SPOTLIGHT: Identifying and Localizing Video Generation Errors Using VLMs

Aditya Chinchure[1,2], Sahithya Ravi[1,2], Pushkar Shukla[3], Vered Shwartz[1,2], and Leonid Sigal[1,2]

[1] University of British Columbia
[2] Vector Institute for AI
[3] Toyota Technological Institute at Chicago
{aditya10,sahiravi,vshwartz,lsigal}@cs.ubc.ca
pushkarshukla@ttic.edu

Abstract. As Text-to-Video (T2V) models progress towards higher visual realism, the artifacts and errors they produce are highly nuanced, fine-grained, and spatio-temporally localized. Vision Language Models (VLMs) are actively being adopted as automatic evaluators for video generation, driven by the promise of their perception and reasoning abilities. Yet, it remains unclear whether they can detect, localize, and explain fine-grained errors in modern high-fidelity video generations. We introduce SPOTLIGHT, a novel benchmark to rigorously assess whether current VLMs can precisely localize and explain nuanced video-generation errors. SPOTLIGHT comprises 600 videos generated by state-of-the-art T2V models (Veo3, Seedance, LTX-2), annotated with over 1,600 fine-grained error localizations and explanations spanning physics, semantics and anatomy. Our experiments reveal that current VLMs lag behind humans significantly, with humans outperforming our best baselines by nearly 2× on the task. Our analysis highlights key areas of improvement for utilizing VLMs as automated video evaluators, including the need for more robust perception and hallucination mitigation. Data and code is available at https://spotlight-video.github.io.

1 Introduction

Video generation has rapidly advanced in recent years, driven by large-scale diffusion and transformer-based architectures. Modern text-to-video (T2V) models such as Veo 3 [7], Seedance [10], and LTX-2 [12] can now produce high-fidelity, temporally coherent, and visually realistic outputs. However, as these models have overcome the obvious, macroscopic artifacts of their predecessors [14, 25], they exhibit a more subtle class of errors that are local and difficult to detect.

These localized errors manifest as momentary inconsistencies in videos, ranging from morphing objects and body parts to interactions that violate physical commonsense. Consider the example in Fig. 1, where a baby kangaroo hops out of its mother's pouch. While this video generated by Seedance looks aesthetically pleasing, the kangaroo's pouch appears to be made of *fabric*, and it *disappears*

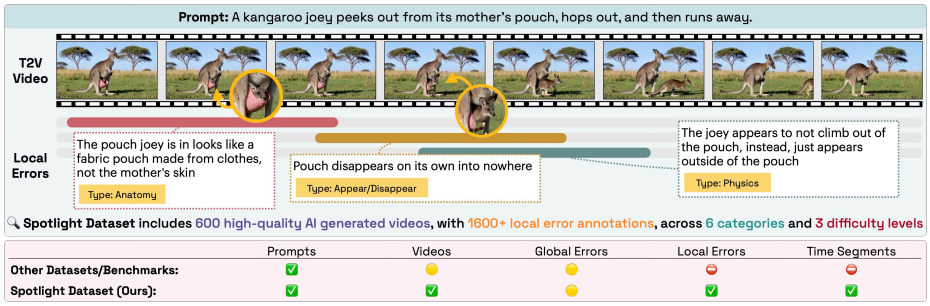

Fig. 1: We introduce SPOTLIGHT, a novel benchmark for evaluating VLMs on precisely localizing and explaining errors in AI-generated videos. We provide 1600+ fine-grained, temporally localized error annotations across 6 error categories and 3 difficulty levels, and task VLMs to identify these errors.

as the baby kangaroo runs away. Such inconsistencies are challenging to detect because they are transient, context-dependent, and seem visually coherent. Despite their subtlety, they can disrupt human immersion, often forcing viewers to instinctively rewatch and scrutinize the problematic moment and understand what went wrong.

Vision-Language Models (VLMs) offer significant promise as automated evaluators for generated videos, and emerging work has actively begun exploring their potential in this domain [13, 16, 30]. However, these initial efforts have not investigated whether VLMs can reliably detect *the subtle, localized inconsistencies* that currently plague high-quality video generation. Bridging this gap is crucial for two reasons. First, it provides a rigorous framework to evaluate the capacity of VLMs to pinpoint precise errors, testing the true limits of their **fine-grained perception and reasoning** abilities. Second, VLMs that demonstrate proficiency on precisely detecting such errors can subsequently be deployed as **robust reward models** to optimize future video generation architectures.

To that end, we introduce SPOTLIGHT, a novel benchmark to evaluate the proficiency of VLMs in localizing and explaining fine-grained errors within high-quality generated videos. SPOTLIGHT focuses on identifying *when* and *why* T2V models fail, demanding interpretable and local explanations for errors in videos. To construct SPOTLIGHT, we collect high-quality human annotations for 1,604 errors across 600 videos. The videos were generated using 200 diverse prompts, including both creative and realistic content, using three state-of-the-art T2V models (Veo 3, Seedance and LTX-2). The errors are categorized into six classes that target the three fundamental error types of physics, semantics, and anatomy as shown in Fig. 2. Crucially, each error is annotated with precise temporal localization (start and end timestamps), a rationale explaining the error, and a severity score reflecting the error's prominence.

We benchmark a suite of open and closed-source VLMs on SPOTLIGHT, and comprehensively cover various prompting strategies, including zero-shot, chain-of-thought, and inference-time decomposition baselines. Our evaluation reveals

that while current VLM models identify broad anomalies, they struggle significantly with fine-grained reasoning and precise localization, and demonstrates that humans consistently outperform the best-performing models by a factor of nearly 2×. An error analysis on our strongest zero-shot baseline shows that 58% of errors stem from insufficient video perception and hallucination, underscoring that current VLMs lack the fine-grained video understanding capabilities that SPOTLIGHT demands. These findings highlight a fundamental gap in the out-of-the-box utility of models as automated evaluators of generated video. We envision SPOTLIGHT to guide future work toward VLMs explicitly trained for temporal localization, fine-grained perception, and intuitive physics understanding.

