, C.D.: Auroracap: Efficient, performant video detailed captioning and a new benchmark. arXiv preprint arXiv:2410.03051 (2024)
7. Chen, L., Wei, X., Li, J., Dong, X., Zhang, P., Zang, Y., Chen, Z., Duan, H., Tang, Z., Yuan, L., et al.: Sharegpt4video: Improving video understanding and generation with better captions. *Advances in Neural Information Processing Systems* **37**, 19472–19495 (2024)
8. Chen, X., Zhang, Y., Rao, C., Guan, Y., Liu, J., Zhang, F., Song, C., Liu, Q., Zhang, D., Tan, T.: Vidcapbench: A comprehensive benchmark of video captioning for controllable text-to-video generation. In: Findings of the Association for Computational Linguistics: ACL 2025. pp. 8543–8563 (2025)
9. Chen, Z., Wang, W., Cao, Y., Liu, Y., Gao, Z., Cui, E., Zhu, J., Ye, S., Tian, H., Liu, Z., et al.: Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. arXiv preprint arXiv:2412.05271 (2024)
10. Chen, Z., Deng, Y., Yuan, H., Ji, K., Gu, Q.: Self-play fine-tuning converts weak language models to strong language models. arXiv preprint arXiv:2401.01335 (2024)
11. Dang, J., Zhang, Y., Ye, H., Wang, T., Chen, S., Zheng, H., Guo, Y., Lai, J., Hu, B.: Synpo: Synergizing descriptiveness and preference optimization for video detailed captioning. arXiv preprint arXiv:2506.00835 (2025)
12. Ding, C., Dong, S., Zhou, Z., Han, J., Wang, Q., He, Y., Gong, Y.: Class-independent increment: An efficient approach for multi-label class-incremental learning. arXiv preprint arXiv:2503.00515 (2025)
13. Ding, C., Gao, X., Dong, S., He, Y., Wang, Q., Kot, A., Gong, Y.: Lobg: less overfitting for better generalization in vision-language model. arXiv preprint arXiv:2410.10247 (2024)
14. Ding, C., Gao, X., Dong, S., He, Y., Wang, Q., Song, X., Kot, A., Gong, Y.: Space rotation with basis transformation for training-free test-time adaptation. arXiv preprint arXiv:2502.19946 (2025)
15. Dong, S., Ding, C., Li, J., Han, J., Wang, Q., He, Y., Gong, Y.: Beyond clip generalization: Against forward&backward forgetting adapter for continual learning of vision-language models. arXiv preprint arXiv:2505.07690 (2025)

16. Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Yang, A., Fan, A., et al.: The llama 3 herd of models. arXiv e-prints pp. arXiv-2407 (2024)
17. Fan, Y., Ma, X., Wu, R., Du, Y., Li, J., Gao, Z., Li, Q.: Videoagent: A memory-augmented multimodal agent for video understanding. In: European Conference on Computer Vision. pp. 75–92. Springer (2024)
18. Feng, K., Gong, K., Li, B., Guo, Z., Wang, Y., Peng, T., Wu, J., Zhang, X., Wang, B., Yue, X.: Video-r1: Reinforcing video reasoning in mllms. arXiv preprint arXiv:2503.21776 (2025)
19. Gabeur, V., Sun, C., Alahari, K., Schmid, C.: Multi-modal transformer for video retrieval. In: European Conference on Computer Vision. pp. 214–229. Springer (2020)
20. Gao, X., Dong, S., He, Y., Wang, Q., Gong, Y.: Beyond prompt learning: Continual adapter for efficient rehearsal-free continual learning. In: European Conference on Computer Vision. pp. 89–106. Springer (2024)
21. Google DeepMind, Gemini Team: Gemini 3.1 pro (2026)
22. Guadarrama, S., Krishnamoorthy, N., Malkarnenkar, G., Venugopalan, S., Mooney, R., Darrell, T., Saenko, K.: Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In: Proceedings of the IEEE international conference on computer vision. pp. 2712–2719 (2013)
23. Han, J., Ding, C., Dong, S., Gao, X., Wang, Q., He, Y., Gong, Y.: Unleashing the potential of all test samples: Mean-shift guided test-time adaptation. IEEE Transactions on Circuits and Systems for Video Technology (2026)
24. Han, J., Ding, C., Dong, S., He, Y., Wang, S., Wang, Q., Gong, Y.: Goal: Geometrically optimal alignment for continual generalized category discovery. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 40, pp. 4565–4573 (2026)
25. Han, J., Ding, C., He, Y., Dong, S., Wang, Q., Gao, X., Gong, Y.: Learn by reasoning: Analogical weight generation for few-shot class-incremental learning. IEEE Transactions on Circuits and Systems for Video Technology (2025)
