PREDICTABILITY IN VIDEO DIFFUSION: EARLY ATTENTION PREDICTS VISUAL FAILURES

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-to-video diffusion models require costly multi-step denoising, yet generation failures are usually identified only after the final video is complete. We find that intermediate video-to-text attention contains early information about whether individual visual constraints will be satisfied. Motivated by this insight, we introduce VIDEOPROBE, a lightweight spatiotemporal probe that predicts token-level failures using only intermediate cross-attention and the denoising timestep. We also construct ATTNPULSE-10K, a benchmark that aligns multi-step attention from 10,000 Wan2.2 trajectories with uncertainty-masked token supervision derived from the completed videos. On Wan2.2, a single probe trained across nine denoising steps achieves .340 mean AUPRC and generalizes without retraining to unseen denoising steps, including the first denoiser forward (.320 AUPRC). Shuffling attention across samples reduces AUPRC to 0.1532 ± 0.0062, supporting the conclusion that matched attention carries predictive information. Independent experiments on Wan2.1 and HunyuanVideo-1.5 further show that early attention predicts final failures across different video generators and attention architectures. These early token-level predictions could support seed routing, prompt refinement alerts, and RL rollout mining.

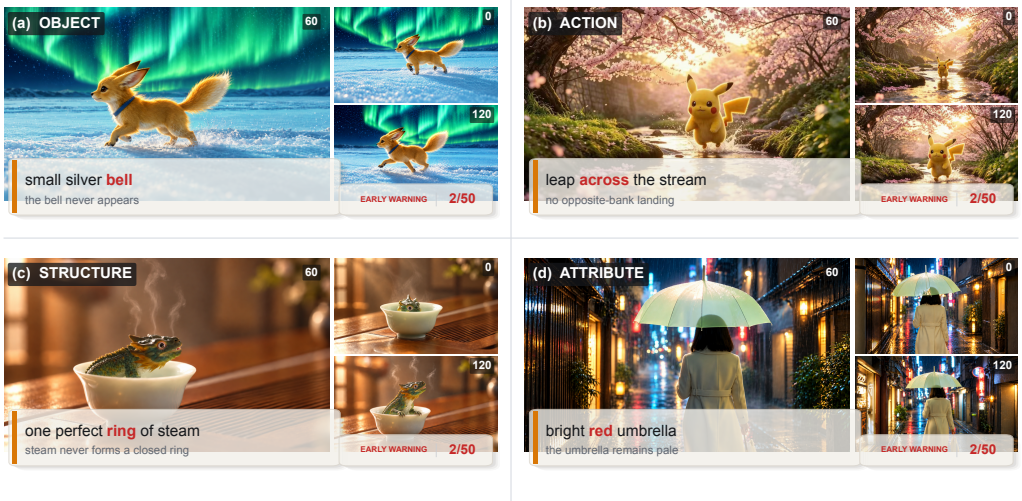

Figure 1: **Localized prompt-following failures detected by VideoProbe.** Each panel shows three temporal slices from a generated video. Orange highlights the human-verified prompt token or phrase whose visual constraint is not satisfied, while blue indicates the denoising steps at which the fixed cross-step probe detects the failure from cross-attention. Complete prompts, original Wan2.2 outputs, and per-step predictions are provided in Appendix H.1.

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

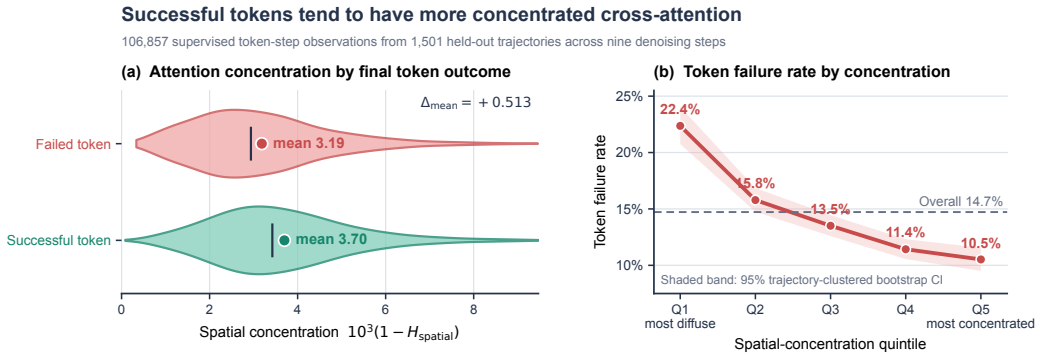

Figure 2: Intermediate attention geometry is associated with final visual-constraint outcomes. Spatial concentration is $C_{\text{spatial}} = 10^3(1 - H_{\text{spatial}})$, where H_{spatial} is the frame-wise normalized spatial entropy averaged over video time. Left: attention associated with satisfied constraints is more concentrated on average than attention associated with failed constraints (mean difference $+0.513$; 95% video-clustered bootstrap confidence interval $[0.434, 0.594]$). Right: the failure rate decreases from 22.4% in the most diffuse quintile to 10.5% in the most concentrated quintile. Statistics include 106,857 supervised token–step observations from 1,501 held-out videos and nine denoising steps.