2 Related Work

2.1 Text-to-Video Models

Recent progress in text-to-video (T2V) generation has been driven by advances in large-scale diffusion and transformer-based architectures. The prior generation of T2V models, such as LTX Video [12] and CogVideoX [28], demonstrated significant capabilities but produced videos with notable limitations in quality. However, modern T2V models have achieved substantial improvements. Commercial systems like Google's Veo 3 [7] can now generate high-quality videos with improved prompt adherence and realistic physics. ByteDance's Seedance 1.0 [10] demonstrates excellent capabilities in handling complex scenarios, multi-shot generation, and long-range temporal coherence. Despite their success in generating visually appealing content, such models still exhibit errors in maintaining object identity, realistic motion, and physical plausibility, especially in complex or long-duration scenes [2, 25]. Our work investigates whether these errors can be detected, described, and temporally localized using VLM models.

2.2 Video Generation Evaluation

Evaluating video generation quality requires multiple dimensions of assessment. Early benchmarks used metrics like FVD [23], IS [26], and CLIP-based similarity [22] to measure fidelity, text-video alignment, and motion smoothness. EvalCrafter [17] expanded metrics to include visual question answering and flow-based methods. Other methods, like TRAJAN [1], rely on motion-based evaluation to measure quality, but these methods lack any reasoning capability. VBench [14] and VBench 2.0 [31] expanded evaluation to 18 dimensions, including physics consistency and controllability, to benchmark and rank T2V models. Specialized benchmarks like Physics-IQ [19] and VideoPhy2 [2] assess physical reasoning, and StoryEval [25] measures prompt adherence for complex scenes. Many of these works use VLMs to evaluate videos, but only provide global feedback.

Another line of works have introduced benchmarks and methods for building VLM-based reward models, including VideoRewardBench [30], MJ-Video [3],

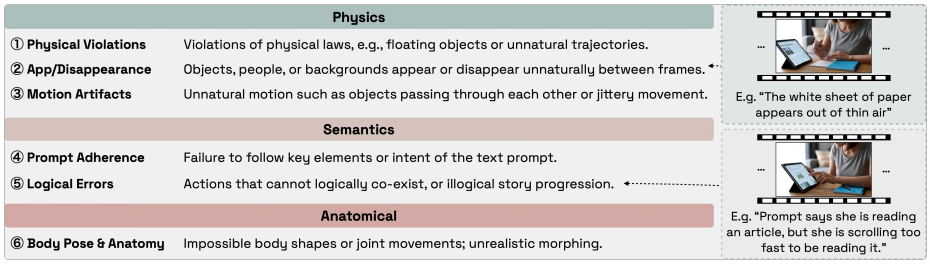

Fig. 2: Taxonomy of error types and categories covered in SPOTLIGHT, with examples.

VideoAlign [16] and Video-Bench [13]. Our work complements these approaches, with the added focus on temporal localization. Recent datasets attempt finer-grained annotation, but address different problems from ours. BrokenVideos [15] annotates videos for low-level artifacts like distortions and implausible object trajectories. DAVID-XR1 [9] provides frame-level annotations with rationales but focuses on detecting AI-generated content rather than understanding generation failures. SPOTLIGHT addresses these gaps by providing temporal localization and natural language explanations of high-level semantic errors that characterize modern high-quality generators.

3 The SPOTLIGHT Task

The SPOTLIGHT task is designed to identify errors in videos generated by T2V models that violate semantic and realism principles, while respecting the creative constraints established by the prompt. While T2V models can synthesize imaginative scenarios that transcend physical constraints (e.g, fantastical creatures or impossible scenes), they must still adhere to fundamental principles of realism within their depicted context. We define an error as a segment of the video where the content exhibits physical implausibilities, anatomical defects, or semantic violations that are not justified by the prompt's creative premise. Our taxonomy, therefore, spans six categories within the aforementioned three error types, as detailed in Fig. 2. These categories encompass the most common failure modes observed in AI-generated videos, and are in-line with existing works [2, 31].

For a given AI-generated video V produced from prompt P with duration d seconds, the goal of SPOTLIGHT is to return a set of errors $E = \{e_1, ...e_i, ...\}$ such that each error $e_i = (t_i^s, t_i^e, c_i, r_i)$ is composed of the start and end times t_i^s and t_i^e, the error category c_i (Fig. 2) and a natural language reasoning for the error, r_i. Two interdependent components are necessary:

Error Localization. We want to determine the temporal boundaries (t_i^s, t_i^e) where each error e_i occurs in the video. Since errors are often localized to specific temporal segments rather than persisting throughout the entire video, precise temporal localization is essential. This component identifies *where* in the temporal dimension the error manifests, enabling targeted analysis. Multiple errors

may have overlapping or disjoint temporal segments, and a single frame may contain multiple distinct errors.

Error Reasoning. For each localized segment (t_i^s, t_i^e), we want to determine the error type c_i and generate the natural language reason r_i. This reason must explain *what* the error is and *why* it is a genuine generation failure rather than an intentional creative choice justified by the prompt P. This r_i must be grounded in the content of the video.

4 The SPOTLIGHT Dataset

To evaluate VLM models on the SPOTLIGHT task, we collected a dataset with 1604 errors, spanning diverse videos from three state-of-the-art T2V models.

4.1 Data Collection

After aggregating a set of diverse prompts, we generated videos using these prompts. We collected detailed human annotations for the errors in these videos.

Prompts. We curated a dataset of 200 prompts from five sources to ensure a wide range of complexity and content. This includes 25 complex prompts from StoryEval [25], 25 prompts from VBench2 [31] that specifically target challenging categories such as "Material", "Mechanics", "Human Anatomy", and "Dynamic Spatial Relationship", and 50 from the VidProM [24] dataset which features creative prompts. For real-world scenes, we obtained 30 prompts from BlackSwan [4] and 70 from LVD-2M [27]. Our prompts vary in level of detail, and provide challenging scenarios for video generation. Prompt selection and filtering steps are outlined in Appendix A.1.

Video Generation. We generated 600 videos from the 200 prompts using three top-performing, state-of-the-art video generation models: Veo 3 Fast [8], Seedance [10], and an open-weight model, LTX-2 [18].