26. Han, J., Ding, C., He, Y., Wang, Q., Wang, S., Dong, S., Gong, Y.: Learning like humans: Analogical concept learning for generalized category discovery. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 28734–28743 (2026)
27. Han, J., Wang, S., He, Y., Ding, C., Wang, Q., Gao, X., Dong, S., Gong, Y.: Consistent supervised-unsupervised alignment for generalized category discovery. Advances in Neural Information Processing Systems **38**, 131223–131253 (2026)
28. Han, M., Yang, L., Chang, X., Wang, H.: Shot2story20k: A new benchmark for comprehensive understanding of multi-shot videos. arXiv preprint arXiv:2312.10300 (2023)
29. Hong, W., Ding, M., Zheng, W., Liu, X., Tang, J.: Cogvideo: Large-scale pretraining for text-to-video generation via transformers. arXiv preprint arXiv:2205.15868 (2022)
30. Hong, W., Wang, W., Lv, Q., Xu, J., Yu, W., Ji, J., Wang, Y., Wang, Z., Dong, Y., Ding, M., et al.: Cogagent: A visual language model for gui agents. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14281–14290 (2024)
31. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. Iclr **1**(2), 3 (2022)

32. Huang, X., Liu, W., Chen, X., Wang, X., Wang, H., Lian, D., Wang, Y., Tang, R., Chen, E.: Understanding the planning of llm agents: A survey. *arXiv preprint arXiv:2402.02716* (2024)
33. Islam, M.M., Ho, N., Yang, X., Nagarajan, T., Torresani, L., Bertasius, G.: Video recap: Recursive captioning of hour-long videos. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 18198–18208 (2024)
34. Krishnamoorthy, N., Malkarnenkar, G., Mooney, R., Saenko, K., Guadarrama, S.: Generating natural-language video descriptions using text-mined knowledge. In: *Proceedings of the AAAI conference on artificial intelligence*. vol. 27, pp. 541–547 (2013)
35. Lei, J., Yu, L., Bansal, M., Berg, T.: Tvqa: Localized, compositional video question answering. In: *Proceedings of the 2018 conference on empirical methods in natural language processing*. pp. 1369–1379 (2018)
36. Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., et al.: Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326* (2024)
37. Li, J., Ding, C., Dong, S., Wang, Q., Zhao, J., He, Y., Gong, Y.: Is parameter isolation better for prompt-based continual learning? *arXiv preprint arXiv:2601.20894* (2026)
38. Li, J., Wan, C., Dong, S., Ding, C., Wang, Q., Ma, Z., Gong, Y.: Trajectory-diversity-driven robust vision-and-language navigation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 9143–9154 (2026)
39. Li, J., Wang, S., Qian, B., He, Y., Wei, X., Wang, Q., Gong, Y.: Dynamic integration of task-specific adapters for class incremental learning. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. pp. 30545–30555 (2025)
40. Li, K., He, Y., Wang, Y., Li, Y., Wang, W., Luo, P., Wang, Y., Wang, L., Qiao, Y.: Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355* (2023)
41. Li, X., Wang, Y., Yu, J., Zeng, X., Zhu, Y., Huang, H., Gao, J., Li, K., He, Y., Wang, C., et al.: Videochat-flash: Hierarchical compression for long-context video modeling. *arXiv preprint arXiv:2501.00574* (2024)
42. Li, Y., Wang, C., Jia, J.: Llama-vid: An image is worth 2 tokens in large language models. In: *European Conference on Computer Vision*. pp. 323–340. Springer (2024)
43. Lian, L., Ding, Y., Ge, Y., Liu, S., Mao, H., Li, B., Pavone, M., Liu, M.Y., Darrell, T., Yala, A., et al.: Describe anything: Detailed localized image and video captioning. *arXiv preprint arXiv:2504.16072* (2025)
44. Lin, B., Ye, Y., Zhu, B., Cui, J., Ning, M., Jin, P., Yuan, L.: Video-llava: Learning united visual representation by alignment before projection. In: *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. pp. 5971–5984 (2024)
45. Liu, Y., Hou, H., Ma, F., Ni, S., Yu, F.R.: Mllm-ta: Leveraging multimodal large language models for precise temporal video grounding. *IEEE Signal Processing Letters* (2024)
46. Maaz, M., Rasheed, H., Khan, S., Khan, F.: Video-chatgpt: Towards detailed video understanding via large vision and language models. In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. pp. 12585–12602 (2024)

47. Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegrefe, S., Alon, U., Dziri, N., Prabhume, S., Yang, Y., et al.: Self-refine: Iterative refinement with self-feedback. *Advances in neural information processing systems* **36**, 46534–46594 (2023)
48. Meng, D., Huang, R., Dai, Z., Li, X., Xu, Y., Zhang, J., Huang, Z., Zhang, M., Zhang, L., Liu, Y., et al.: Videocap-r1: Enhancing mllms for video captioning via structured thinking. *arXiv preprint arXiv:2506.01725* (2025)
49. Nan, K., Xie, R., Zhou, P., Fan, T., Yang, Z., Chen, Z., Li, X., Yang, J., Tai, Y.: Openvid-1m: A large-scale high-quality dataset for text-to-video generation. *arXiv preprint arXiv:2407.02371* (2024)
50. OpenAI: Gpt-5.4 large language model (2026)
51. Pattnaik, P., Maheshwary, R., Ogueji, K., Yadav, V., Madhusudhan, S.T.: Enhancing alignment using curriculum learning & ranked preferences. In: *Findings of the Association for Computational Linguistics: EMNLP 2024*. pp. 12891–12907 (2024)
52. Qasim, I., Horsch, A., Prasad, D.: Dense video captioning: A survey of techniques, datasets and evaluation protocols. *ACM Computing Surveys* **57**(6), 1–36 (2025)
53. Qin, L., Tan, Z., Yang, M., Yang, X., Li, H.: Cockatiel: Ensembling synthetic and human preferred training for detailed video caption. *arXiv preprint arXiv:2503.09279* (2025)
54. Rafailov, R., Sharma, A., Mitchell, E., Manning, C.D., Ermon, S., Finn, C.: Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems* **36**, 53728–53741 (2023)
55. Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., Yao, S.: Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems* **36**, 8634–8652 (2023)
56. Shutterstock, Inc.: Content review | shutterstock contributor help (2023), <https://submit.shutterstock.com/help/en/collections/11599943-content-review>, accessed 2026-06-29
57. Song, X., He, Y., Li, J., Wang, Q., Gong, Y.: Learning endogenous attention for incremental object detection. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. pp. 30354–30364 (2025)
58. Sun, R., Zhang, Y., Shah, T., Sun, J., Zhang, S., Li, W., Duan, H., Wei, B., Ranjan, R.: From sora what we can see: A survey of text-to-video generation. *arXiv preprint arXiv:2405.10674* (2024)
59. Tang, J., Li, H., Du, Y., Zhao, W.X.: Avc-dpo: Aligned video captioning via direct preference optimization. *arXiv preprint arXiv:2507.01492* (2025)
60. Team, G., Georgiev, P., Lei, V.I., Burnell, R., Bai, L., Gulati, A., Tanzer, G., Vincent, D., Pan, Z., Wang, S., et al.: Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530* (2024)
61. Wang, J., Yuan, L., Zhang, Y., Sun, H.: Tarsier: Recipes for training and evaluating large video description models. *arXiv preprint arXiv:2407.00634* (2024)
62. Wang, Q., He, Y., Dong, S., Gao, X., Wang, S., Gong, Y.: Non-exemplar domain incremental learning via cross-domain concept integration. In: *European Conference on Computer Vision*. pp. 144–162. Springer (2024)
63. Wang, Q., He, Y., Dong, S., Song, X., Han, J., Luo, H., Gong, Y.: Dualcp: Rehearsal-free domain-incremental learning via dual-level concept prototype. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. vol. 39, pp. 21198–21206 (2025)
64. Wang, Q., Song, X., He, Y., Han, J., Ding, C., Gao, X., Gong, Y.: Boosting domain incremental learning: Selecting the optimal parameters is all you need. In: *Proceed-*

- ings of the Computer Vision and Pattern Recognition Conference. pp. 4839–4849 (2025)
65. Wang, X., Zhang, Y., Zohar, O., Yeung-Levy, S.: Videoagent: Long-form video understanding with large language model as agent. In: European Conference on Computer Vision. pp. 58–76. Springer (2024)
 66. Wang, X., Li, C., Wang, Z., Bai, F., Luo, H., Zhang, J., Jojic, N., Xing, E.P., Hu, Z.: Promptagent: Strategic planning with language models enables expert-level prompt optimization. arXiv preprint arXiv:2310.16427 (2023)
 67. Wang, Y., Kordi, Y., Mishra, S., Liu, A., Smith, N.A., Khashabi, D., Hajishirzi, H.: Self-instruct: Aligning language models with self-generated instructions. In: Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers). pp. 13484–13508 (2023)