1 Introduction

Diffusion-based text-to-video (T2V) models now generate visually convincing videos with coherent motion, supporting applications in content creation, simulation, and interactive systems (Yang et al., 2025; Wan et al., 2025). But visual quality does not ensure prompt faithfulness. A generated video may omit a requested object, assign an attribute to the wrong entity, violate a count or spatial relation, or fail to realize a requested action, interaction, or motion pattern (Huang et al., 2024; Liu et al., 2024; Sun et al., 2025).

Video evaluators such as VBench and EvalCrafter assess visual quality, motion, and prompt alignment after generation is complete (Huang et al., 2024; Liu et al., 2024). Each failed attempt therefore consumes a full denoising and decoding pass. In short-form video production, for example, a creator may generate the same shot many times and revise the prompt until the subjects, actions, and camera motion match the intended scene. Detecting likely failures during denoising could reduce this trial-and-error cost by stopping high-risk trajectories, switching to another seed, or alerting the creator to revise the relevant part of the prompt. Early diagnosis could thus increase the yield of usable videos under a fixed compute budget.

Probing methods offer a way to read early predictive signals from intermediate representations. In language models, lightweight probes can recover linguistic structure and behaviorally relevant information from hidden representations (Hewitt & Manning, 2019; McKenzie et al., 2026). In text-to-image diffusion models, cross-attention has been used to study word–region correspondence and control generation (Tang et al., 2023; Chefer et al., 2023). Closer to our setting, Diffusion Probe uses early cross-attention to predict scalar properties of the final image (Cui et al., 2026). However, it operates on two-dimensional spatial attention and produces image-level scores. Video failures often unfold over time: an action may not occur, a requested interaction may fail, or a state change may be missing. The attention representation also changes over the course of denoising. A video probe must therefore preserve a separate spatiotemporal attention structure for each prompt token and condition its predictions on the denoising timestep.

We find that tokens associated with satisfied and failed visual constraints have different intermediate cross-attention distributions. As shown in Figure 2, attention associated with satisfied constraints tends to be more spatially concentrated. Across concentration quintiles, the token failure rate falls from 22.4% to 10.5%. Satisfied constraints have higher mean spatial concentration than failed constraints at all nine denoising steps.

This insight motivates VideoProbe, a lightweight spatiotemporal probe that predicts token-level failures from intermediate video-to-text cross-attention. A 3D CNN encodes the $T \times H \times W$ atten-

tion volume of each prompt token, while token mixing shares the encoded attention features across tokens. The denoising timestep conditions the final prediction.

To train and evaluate the predictor, we construct ATTNPULSE-10K, which aligns multi-step cross-attention from 10,000 Wan2.2 trajectories with uncertainty-masked token supervision from completed videos. On the fixed 1,501-trajectory test set, VideoProbe reaches $.3398 \pm .0093$ AUPRC over nine denoising steps and three seeds. It also generalizes without retraining to unseen steps 0, 6, and 9, including the first denoiser forward. Same-step attention shuffling reduces AUPRC to $.1532 \pm .0062$. Separately trained probes on Wan2.1 and HunyuanVideo-1.5 recover the same early signal under explicit cross-attention and joint attention.

We make two contributions:

- We observe distributional differences in intermediate video-to-text cross-attention for tokens associated with satisfied and failed visual constraints. This reveals an early predictive property of video diffusion: before generation is complete, cross-attention carries information about whether individual prompt constraints will be satisfied in the final video. We design VIDEOPROBE and confirm this early predictability through token-level failure prediction on held-out trajectories.

- We introduce ATTNPULSE-10K, a benchmark of 10,000 Wan2.2 generation trajectories that aligns cross-attention from multiple denoising steps with uncertainty-masked token-level supervision derived from final videos. It combines atomic and compositional prompts with cleaned VATEX captions, covering static and temporal visual constraints. The benchmark provides the research community with a shared resource for studying early signals in video diffusion.

2 RELATED WORK

Text-to-Video Diffusion Models. Text-to-video generation has advanced from early video diffusion models to large diffusion transformers capable of synthesizing high-resolution videos with complex motion (Ho et al., 2022; Yang et al., 2025; Kong et al., 2024; Wan et al., 2025). These models must represent both visual appearance and temporal coherence, yet their outputs can still violate prompt constraints involving objects, spatial relations, actions, interactions, and state changes. While most work seeks to improve generation, we ask whether intermediate attention can forecast such failures before a trajectory is complete.

Post-hoc Video Evaluation. Video-generation benchmarks such as VBench and EvalCrafter assess visual quality, temporal consistency, motion, and text–video alignment on completed videos (Huang et al., 2024; Liu et al., 2024). Learned evaluators such as VideoScore approximate human judgments for output ranking and post-training (He et al., 2024). These methods require a fully generated and decoded video. VideoProbe instead predicts the outcome of prompt-specific visual constraints from an unfinished denoising trajectory.

Probing and Early Prediction in Diffusion Models. Lightweight probes have been used to read information from intermediate neural representations (Alain & Bengio, 2016; Hewitt & Manning, 2019). In text-to-image diffusion models, cross-attention supports word–region attribution and generation control (Tang et al., 2023; Hertz et al., 2022; Chefer et al., 2023). These methods mainly interpret or modify attention rather than predict the outcome of an unchanged trajectory.

Diffusion Probe is the closest image-domain work: it uses early cross-attention to predict scalar properties of the final image (Cui et al., 2026). Its input is two-dimensional spatial attention and its output is an image-level score. VideoProbe instead models token-specific spatiotemporal attention and predicts which visual constraints will fail. We also study whether this signal persists across denoising steps and video generators with different attention mechanisms.