Human Annotation. Every generated video is annotated using a two-step process. Each annotation included the error's start and end times (in seconds), a textual description, and a category/type label based on our taxonomy. In the first step of annotation, we opened the task to all qualified annotators on the CloudConnect[4] platform and label the full set of 600 videos. To identify high quality annotators, we use two criteria: they identified at least three errors per video, and wrote descriptions of at least 8 words. This practical method allowed us to filter out *potentially poor* annotators and mark *capable* annotators for our task. In the second round, only these selected annotators re-annotated all 600 videos to ensure comprehensive, detailed error documentation. Each annotator was compensated \$1.1 per annotation task, which sums up to \$13 per hour. In addition, we collected a difficulty rating (1-5) for each error annotated. These ratings, albeit subjective, are an approximate measure of how obvious the error is to find in the video. This allowed us to split our dataset into three difficulty

[4] By CloudResearch. Qualification criterion provided in Appendix A.2.

levels: easy (score of 4-5, most identifiable errors), medium (score of 3), and hard (score of 1-2, most challenging to detect errors).

Data Filtering and Merging. To reduce redundancy, we filtered and merged annotations across both rounds. We merged semantically identical errors that overlapped significantly in time (IoU ≥ 0.6) by using an LLM to assess semantic similarity. When merging, we retained the annotation with the more detailed description. This process yielded the final dataset. We provide additional details of the annotation process in Appendix A.2.

Final Dataset Statistics. We obtain a dataset of 1604 error annotations across 600 videos, with 484 videos having three or more errors. The annotations are detailed, with an average error reasoning length of 14.6 words. The average video length is 6.5s, while the average marked error segment is 2.49s, with a mix of local and global errors. For high-quality annotations across both rounds, the inter-annotator agreement over matched errors on the error types is 83%.

Dataset Quality. We sample 100 error annotations from the dataset to perform human validation of the data. Human evaluators rated 93% of error reasons and 99% of segments as correct. Additional analysis is in Appendix A.3.

4.2 Analysis of Video Generators

Duration of Errors. We observe that prompt adherence errors exhibit the longest duration, followed by physical violations and logical errors (see Fig. 3(a)). In contrast, appearance/disappearance, motion, and body anatomy errors tend to be shorter in duration, suggesting they are more local.

Difficulty Levels. A large number of errors (793 out of 1604) are easy to detect (Fig. 3(a)). Medium (396) and Hard (415) errors may be more nuanced and harder to detect, as we analyze later in Sec. 8.

Error Rates. Veo 3 produces the fewest errors (507 annotations), followed closely by Seedance (517). The open-weight model LTX-2 produces more errors (580), highlighting a gap between commercial-only and open-weights models.

Types. We observe that prompt adherence errors are the most frequent, followed by physics and appearance-disappearance errors. This is in line with prior works (e.g. [25]), as current video generation models struggle to generate videos for complex, multi-event prompts. We share additional details in Appendix A.3.

5 Metrics

We define three metrics that measure how well the predicted error $\hat{e}_j$ aligns with the human-annotated ground truth errors e_i. We then use bipartite matching to compute results across errors in the two sets, $\hat{E} = \{\hat{e}_j\}$ and $E = \{e_i\}$.

5.1 Base Metrics

We define the following pairwise metrics to quantify alignment between each ground truth error $e_i = (t_i^s, t_i^e, c_i, r_i)$ and predicted error $\hat{e}_j = (\hat{t}_j^s, \hat{t}_j^e, \hat{c}_i, \hat{r}_j)$.

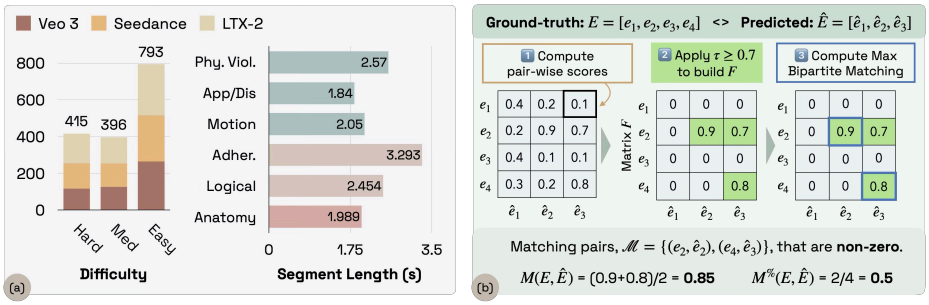

Fig. 3: (a) Analysis of T2V models on SPOTLIGHT data. (b) Example of Pairwise Matching. After filling in matrix F, we compute the best match pairs. M and $M^{\%}$ are computed subsequently.

Precision (P): Measures the fraction of the predicted time segment that overlaps with the ground truth segment:

$$\mathrm{P}(e_i, \hat{e}_j) = \frac{[t_i^s, t_i^e] \cap [\hat{t}_j^s, \hat{t}_j^e]}{(\hat{t}_i^e - \hat{t}_i^s)} \in [0, 1]. \tag{1}$$

Recall (R): Measures the fraction of the ground truth time segment covered by the prediction:

$$\mathrm{R}(e_i, \hat{e}_j) = \frac{[t_i^s, t_i^e] \cap [\hat{t}_j^s, \hat{t}_j^e]}{(t_i^e - t_i^s)} \in [0, 1]. \tag{2}$$

Reason Similarity (S): Measures the semantic similarity between predicted and ground-truth reasons using an LLM evaluator. We use GPT-4o-mini for this task (prompt and analysis is in Appendix A.1). The model rates similarity from 1 to 10, representing how similar the ground truth r_i is to the predicted $\hat{r}_j$,

$$\mathrm{S}(e_i, \hat{e}_j) = \frac{\text{LLM-Match}(r_i, \hat{r}_j) \in [0, 10]}{10} \in [0, 1]. \tag{3}$$

When computing the metrics, we relax the need for a category label $\hat{c}_i$ match, because it is sensitive to how the error is interpreted during annotation. However, we use it to analyze differences between our baseline methods.