 68. Wei, H., Tan, Z., Hu, Y., Chen, C.W., Chen, Z.: Longcaptioning: Unlocking the power of long video caption generation in large multimodal models. arXiv preprint arXiv:2502.15393 (2025)
 69. Xu, L., Zhao, Y., Zhou, D., Lin, Z., Ng, S.K., Feng, J.: Pllava: Parameter-free llava extension from images to videos for video dense captioning. arXiv preprint arXiv:2404.16994 (2024)
 70. Yang, D., Huang, S., Lu, C., Han, X., Zhang, H., Gao, Y., Hu, Y., Zhao, H.: Vript: A video is worth thousands of words. *Advances in Neural Information Processing Systems* **37**, 57240–57261 (2024)
 71. Yang, Z., Li, L., Wang, J., Lin, K., Azarnasab, E., Ahmed, F., Liu, Z., Liu, C., Zeng, M., Wang, L.: Mm-react: Prompting chatgpt for multimodal reasoning and action. arXiv preprint arXiv:2303.11381 (2023)
 72. Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K.R., Cao, Y.: React: Synergizing reasoning and acting in language models. In: The eleventh international conference on learning representations (2022)
 73. Yin, C., Cheng, D., Luo, W., Zeng, M., Zhang, S., Wang, N., Gao, X.: Incremental object detection via future-aware decoupled cross-head distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 39935–39944 (2026)
 74. Yin, J., Chen, T., Chen, Y., Pei, G., Shu, X., Yao, Y., Shen, F.: Pca-seg: Revisiting cost aggregation for open-vocabulary semantic and part segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 27633–27643 (2026)
 75. Yin, J., Chen, T., Pei, G., Liu, H., Yao, Y., Nie, L., Hua, X.: Semi-supervised semantic segmentation with multi-constraint consistency learning. *IEEE Transactions on Multimedia* **27**, 6449–6461 (2025)
 76. Yin, J., Jiang, X., Chen, T., Pei, G., Yao, Y., Shen, F., Shen, H.T.: Depmatch: Boosting semi-supervised semantic segmentation by exploring depth difference knowledge. *IEEE Transactions on Image Processing* **35**, 3256–3270 (2026)
 77. Yu, T., Zhang, H., Li, Q., Xu, Q., Yao, Y., Chen, D., Lu, X., Cui, G., Dang, Y., He, T., et al.: Rlaif-v: Open-source ai feedback leads to super gpt-4v trustworthiness. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 19985–19995 (2025)
 78. Yuan, L., Wang, J., Sun, H., Zhang, Y., Lin, Y.: Tarsier2: Advancing large vision-language models from detailed video description to comprehensive video understanding. arXiv preprint arXiv:2501.07888 (2025)
 79. Yuan, W., Pang, R.Y., Cho, K., Li, X., Sukhbaatar, S., Xu, J., Weston, J.E.: Self-rewarding language models. In: Forty-first International Conference on Machine Learning (2024)

80. Zhang, C., Yang, Z., Liu, J., Li, Y., Han, Y., Chen, X., Huang, Z., Fu, B., Yu, G.: Appagent: Multimodal agents as smartphone users. In: Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems. pp. 1–20 (2025)
81. Zhang, Q., Fang, J., Yuan, R., Tang, X., Qi, Y., Zhang, K., Yuan, C.: Weakly supervised temporal action localization via dual-prior collaborative learning guided by multimodal large language models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 24139–24148 (2025)
82. Zhang, Y., Wu, J., Li, W., Li, B., Ma, Z., Liu, Z., Li, C.: Video instruction tuning with synthetic data. arXiv preprint arXiv:2410.02713 (2024)
83. Zhao, J., Ding, C., Dong, S., Li, J., Wang, Q., He, Y., Gong, Y.: Shared & domain self-adaptive experts with frequency-aware discrimination for continual test-time adaptation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 40, pp. 28697–28705 (2026)
84. Zhao, W., Ji, Y., Li, Y., Cheng, D., Han, J., Zhang, D.: Learning task-shared and specific knowledge via mixture-of-experts in generative model for continual learning. Pattern Recognition p. 113504 (2026)
85. Zhong, C., Hou, Q., Zhou, Z., Hao, S., Lu, H., Zhang, Y., Tang, H., Bai, X.: Owlcap: Harmonizing motion-detail for video captioning via hmd-270k and caption set equivalence reward. arXiv preprint arXiv:2508.18634 (2025)
86. Zhong, Y., Ji, W., Xiao, J., Li, Y., Deng, W., Chua, T.S.: Video question answering: Datasets, algorithms and challenges. In: Proceedings of the 2022 conference on empirical methods in natural language processing. pp. 6439–6455 (2022)