3 METHOD

3.1 CROSS-ATTENTION GEOMETRY AND CONSTRAINT OUTCOMES

At each denoising step, video-to-text attention associates every prompt token with a response over the latent video grid. We interpret this response as the spatiotemporal support assigned to the corresponding visual constraint in the evolving video representation. A constraint that is successfully realized may develop a localized response around the relevant visual content, whereas an unrealized constraint may retain a more diffuse pattern. This motivates comparing the geometry of token-level attention associated with satisfied and failed constraints.

Let

$$O_{i,l,s} \in \mathbb{R}_{+}^{T_s \times H_s \times W_s}$$

denote the valid-token-normalized attention volume for token l in trajectory i at denoising step s. Let $Y_{i,l} = 0$ denote a satisfied visual constraint, $Y_{i,l} = 1$ a failed constraint, and $M_{i,l} = 1$ a token with reliable supervision. At each fixed step s, the two outcome-conditioned attention distributions are

$$\mu_s^{\mathrm{sat}}(B) = \Pr(O_{i,l,s} \in B \mid Y_{i,l} = 0, M_{i,l} = 1)\,, \tag{1}$$

$$\mu_s^{\mathrm{fail}}(B) = \Pr(O_{i,l,s} \in B \mid Y_{i,l} = 1, M_{i,l} = 1)\,, \tag{2}$$

for measurable sets B. These distributions describe the geometry of intermediate attention conditioned on the final constraint outcome.

We examine one interpretable projection of this geometry. For frame t, we normalize a token's response over the spatial grid:

$$p_{i,l,s,t}(h,w) = \frac{O_{i,l,s,t,h,w}}{\sum_{h',w'} O_{i,l,s,t,h',w'}}. \tag{3}$$

Let $u_s(h,w) = 1/(H_s W_s)$ be the uniform spatial distribution. We define spatial concentration as the normalized departure from this uniform allocation:

$$\begin{aligned} C(O_{i,l,s}) &= \frac{10^3}{T_s \log(H_s W_s)} \sum_{t=1}^{T_s} D_{\mathrm{KL}}(p_{i,l,s,t} \,\|\, u_s) \\ &= 10^3 \left[1 - \frac{1}{T_s \log(H_s W_s)} \sum_{t=1}^{T_s} H(p_{i,l,s,t}) \right], \end{aligned} \tag{4}$$

where $H(\cdot)$ is Shannon entropy. The score is zero for spatially uniform attention and increases as the response becomes more localized. The factor 10^3 only improves numerical readability.

We summarize the outcome-dependent displacement using

$$\Delta_s^C = \mathbb{E}[C(O_{i,l,s}) \mid Y_{i,l} = 0, M_{i,l} = 1] - \mathbb{E}[C(O_{i,l,s}) \mid Y_{i,l} = 1, M_{i,l} = 1]\,. \tag{5}$$

If $\mu_s^{\mathrm{sat}} = \mu_s^{\mathrm{fail}}$, the conditional expectation of every integrable statistic, including C, must be equal. Hence, $\Delta_s^C \neq 0$ establishes a difference under this projection. We observe $\Delta_s^C > 0$ across the evaluated steps: attention associated with satisfied constraints is more spatially concentrated on average (Figure 2). Because C discards response location, temporal evolution, and inter-token structure, VideoProbe uses the complete attention tensor rather than this scalar alone.

3.2 VIDEOPROBE

Figure 3 summarizes the complete training and inference pipeline. Given intermediate attention at step s, VideoProbe predicts

$$\widehat{y}_{i,l,s} = \sigma\Big(f_\theta \left(\widetilde{\mathbf{A}}_{i,s}, \mathbf{a}_i, \mathbf{e}_s \right)_l \Big)\,, \tag{6}$$

where $\widetilde{\mathbf{A}}_{i,s}$ is the token-aligned attention representation, $\mathbf{a}_i$ is the valid-token mask, and $\mathbf{e}_s$ is the denoising-timestep embedding. The generator is frozen throughout training.