5.2 Pairwise Matching Process

To evaluate the set of predicted errors $\hat{E}$ against the ground-truth set E for a given video, we must find an optimal one-to-one mapping between them. We formulate this as a maximum-weight bipartite matching problem, as shown in Fig. 3(b). This process consists of three steps: (1) constructing a pairwise score matrix, (2) solving for the optimal matching, and (3) calculating the final score. **Pairwise Score Matrix.** For a given metric $\mathrm{B} \in \{\mathrm{P}, \mathrm{R}, \mathrm{S}\}$, we construct pairwise matrix, $F_{i,j}$, where each entry represents the alignment between errors e_i

and $\hat{e}_j$ as shown in Fig. 3(b). To filter weak matches, we apply a threshold τ below which pairs are considered invalid. We build the pairwise matrix as,

$$F_{i,j} = \begin{cases} 0 & \text{if } \mathrm{B}(e_i, \hat{e}_j) < \tau \\ \mathrm{B}(e_i, \hat{e}_j) & \text{if } \mathrm{B}(e_i, \hat{e}_j) \geq \tau \end{cases}. \tag{4}$$

However, since identifying an error requires both precise temporal localization and accurate reasoning, we also define a stricter matching criterion that combines precision and semantic similarity. For our task, because the ground truth segments are generally small (avg. 2.49s), precisely marking errors is more important than recall over the marked region. We consider a prediction accurate only when both $P_{i,j} = \mathrm{P}(e_i, \hat{e}_j)$ and $S_{i,j} = \mathrm{S}(e_i, \hat{e}_j)$ exceed threshold τ:

$$F_{i,j} = \begin{cases} 0 & \text{if } P_{i,j} < \tau \vee S_{i,j} < \tau \\ (P_{i,j} + S_{i,j})/2 & \text{otherwise} \end{cases}. \tag{5}$$

For simplicity, we call this the S+P metric when reporting results.

Optimal Matching. We then solve for the maximum-weight bipartite matching (refer to Appendix B.2) on F to find the optimal set of matched pairs $\mathcal{M}$, only keeping non-zero pairs. This set maximizes the total alignment score $\sum_{(e_i,\hat{e}_j)\in\mathcal{M}} F_{i,j}$.

Final Scores. From the optimal matching $\mathcal{M}$, we compute the final scores, $M(E, \hat{E})$ for video V is the average over matched pairs as show in Fig. 3(b):

$$M(E, \hat{E}) = \frac{1}{|\mathcal{M}|} \sum_{(e_i,\hat{e}_j)\in\mathcal{M}} F_{i,j}. \tag{6}$$

We also report coverage, the fraction of ground truth errors successfully matched:

$$M^{\%}(E, \hat{E}) = |\mathcal{M}|/|E|. \tag{7}$$

6 Experimental Methods

To test the capabilities of VLMs on SPOTLIGHT, we consider state-of-the-art VLMs capable of understanding videos. Among open-weight models, we evaluate Qwen3-VL [21], Molmo 2 [5], MiniCPM-V-4.5 [29] and InternVL3 [32]. Gemini 2.5 Pro [6], Gemini 3 Pro [11], and GPT-5.1 [20] provide us with closed-source performance bounds. In addition to zero-shot performance on SPOTLIGHT, we investigate a suite of inference-time baselines, as detailed below. We also compare against two non-VLM baselines for localization, optical flow and CLIP similarity. Prompts for all VLM baselines and additional details are in Appendix C.

6.1 Zero-shot

We begin by evaluating the models' zero-shot localization and reasoning capabilities. The model receives the full video and is prompted to produce a JSON list

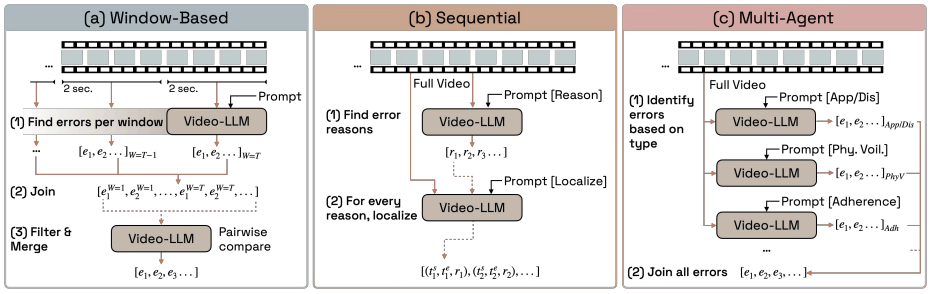

Fig. 4: Inference-time Baselines. Apart from zero-shot and CoT approaches, we experiment with three other baseline strategies.

of detected errors. Each entry includes a timestamp interval localizing the error, the error type category, and a short reason describing the error. Since generated videos are likely out-of-distribution (OOD) for VLMs, this setting represents a challenging evaluation of zero-shot generalization.

6.2 Inference-Time Baselines

Chain of Thought (CoT). We extend the zero-shot baseline to a Chain-of-Thought approach. In this strategy, the VLM is required to first analyze frames and identify anomalies, before arriving at the list of errors.

Sliding Window (SW). To improve temporal resolution of VLMs, we design a sliding-window baseline as shown in Fig. 4(a). We partition each video into 2-second sub-clips and evaluate each sub-clip independently at 4 fps. The prompt explicitly specifies which sub-clip of the original video is being analyzed. This approach trades-off global context for improved frame density per window, allowing the model to detect more localized and transient errors. We then consolidate predictions across all sub-clips by mapping detected error segments to the global video timeline. To handle redundant detections, we use the same VLM in text-only mode to identify semantically equivalent error descriptions. Matching errors are merged by taking the union of their temporal intervals, with this process applied transitively until convergence. This gives us a unified set of error detections spanning the full video.

Sequential (Seq): Reason then Localize. Inspired by how humans might typically identify errors, the *Reason then Localize* baseline decomposes the task into two sequential phases as shown in Fig. 4(b). First, a reasoning stage scans the entire video (2 fps) to identify what errors occurred and why, compiling a list without timestamps. Next, a localization stage takes each of these identified errors as a query. The model re-processes the video to find the precise temporal boundaries (start and end times) for that specific error. This design deliberately isolates the reasoning task from localization.

Multi-Agent (MA). This baseline adopts a divide and conquer strategy by evaluating each of the six error types independently. We query the model six

times, once for each error category, using the full video in a zero-shot setting. After obtaining all six outputs, we combine them into a single list. By narrowing the focus to one error type at a time, this baseline encourages more deliberate reasoning for specific error categories.

6.3 Experimental Setup

We evaluate both Qwen3-VL-8B-Instruct and Qwen3-VL-30B-A3B-Instruct for the Qwen3-VL family. For Molmo2, InternVL3 and MiniCPM-V-4.5, we use the 8B variants of these models. All open-weight models are evaluated on up to 2 NVIDIA H100 80GB GPUs. Closed-source models use their respective official APIs (from Google and OpenAI). Since GPT-5.1 is the only model that does not support video as an input, we uniformly sample frames and feed it to the model in a multi-frame setting. Prompts, generation configurations and runtimes are in Appendix C. We also report human performance by crowd-sourcing expert annotations for a random subset of 80 videos (see Appendix C.4).

We report results at $\tau = 0.7$ for all metrics, Similarity, Precision and Recall, as well as the final score with Similarity and Precision together (S+P). For all metrics, higher is better, indicating better alignment between predictions and ground-truth annotations.

7 Results

Table 1 summarizes the performance of different baselines.

Non-VLM Baselines. Optical flow and CLIP-based non-VLM baselines demonstrate moderate precision, but very low recall, failing to detect semantically salient errors. This demonstrates the need for VLMs to perform our task, as it requires models to reason about the contents in the video.

Zero-shot Performance. Among all models, **Gemini 2.5 Pro achieves the strongest zero-shot performance**, obtaining an S+P score of 0.250 and correctly matching 12.6% of errors with the ground truth. Gemini 3 Pro performs nearly identically, showing only a minor decrease in performance with a 12.1% match, demonstrating no improvement from its predecessor on S+P score. It does, however, achieve strong precision (0.807), suggesting that this model is more capable at accurate timestamp generation for localization. Gemini models perform nearly twice as high compared to Qwen-3-VL-8B-Instruct and Qwen-3-VL-30B-A3B-Instruct. Other open-weight models perform substantially worse. MiniCPM-V-4.5 is optimized for efficiency, with compact encoding for images and videos, which may have led to degraded performance on SPOTLIGHT. Molmo 2, despite having object localization and pointing capabilities [5], fails to improve upon Qwen-3-VL-8B-Instruct. InternVL3, on the other hand, is an older VLM based on the Qwen2.5 LLM architecture. The only model that approaches the performance of the Gemini family is GPT-5.1, further underscoring the current advantage of closed-source models in this setting. This trend continues when only computing pairwise matching with similarity (S) and precision (P), although the

		S+P		Similarity (S)		Precision (P)		Recall (R)	
@ Threshold 0.7	**FPS**	M	$M^{\%}$	M	$M^{\%}$	M	$M^{\%}$	M	$M^{\%}$
Non-VLM Baselines									
Optical Flow	24	-	-	-	-	0.462	32.2%	0.049	3.0%
CLIP Sim.	2	-	-	-	-	0.663	38.1%	0.029	1.2%
Zero-Shot Baselines									
InternVL3-8B	2	0.041	2.2%	0.150	8.6%	0.408	22.8%	0.593	35.8%
MiniCPM-V4.5 8B	2	0.118	5.7%	0.248	14.0%	0.621	34.5%	0.530	28.6%
Molmo 2 8B	2	0.087	5.0%	0.291	19.7%	0.443	26.3%	0.845	74.8%
Qwen3 8B-I	2	0.148	8.0%	0.352	23.3%	0.472	27.1%	0.903	73.1%
Qwen3 30B-A3B-I	2	0.137	6.1%	0.339	21.5%	0.510	29.9%	0.910	**73.9%**
GPT 5.1	2	0.218	11.6%	0.340	23.3%	0.506	33.6%	0.599	43.5%
Gemini 2.5 Pro	2	0.250	12.6%	0.422	26.7%	0.691	44.3%	0.836	68.4%
Gemini 3 Pro	2	0.250	12.1%	0.387	24.1%	0.807	56.3%	0.681	45.9%
Inference-Time Baselines									
Qwen3 8B-I - CoT	2	0.163	7.9%	0.281	17.4%	0.617	40.2%	0.630	48.7%
Qwen3 8B-I - SW	4	0.189	10.0%	0.386	24.6%	0.664	41.8%	**0.931**	67.6%
Qwen3 8B-I - Seq	2	0.189	9.3%	0.394	25.7%	0.645	39.4%	0.822	60.9%
Qwen3 8B-I - MA	2	0.254	13.3%	0.405	30.6%	0.675	49.8%	0.767	61.7%
Gemini 2.5P - MA	2	0.308	17.0%	0.514	33.8%	0.812	**57.7%**	0.829	67.6%
Human									
Human	-	**0.508**	**26.8%**	**0.599**	**34.4%**	**0.840**	48.5%	0.880	50.0%

Table 1: Comparing VLM methods on SPOTLIGHT. For all metrics, higher is better (↑). We bold best performing method and underline the best method within each section. We observe that Gemini 2.5 Pro is the strongest zero-shot method, while still lagging behind humans by half. Among our inference-time strategies, MA performs the best, matching Gemini 2.5 Pro with a significantly smaller parameter model.

improvements are narrower. It is important to note that S+P is a more challenging but more important metric, since it requires both error reasoning and temporal localization to meet the threshold simultaneously, indicating a complete match with the ground truth annotation.

Finally, we observe that recall (R) is high for all models except InternVL3-8B and MiniCPM-V-4.5. While this may look like a positive result, a high recall value indicates a failure of localization in models like Qwen3, which often **predict the entire duration of the clip as the erroneous segment**. We show this error in qualitative results (Sec. 8). Comparing zero-shot performance to human performance, we observe that humans perform 2× better (0.25 versus 0.5) on the S+P metric, highlighting a clear gap from human-level understanding.