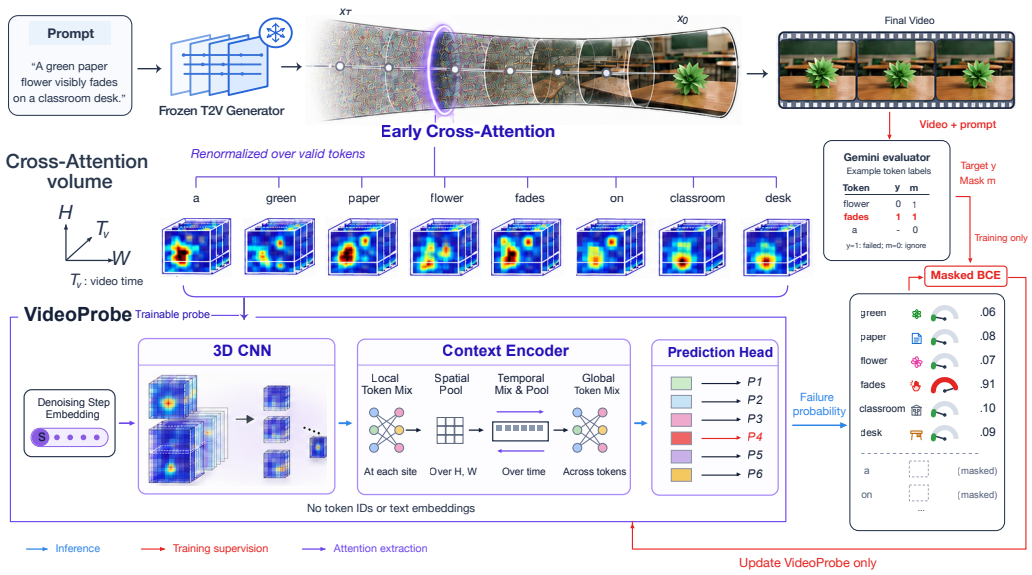

Figure 3: **Overview of VideoProbe.** VideoProbe extracts token-aligned attention from an intermediate denoising step, renormalizes it over valid prompt tokens, and uses a timestep-conditioned spatiotemporal encoder to predict a failure score for each token. Completed-video annotations supervise the probe during training only; the video generator remains frozen.

Attention representation. For the Wan generators, we collect video-to-text attention during the conditional forward pass, average the selected heads and Transformer blocks, and reshape the video-query axis into its latent spatiotemporal grid. This gives

$$\mathbf{A}_{i,s} \in \mathbb{R}^{L_{\max} \times T_s \times H_s \times W_s},$$

with one attention volume for each text channel. Attention capture is read-only and does not modify the denoising trajectory.

The text axis may contain padding channels. Let $a_{i,l} \in \{0, 1\}$ indicate whether channel l corresponds to a valid text token, and let $u = (t, h, w)$ index a latent video location. We condition the captured attention on valid prompt tokens:

$$\widetilde{A}_{i,s,l,u} = \frac{a_{i,l} A_{i,s,l,u}}{\sum_{j=1}^{L_{\max}} a_{i,j} A_{i,s,j,u} + \varepsilon}. \tag{7}$$

The mask $\mathbf{a}_i$ is determined by tokenization and contains no information about the final constraint outcomes.

HunyuanVideo-1.5 uses joint video–text attention rather than a separate cross-attention layer. We reuse its projected and normalized video queries and primary-text keys, recompute their dot-product logits, mask padded text tokens, and apply softmax over valid primary-text keys only. Thus, the Hunyuan representation uses the model's native attention projections but is not obtained by cropping the original joint-attention probability matrix. Video keys and auxiliary conditioning keys are excluded from its normalization denominator. Further extraction details are provided in Appendix A.2.

Spatiotemporal readout. A shared residual 3D CNN processes the $T_s \times H_s \times W_s$ attention volume of each valid token. Features from multiple CNN levels are projected to a common grid and fused with three-dimensional coordinate embeddings. The context encoder then exchanges information across prompt tokens at each video location, aggregates the spatial grid, models the resulting temporal sequence, and performs a final token-context update. A linear prediction head outputs one failure logit $z_{i,l,s}$ for each valid token.

One probe is shared across denoising steps. Following the generator's timestep-conditioning design, the scheduler timestep is encoded with a sinusoidal basis and an MLP. The resulting embedding

$\mathbf{e}_s$ modulates the residual CNN blocks through feature-wise scale and shift (Perez et al., 2018) and is added to the fused features. VideoProbe receives no explicit token identities, token strings, text embeddings, constraint categories, or learned token-position embeddings. Exact layer dimensions and optimization settings are given in Appendix A.3.

3.3 Multi-Step Learning and Inference

Token supervision. We derive token-level supervision from the completed video. A fixed video-language evaluator compares each generated video with its annotated visual constraints and assigns every constraint one of three outcomes: satisfied, failed, or uncertain. Failed constraints receive $y_{i,l} = 1$, satisfied constraints receive $y_{i,l} = 0$, and both receive supervision mask $m_{i,l} = 1$. Uncertain constraints, non-visual words, special tokens, and padding tokens have $m_{i,l} = 0$ and do not contribute to the loss.

Annotated character spans are aligned with the native tokenizer of each generator. When a constraint overlaps multiple subword tokens, each overlapping token inherits its outcome and supervision mask. Dependent constraints are handled conservatively: when an event required to judge another constraint is absent, the dependent constraint is marked uncertain rather than counted as an additional failure. The complete annotation prompt, output schema, dependency rules, validation procedure, and tokenizer-alignment protocol are reported in Appendix B.

Multi-step training. Let $\mathcal{S}_{\text{train}}$ denote the set of denoising steps used for training. During each epoch, we sample one available $s \in \mathcal{S}_{\text{train}}$ for every trajectory. The attention tensor and timestep therefore vary across epochs, while the completed-video labels remain fixed. This trains one probe across steps rather than fitting an independent predictor at each stage.

Failure tokens are less frequent than satisfied tokens. We compute the positive-class weight

$$w_+ = \min\left(\frac{N_-}{N_+},\, 4\right)$$

from supervised training tokens only. For trajectory i, let $M_i = \sum_l m_{i,l}$ be the number of supervised tokens. We minimize the sample-normalized, class-weighted binary cross-entropy

$$\ell_{i,s} = -\frac{1}{M_i} \sum_l m_{i,l} \left[w_+ y_{i,l} \log \sigma(z_{i,l,s}) + (1 - y_{i,l}) \log(1 - \sigma(z_{i,l,s})) \right]. \tag{8}$$

The final objective averages $\ell_{i,s}$ over trajectories with $M_i > 0$. Normalizing within each trajectory prevents prompts with more supervised tokens from dominating training.