Inference-time Baselines. We observe that all inference-time baselines improve performance over zero-shot Qwen3 and Gemini 2.5 Pro baselines. Notably, the multi-agent (MA) approach is effective, closely matching (S and P) or surpassing (S+P) the zero-shot performance of Gemini 2.5 Pro on SPOTLIGHT with the smaller Qwen-3 model, and improving even further when applied to the Gemini 2.5 Pro model. On Qwen-3, this multi-agent approach achieves the highest

performance among inference-time baselines in precision (49.8%), and a modest decrease in recall compared to the base model. The chain-of-thought approach proves to be the least successful, although it decreases recall from 73.1% to 48%, which is in-line with humans. The sequential approach yields only minor improvements. As identifying and localizing errors are closely related, disentangling them may prove to be challenging for models. Finally, we observe that the window-based approach achieves very high recall scores. This could be an artifact of merging errors across windows through semantic similarity matching of errors. Ultimately, the multi-agent baseline on Gemini 2.5 Pro is the strongest VLM performance we achieve on SPOTLIGHT.

Human. Although humans perform SPOTLIGHT significantly better than models, they still do not achieve very high M or $M^{\%}$ scores against ground-truth error annotations. This is likely because the SPOTLIGHT dataset contains error annotations from multiple rounds, combining the efforts of multiple human annotators in identifying and localizing errors. As each person approaches this task with a slightly different lens due to cognitive biases, it is likely that they will identify different errors. Additionally, no single human annotator would write a large number of errors, as it is tedious. These reasons explain the low scores. However, these scores do not reflect negatively on the dataset quality; the quality of our annotations is independently measured and reported in Sec. 4.

Additional Results. We study the effects of different threshold values (τ), the impact of varying frame sampling rates, and an ablation with oracle (ground-truth) timestamps to independently evaluate reasoning capability of VLMs in Appendix C.3.

8 Analysis

To explain our findings on SPOTLIGHT, we conduct a series of qualitative and quantitative analyses, and explain the failure modes of VLM models.

8.1 Qualitative Examples

Figure 5 presents qualitative samples of model predictions. (a) shows Qwen3-MA achieving perfect matching $M^{\%} = 100\%$, with predicted errors exhibiting high similarity scores to ground truth error reasons and with precise temporal localization. Task decomposition enables more comprehensive multi-error detection. However, our final score M accurately captures that the predicted reasons are not an exact match, with Qwen3-MA predicting conflicting statements about the plant being knocked over (marked in red). Human baseline descriptions align with ground truth explanations, but we observe that human annotators partially disagree on the exact localization of the errors.

Figure 5(b) reveals the limitations of current VLMs through a failure case where all models miss the primary error in a relatively simple scene. The ground truth annotation identifies that a large portion of the car is invisible throughout

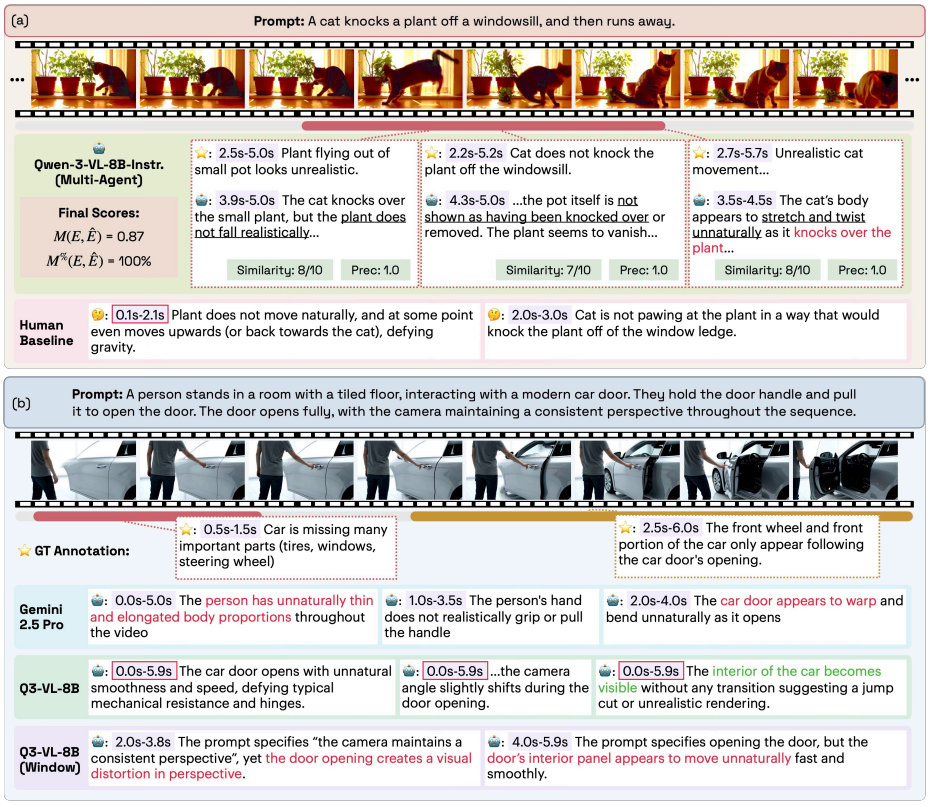

Fig. 5: Qualitative Examples. (a) A video where Qwen3-MA is able to detect all ground truth errors, achieving $M^{\%} = 100\%$, and Human baseline reasons that describe similar failures. (b) Model failures due to incorrect error reasoning and imprecise localization (highlighted in red). Not all errors shown due to space constraints.

most of the video, only appearing after the door opens. Gemini 2.5 Pro incorrectly focuses on the person's body (which appears normal), while Qwen3 fails at temporal localization entirely, marking the full video in each error. The window-based approach narrows the error segments in this case, but fails to identify any errors correctly.

8.2 Analysis on Errors

What are common failure modes of models? We analyzed 50 randomly sampled outputs from our best performing zero-shot baseline, Gemini 2.5 Pro, and found that 39 (78%) were incorrect. The primary failure modes were perception issues (29%), hallucination (29%), and incorrect physical/world knowledge (19%). Perception errors often occurred when models failed to compare frames accurately. For example, for a video about "grass swinging in the wind", Gemini

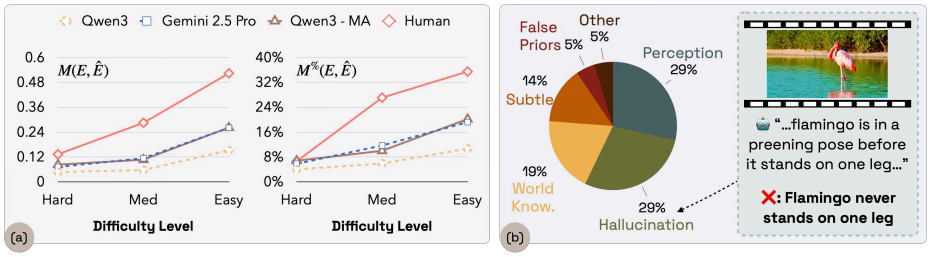

Fig. 6: (a) Difficulty vs. S+P. We observe that more obvious errors are more easily detected by all methods. **(b) Error Analysis.** Gemini 2.5 pro fails across different dimensions, with incorrect perception and hallucinations being the key failure modes.

	Physics						Semantics				Anatomy	
	Phy. Viol.		App/Dis		Motion		Adherence		Logical		Body An.	
	M	M%	M	M%	M	M%	M	M%	M	M%	M	M%
Q3 - MA	0.205	20.3%	0.017	1.5%	0.122	13.0%	0.243	21.4%	0.132	11.4%	0.068	7.0%
G2.5 Pro	0.149	14.4%	0.099	9.0%	0.117	11.1%	0.246	20.8%	0.083	5.9%	0.078	8.3%
Human	0.367	33.9%	0.139	12.1%	0.264	30.0%	0.532	45.0%	0.21	21.4%	0.153	17.6%

Table 2: Analysis across types. We show S+P for all. Both humans and models are stronger at detecting adherence errors over other types.

2.5 says that "the grass remains almost completely static", which is inaccurate because the video shows rapid movements in the blades of the grass that humans can easily perceive. Other times, we observe that hallucinations occur within error reasons. In the video of the flamingo shown in Fig 6(b), the flamingo never stands on one leg, even when the prompt asks the T2V models to do so. Yet, the error produced by Gemini says that the flamingo stands on one leg, which is a hallucination. Furthermore, our error analysis considers how accurate the model localized these errors. We find that 48.7% of the timestamps were accurate, but 35.9% were too wide (i.e. insufficient localization) and 15.4% were too narrow.

How does difficulty influence accuracy? As shown in Fig. 6(a), we observe that more obvious (easy) errors are easier to detect than medium and hard errors for all methods, including humans. Qwen3-MA and Gemini 2.5 Pro show similar improvements across severity levels, while the base Qwen3 lags behind. However, humans show more substantial improvements on medium difficulty errors compared to all models. This performance gap shows that VLMs lag behind human capability for detecting errors that may not be as obvious, but are still apparent to humans. Hard errors are challenging for all methods, including humans, as they may be influenced by cognitive biases from one human to another, or might require significantly more attention to identify.

How does error type influence detection accuracy? Table 2 presents a performance breakdown across six error types. We observe that both models and humans can more reliably detect errors related to prompt adherence over

other types. Qwen3-MA is stronger at detecting physics and logical errors (0.2 and 0.13) than Gemini 2.5 Pro (0.14 and 0.08), whereas Gemini is stronger at detecting appearance and disappearance errors (0.099 vs 0.017).

9 Conclusion

SPOTLIGHT is a novel task to identify and localize fine-grained errors in AI-generated videos. Through comprehensive human annotation of 1,604 errors across 600 videos from three state-of-the-art T2V generators, we create a challenging benchmark to test VLM models as video generation evaluators. Our analysis reveals key limitations of VLM models on SPOTLIGHT such as lacking precise temporal reasoning for detecting subtle and localized inconsistencies, perception errors, and hallucination. Our experiments show humans outperform state-of-the-art models by 2×, and while inference-time strategies can improve performance, a substantial gap remains. We hope that SPOTLIGHT inspires VLM and T2V developers to improve video understanding methods on out-of-distribution generated videos, and create stronger evaluators for video generation.

While we have collected high-quality error annotations over all videos, we acknowledge that our dataset may not cover all possible errors in these videos due to the differences in how humans perceive these videos, along with cognitive biases. Still, SPOTLIGHT remains a challenging benchmark for VLMs, driving progress towards VLMs with stronger temporal understanding and diagnostic capabilities.

Acknowledgments

This work was funded, in part, by the Vector Institute for AI, Canada CIFAR AI Chair, NSERC CRC, NSERC DG and Accelerator Grants. Hardware resources used in preparing this research were provided, in part, by the Province of Ontario, the Government of Canada through CIFAR, companies sponsoring the Vector Institute, and Digital Research Alliance of Canada. Additionally, this work utilizes API credits provided by Google.

References

1. Allen, K., Doersch, C., Zhou, G., Suhail, M., Driess, D., Rocco, I., Rubanova, Y., Kipf, T., Sajjadi, M.S., Murphy, K., et al.: Direct motion models for assessing generated videos. arXiv preprint arXiv:2505.00209 (2025)
2. Bansal, H., Lin, Z., Xie, T., Zong, Z., Yarom, M., Bitton, Y., Jiang, C., Sun, Y., Chang, K.W., Grover, A.: Videophy: Evaluating physical commonsense for video generation. arXiv preprint arXiv:2406.03520 (2024)
3. Chen, Z., Du, Y., Wen, Z., Zhou, Y., Cui, C., Weng, Z., Tu, H., Wang, C., Tong, Z., Huang, Q., et al.: Mj-bench: Is your multimodal reward model really a good judge for text-to-image generation? arXiv preprint arXiv:2407.04842 (2024)