At inference, VideoProbe receives only intermediate attention, the valid-token mask, and the current denoising timestep. It returns a failure probability for every valid prompt token. The completed video and its supervision labels are unavailable, and the video generator remains frozen.

4 Experiments

4.1 Experimental Setup

Generators and attention. We study Wan2.2-TI2V-5B, Wan2.1-T2V-14B (Wan et al., 2025), and HunyuanVideo-1.5 (Wu et al., 2025), each run for 50 denoising steps. The Wan models provide explicit video-to-text cross-attention. For HunyuanVideo-1.5, we extract the video-query to primary-text-key portion of its joint attention. We train a separate VideoProbe for each generator.

Data and supervision. Our primary benchmark, AttnPulse-10K, contains 10,000 Wan2.2 generation trajectories from 1,000 compositional prompts, 3,000 controlled atomic prompts, and 6,000 cleaned VATEX captions (Wang et al., 2019). We also use 3,704 Wan2.1 and 4,529 HunyuanVideo-1.5 trajectories. Their fixed held-out test sets contain 1,501, 556, and 631 trajectories, respectively. A video-language evaluator labels each annotated visual constraint in the completed video as satisfied, failed, or uncertain. We align these labels to each generator's native text tokens and exclude uncertain, non-visual, special, and padding tokens. All attention steps from one trajectory remain in the same data split.

Table 1: AUROC (↑) for early token-level failure prediction. Values are means over three seeds on each generator's held-out test set. Step is the zero-based index in a 50-step run. "Dynamics" is the union of action, motion, and interaction tokens; tokens in multiple categories are counted once. Attr. includes color, and Change denotes state change.

Model	Step	AUROC ↑							
		All	Obj.	Attr.	Num.	Spat.	Dyn.	Chg.	Scene
Wan2.2	2	0.768	0.645	0.573	0.728	0.559	0.517	0.696	0.606
	5	0.766	0.649	0.561	0.729	0.547	0.532	0.661	0.606
	8	0.766	0.648	0.581	0.727	0.540	0.538	0.625	0.610
	12	0.768	0.649	0.567	0.736	0.529	0.547	0.665	0.607
Wan2.1	2	0.694	0.653	0.699	0.619	0.480	0.539	0.658	0.649
	5	0.688	0.640	0.660	0.650	0.537	0.525	0.649	0.632
	8	0.688	0.638	0.658	0.668	0.550	0.527	0.593	0.621
	12	0.687	0.640	0.669	0.680	0.550	0.528	0.588	0.618
Hunyuan	2	0.765	0.741	0.526	0.725	0.562	0.621	0.451	0.636
	5	0.777	0.761	0.649	0.766	0.590	0.582	0.564	0.766
	8	0.766	0.732	0.675	0.770	0.573	0.595	0.517	0.689
	12	0.750	0.711	0.676	0.764	0.577	0.573	0.570	0.680

Training and evaluation. For each generator, we train one probe using attention from steps $\{2, 5, 8, 12, 15, 24, 36, 45, 49\}$. Before probing, attention is renormalized over valid text tokens. We select checkpoints by validation average precision and report results from three training seeds. The main table evaluates each generator's fixed held-out trajectories at zero-based steps 2, 5, 8, and 12. For the Wan2.2 cross-step analysis, the same checkpoints are also evaluated without retraining at steps $\{0, 6, 9\}$, none of which appears in the training-step set.

Metrics. Failed constraints are the positive class. We report token-level AUROC and AUPRC, together with failure prevalence because it determines the random AUPRC baseline. In Table 1, "All" pools all supervised tokens within each generator, while category scores are computed on their corresponding token subsets.

4.2 Early Prediction Across Generators and Constraint Types

Table 1 reports token-level AUROC at four early denoising steps. At step 2, the overall AUROC reaches 0.768 on Wan2.2, 0.694 on Wan2.1, and 0.765 on HunyuanVideo-1.5. Scores change little across these steps for the two Wan models, while HunyuanVideo-1.5 reaches 0.777 at step 5, its highest value in this range. We find that attention early in denoising can predict final token-level failures.

This pattern appears under different attention mechanisms. The Wan models provide explicit video-to-text cross-attention, whereas HunyuanVideo-1.5 uses the video-to-text portion of joint attention. Although the probe weights are trained separately for each generator, the same probing formulation extracts early predictive information from both types of representation.

The signal also spans several visual constraint types. At step 2, numeracy reaches an AUROC of 0.728 on Wan2.2, while object existence reaches 0.741 on HunyuanVideo-1.5. At the same step, scene-context AUROC exceeds 0.60 on all three generators. Dynamic constraints are more difficult to distinguish, with step-2 AUROC values of 0.517 on Wan2.2 and 0.621 on HunyuanVideo-1.5. VideoProbe therefore supports early diagnosis across multiple visual constraints, while prediction of dynamic failures remains more limited.

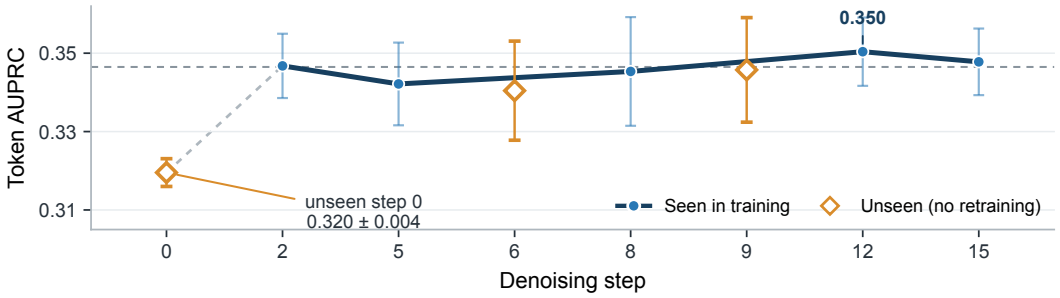

Figure 4: Generalization to unseen denoising steps on the fixed 1,501-trajectory Wan2.2 test set. Points and error bars show mean ± sample standard deviation over three training seeds. Blue circles denote training steps $\{2, 5, 8, 12, 15\}$ shown in the early regime; orange diamonds denote unseen steps $\{0, 6, 9\}$ evaluated without retraining. The dashed line shows the mean across the five displayed training steps. Checkpoints are equally spaced for readability, with tick labels indicating their actual denoising-step indices.

4.3 Generalization to Unseen Denoising Steps

We train one Wan2.2 probe jointly on nine denoising steps. To test whether the learned readout generalizes beyond the training-step set, we evaluate the same checkpoints on the same 1,501 held-out trajectories at unseen steps $\{0, 6, 9\}$, without retraining.

Figure 4 focuses on the early denoising regime. At unseen step 6, VideoProbe reaches $.3404 \pm .0126$ AUPRC, while at unseen step 9 it reaches $.3457 \pm .0133$; both results fall within the performance range of neighboring training steps. At unseen step 0, corresponding to the first denoiser forward, the probe retains $.3195 \pm .0035$ AUPRC and $.7556 \pm .0050$ AUROC. These results show that the learned readout is not restricted to the denoising stages presented during training. Complete results across all evaluated steps and metrics are provided in Appendix Table 5.

4.4 Dependence on Sample-Matched Attention

We test whether prediction depends on attention from the evaluated trajectory. At each step, we permute attention among held-out trajectories with no fixed points, while keeping the trained probe, token labels, and supervision masks fixed. This preserves the collection of attention tensors at each step while breaking their correspondence with the final video outcomes.

Table 2 reports results averaged over steps $\{2, 5, 8, 12\}$. Shuffling reduces AUROC from .767 to .524 on Wan2.2, from .683 to .534 on Wan2.1, and from .765 to .540 on HunyuanVideo-1.5. AUPRC also falls from .346 to .157, from .267 to .163, and from .277 to .139, respectively. The classification and prompt-diagnosis metrics follow the same pattern. Because the labels and denoising steps remain fixed, these gaps show that prediction depends on attention matched to the evaluated trajectory, rather than on failure prevalence or timestep information alone.

4.5 Potential Downstream Applications

VideoProbe produces token-aligned constraint risks before a trajectory is fully generated. This output provides a common interface for generation and post-training workflows that require early decisions. Figure 5 illustrates three potential applications.

First, *seed routing* can begin several trajectories for the same prompt, use VideoProbe to assess user-prioritized constraints, and allocate the remaining denoising budget to lower-risk candidates. High-risk candidates can instead be stopped or resampled. Second, *prompt alerts* can identify constraints that receive consistently high risk across multiple seeds. Such agreement may indicate that the prompt or generation setting should be reviewed before launching additional trajectories. Third, *RL rollout mining* can use early risk estimates to route trajectories for full evaluation. Low-risk rollouts may provide promising candidates, while high-risk rollouts can serve as informative failure cases for preference or reinforcement learning.

Table 2: Effect of sample-matched attention on early failure prediction. Results are averaged over steps $\{2, 5, 8, 12\}$ within each seed and then over three seeds on each generator's held-out test set. M denotes matched attention; S pairs each trajectory's labels with attention from another trajectory at the same step. Parentheses show AP lift.

Generator	Attn.	Token ranking		Token classification		Prompt diagnosis	
		AUPRC (AP lift)↑	AUROC↑	F_1^+ ↑	BAcc.↑	IoU↑	Top-3↑
Wan2.2	M	.346 (2.35×)	.767	.423	.701	.295	.858
	S	.157 (1.07×)	.524	.248	.518	.152	.637
Wan2.1	M	.267 (1.79×)	.683	.342	.631	.234	.801
	S	.163 (1.09×)	.534	.256	.525	.152	.660
Hunyuan-1.5	M	.277 (2.33×)	.765	.371	.692	.288	.902
	S	.139 (1.17×)	.540	.204	.526	.146	.647

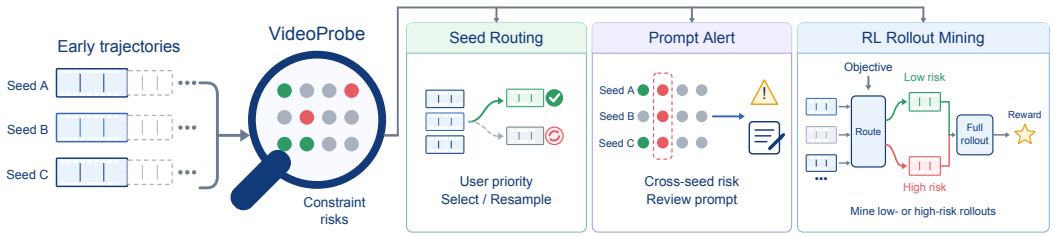

Figure 5: **Potential downstream applications of VideoProbe.** Given partially denoised trajectories from multiple seeds, VideoProbe estimates token-level visual-constraint risks. These estimates can support seed routing, cross-seed prompt alerts, and rollout mining for post-training. The figure illustrates possible uses of the predictive signal rather than evaluated downstream improvements.