4. Chinchure, A., Ravi, S., Ng, R., Shwartz, V., Li, B., Sigal, L.: Black swan: Abductive and defeasible video reasoning in unpredictable events. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 24201–24210 (2025)
5. Clark, C., Zhang, J., Ma, Z., Park, J.S., Salehi, M., Tripathi, R., Lee, S., Ren, Z., Kim, C.D., Yang, Y., et al.: Molmo2: Open weights and data for vision-language models with video understanding and grounding. arXiv preprint arXiv:2601.10611 (2026)
6. Comanici, G., et al., E.B.: Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities (2025)
7. DeepMind, G.: Veo: Advanced video generation model. Google DeepMind (2024)
8. Deepmind, G.: Veo 3 (2025)
9. Gao, Y., Ding, Y., Su, H., Li, J., Zhao, Y., Luo, L., Chen, Z., Wang, L., Wang, X., Wang, Y., Ma, X., Jiang, Y.G.: David-xr1: Detecting ai-generated videos with explainable reasoning (2025)
10. Gao, Y., Guo, H., Hoang, T., Huang, W., Jiang, L., Kong, F., Li, H., Li, J., Li, L., Li, X., Li, X., Li, Y., Lin, S., Lin, Z., Liu, J., Liu, S., Nie, X.P., Qing, Z., Ren, Y., Sun, L., Tian, Z., Wang, R., Wang, S., Wei, G., Wu, G., Wu, J., Xia, R., Xiao, F., Xiao, X., Yan, J., Yang, C., Yang, J., Yang, R., Yang, T., Yang, Y., Ye, Z.N., Zeng, X., Zeng, Y., Zhang, H., Zhao, Y., Zheng, X., Zhu, P., Zou, J., Zuo, F.: Seedance 1.0: Exploring the boundaries of video generation models. ArXiv **abs/2506.09113** (2025)
11. GoogleDeepMind: Gemini 3 pro model card, model card update: December, 2025 (2025)
12. HaCohen, Y., Chiprut, N., Brazowski, B., Shalem, D., Moshe, D., Richardson, E., Levin, E., Shiran, G., Zabari, N., Gordon, O., Panet, P., Weissbuch, S., Kulikov, V., Bitterman, Y., Melumian, Z., Bibi, O.: Ltx-video: Realtime video latent diffusion (2024)
13. Han, H., Li, S., Chen, J., Yuan, Y., Wu, Y., Deng, Y., Leong, C.T., Du, H., Fu, J., Li, Y., et al.: Video-bench: Human-aligned video generation benchmark. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 18858–18868 (2025)
14. Huang, Z., He, Y., Yu, J., Zhang, F., Si, C., Jiang, Y., Zhang, Y., Wu, T., Jin, Q., Chanpaisit, N., Wang, Y., Chen, X., Wang, L., Lin, D., Qiao, Y., Liu, Z.: VBench: Comprehensive benchmark suite for video generative models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024)
15. Lin, J., Peng, W., Zi, B., Gao, Y., Qi, X., Ma, X., Jiang, Y.G.: Brokenvideos: A benchmark dataset for fine-grained artifact localization in ai-generated videos (2025)
16. Liu, J., Liu, G., Liang, J., Yuan, Z., Liu, X., Zheng, M., Wu, X., Wang, Q., Xia, M., Wang, X., et al.: Improving video generation with human feedback. arXiv preprint arXiv:2501.13918 (2025)
17. Liu, Y., Cun, X., Liu, X., Wang, X., Zhang, Y., Chen, H., Liu, Y., Zeng, T., Chan, R., Shan, Y.: Evalcrafter: Benchmarking and evaluating large video generation models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22139–22149 (2024)
18. LTX: Ltx-2: Production-grade ai video generation model (2025)
19. Motamed, S., Culp, L., Swersky, K., Jaini, P., Geirhos, R.: Do generative video models understand physical principles? arXiv preprint arXiv:2501.09038 (2025)
20. OpenAI: Gpt-5.1: A smarter, more conversational chatgpt, november 12, 2025 (2025)

21. Qwen: Qwen3-vl (2025)
22. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision (2021)
23. Unterthiner, T., van Steenkiste, S., Kurach, K., Marinier, R., Michalski, M., Gelly, S.: FVD: A new metric for video generation (2019)
24. Wang, W., Yang, Y.: Vidprom: A million-scale real prompt-gallery dataset for text-to-video diffusion models. Thirty-eighth Conference on Neural Information Processing Systems (2024)
25. Wang, Y., He, X., Wang, K., Ma, L., Yang, J., Wang, S., Du, S.S., Shen, Y.: Is your world simulator a good story presenter? a consecutive events-based benchmark for future long video generation. arXiv preprint arXiv:2412.16211 (2024)
26. Wu, J.Z., Fang, G., Wu, H., Wang, X., Ge, Y., Cun, X., Zhang, D.J., Liu, J.W., Gu, Y., Zhao, R., Lin, W., Hsu, W., Shan, Y., Shou, M.Z.: Towards a better metric for text-to-video generation (2024)
27. Xiong, T., Wang, Y., Zhou, D., Lin, Z., Feng, J., Liu, X.: Lvd-2m: A long-take video dataset with temporally dense captions. arXiv preprint arXiv:2410.10816 (2024)
28. Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., Yin, D., Zhang, Y., Wang, W., Cheng, Y., Xu, B., Gu, X., Dong, Y., Tang, J.: Cogvideox: Text-to-video diffusion models with an expert transformer (2025)
29. Yu, T., Wang, Z., Wang, C., Huang, F., Ma, W., He, Z., Cai, T., Chen, W., Huang, Y., Zhao, Y., Xu, B., Cui, J., Xu, Y., Ruan, L., Zhang, L., Liu, H., Tang, J., Liu, H., Guo, Q., Hu, W., He, B., Zhou, J., Cai, J., Qi, J., Guo, Z., Chen, C., Zeng, G., Li, Y., Cui, G., Ding, N., Han, X., Yao, Y., Liu, Z., Sun, M.: Minicpm-v 4.5: Cooking efficient mllms via architecture, data, and training recipe (2025)
30. Zhang, Z., Huang, X., Xu, J., Luo, Z., Wang, X., Wei, J., Chen, X.: Videorewardbench: Comprehensive evaluation of multimodal reward models for video understanding. arXiv preprint arXiv:2509.00484 (2025)
31. Zheng, D., Huang, Z., Liu, H., Zou, K., He, Y., Zhang, F., Zhang, Y., He, J., Zheng, W.S., Qiao, Y., Liu, Z.: VBench-2.0: Advancing video generation benchmark suite for intrinsic faithfulness. arXiv preprint arXiv:2503.21755 (2024)
32. Zhu, J., Wang, W., Chen, Z., Liu, Z., Ye, S., Gu, L., Tian, H., Duan, Y., Su, W., Shao, J., et al.: Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. arXiv preprint arXiv:2504.10479 (2025)