These applications require no change to the frozen video generator: VideoProbe acts on intermediate attention and can be removed from the generation path after making the routing decision. We present them as potential uses of the early diagnostic signal and leave their quantitative evaluation to future work.

5 Conclusion

We find that intermediate video-to-text attention contains information about which prompt-specific visual constraints will fail before generation is complete. Tokens associated with satisfied and failed constraints exhibit different attention distributions. On Wan2.2, a probe trained jointly across nine denoising steps distinguishes these outcomes early and generalizes without retraining to unseen steps 0, 6, and 9, including the first denoiser forward. Separately trained probes recover this signal from both explicit cross-attention and the video-to-text portion of joint attention across three video generators.

We use VideoProbe to read out this signal at the token level. Its performance falls substantially when attention is shuffled among trajectories at the same denoising step, linking prediction to sample-matched internal representations. We also introduce AttnPulse-10K, which aligns multi-step attention from 10,000 Wan2.2 trajectories with token-level supervision from the final videos. Prediction remains weaker for dynamic constraints, and the practical benefits of using early risk estimates to guide generation decisions remain to be measured. Overall, our results identify intermediate attention as an early predictive representation of final video-generation outcomes.

Reproducibility Statement

Appendices A–C document the datasets, fixed splits, attention extraction, probe architecture, optimization settings, supervision protocol, and metric definitions. Appendices D–G report complete cross-step results, constraint-level analyses, attention statistics, and ablations. The complete evaluator instruction is provided in Appendix I, and our code and data release plan is described in Appendix J. Unless stated otherwise, the main results use fixed held-out test sets and three training seeds.

AI Use Statement

Google Gemini 3.1 Pro Preview was used as the video-language evaluator for constructing token-level supervision from completed videos, following the fixed constraint schema, validation rules, dependency rules, and uncertainty masking described in the paper. OpenAI ChatGPT and Codex assisted with research brainstorming, conceptual framing and mathematical formulation, experimental design, method implementation and debugging, literature search and reference formatting, result analysis and interpretation, LaTeX preparation, and drafting and revising parts of the manuscript, including the appendix. GPT Image was used to generate preliminary sketches and layout concepts for several figures; the final figures were revised and assembled by the authors.

The authors selected the final methodology, ran the reported experiments, verified the code and numerical results, reviewed and revised all AI-assisted materials, and take full responsibility for the content of this work.

References

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

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

Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM transactions on Graphics (TOG)*, 42(4):1–10, 2023.

Hyung Won Chung, Noah Constant, Xavier Garcia, Adam Roberts, Yi Tay, Sharan Narang, and Orhan Firat. Unimax: Fairer and more effective language sampling for large-scale multilingual pretraining. *arXiv preprint arXiv:2304.09151*, 2023.

Benlei Cui, Bukun Huang, Zhizeng Ye, Xuemei Dong, Tuo Chen, Hui Xue, Dingkang Yang, Longtao Huang, Jingqun Tang, and Haiwen Hong. Diffusion probe: Generated image result prediction using cnn probes. *arXiv preprint arXiv:2602.23783*, 2026.

Xuan He, Dongfu Jiang, Ge Zhang, Max Ku, Achint Soni, Sherman Siu, Haonan Chen, Abhranil Chandra, Ziyan Jiang, Aaran Arulraj, et al. Videoscore: Building automatic metrics to simulate fine-grained human feedback for video generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 2105–2123, 2024.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4129–4138, Minneapolis,

Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1419. URL https://aclanthology.org/N19-1419/.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Tim Salimans, Alexey A Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *ICLR workshop on deep generative models for highly structured data*, 2022.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21807–21818, 2024.

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

Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Zeng, Raymond Chan, and Ying Shan. 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.

Alex McKenzie, Urja Pawar, Phil Blandfort, William Bankes, David Krueger, Ekdeep Singh Lubana, and Dmitrii Krasheninnikov. Detecting high-stakes interactions with activation probes, 2026. URL https://arxiv.org/abs/2506.10805.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Kaiyue Sun, Kaiyi Huang, Xian Liu, Yue Wu, Zihan Xu, Zhenguo Li, and Xihui Liu. T2v-compbench: A comprehensive benchmark for compositional text-to-video generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8406–8416. IEEE, 2025.

Raphael Tang, Linqing Liu, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar, Pontus Stenetorp, Jimmy Lin, and Ferhan Türe. What the daam: Interpreting stable diffusion using cross attention. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5644–5659, 2023.

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

Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4580–4590. IEEE, 2019.

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

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, volume 2025, pp. 83048–83077, 2025.

Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. Unipc: A unified predictor-corrector framework for fast sampling of diffusion models. *Advances in Neural Information Processing Systems*, 36:49842–49869, 2023.

Table 3: Generator-specific datasets and evaluation protocols. Splits are formed at the trajectory level and are disjoint in normalized prompt groups. All denoising steps belonging to a trajectory remain in the same partition.

Generator	Total	Train	Val.	Test	Attention layout	Text tokenizer
Wan2.2-TI2V-5B	10,000	6,999	1,500	1,501	video-to-text cross-attn.	UMT5 (Chung et al., 2023)
Wan2.1-T2V-14B	3,704	2,593	555	556	video-to-text cross-attn.	UMT5
HunyuanVideo-1.5	4,529	3,195	703	631	video-query/text-key joint-attn. block	Qwen2.5-VL (Bai et al., 2025)

A Additional Experimental Details

We provide full details of the data, attention extraction, training protocol, metrics, and auxiliary experiments. Unless noted otherwise, failed visual constraints form the positive class, step indices are zero-based, and results are the mean $\pm$ sample standard deviation over training seeds 42, 123, and 456.

A.1 Datasets and Splits

We use three generator-specific datasets (Table 3). AttnPulse-10K contains 1,000 compositional target-span prompts, 3,000 controlled atomic prompts, and 6,000 cleaned VATEX captions. We generated and annotated the Wan2.1 and HunyuanVideo-1.5 datasets independently, using the same constraint schema. Each generator has its own trained VideoProbe, so the cross-generator comparison evaluates the probing formulation rather than zero-shot transfer of probe weights.

We form prompt groups after lowercasing, collapsing whitespace, and removing non-alphanumeric separators. A single split is shared by all attention steps and probe seeds. No test trajectory or normalized prompt group enters checkpoint selection. At every reported denoising step, the fixed test panels contain 1,501 Wan2.2, 556 Wan2.1, and 631 HunyuanVideo-1.5 trajectories.

A.2 Video Generation and Attention Capture

Wan2.2 videos contain 121 frames at 1280×704 resolution. Generation uses 50 denoising steps, flow shift 5.0, classifier-free guidance (Ho & Salimans, 2022) scale 5.0, and the UniPC solver (Zhao et al., 2023). Wan2.1 videos also contain 121 frames and use 50 steps, at a resolution of 1280×720. The release retains generator revisions, random seeds, and complete inference manifests.

For the Wan models, we capture conditional video-to-text cross-attention. We average the heads in the selected final ten Transformer blocks and reshape the video-query dimension to the latent spatiotemporal grid. The resulting Wan2.2 tensor has shape $[512, 31, 22, 40]$, ordered as text channel, video time, height, and width. HunyuanVideo-1.5 uses joint attention, so we reuse its native projected queries and keys and compute $\mathrm{softmax}(Q_{\mathrm{video}} K_{\mathrm{primary\ text}}^{\top})$ over valid primary-text keys only. Video keys, auxiliary conditioning tokens, and padded primary-text keys are excluded from this denominator. The resulting representation uses the joint-attention projections, but it is not the model's original probability distribution over all joint keys. For the Wan models, Eq. 7 conditions the captured cross-attention on tokenizer-valid text channels. Extraction is read-only and does not alter the denoising trajectory.

The primary early-step comparison uses ordinal steps $\{2, 5, 8, 12\}$. For the detailed Wan2.2 cross-step analysis, the probe is trained on steps $\{2, 5, 8, 12, 15, 24, 36, 45, 49\}$ and is additionally evaluated, without retraining, on unseen steps $\{0, 6, 9\}$. Every step uses the same held-out videos.

A.3 Probe Architecture and Optimization

The standard training configuration appears in Table 4. During each epoch, we draw one available step at random from every training trajectory instead of treating its steps as independent samples. We select the checkpoint by validation token AUPRC, then choose one decision threshold on the validation predictions to maximize positive-class F1. This threshold is fixed for test evaluation.

Table 4: Default VideoProbe training configuration.

Component	Setting
Backbone	hierarchical residual 3D CNN
Base channels / hidden dim.	8 / 32
Fused feature grid	$8 \times 4 \times 6$
Local / temporal / global layers	1 / 1 / 1
Attention heads	4
Dropout	0.4
Optimizer	AdamW
Learning rate	7.5×10^{-5}
Minimum learning rate	10^{-6}
Weight decay	3×10^{-3}
Epochs	35
Warmup / cosine decay	1 / 25 epochs
Batch size / accumulation	1 / 8
Gradient clipping	1.0
Precision	FP16 automatic mixed precision
Positive-class weight	$\min(N_-/N_+, 4)$
Checkpoint criterion	validation token AUPRC
Threshold criterion	validation positive-class F1
Training seeds	42, 123, 456

The denoising step is represented by its scheduler timestep τ_s. We construct a sinusoidal embedding

$$\mathbf{e}_s = \mathrm{MLP}\left([\cos(\tau_s\omega_k), \sin(\tau_s\omega_k)]_{k=0}^{127}\right), \qquad \omega_k = 10000^{-k/128}. \tag{9}$$

The embedding modulates each residual 3D CNN block through feature-wise scale and shift and is also added to the fused spatiotemporal features.

The model takes valid-token-renormalized attention and a sinusoidal embedding of the generator's denoising timestep. It does not receive token IDs, token strings, or text embeddings. The generator remains frozen.

B Token-Supervision Protocol

B.1 Constraint Construction

We decompose each prompt into atomic constraints that can be judged visually. The inventory covers object existence, attribute binding, numeracy, spatial relations, actions, motion, interactions, dynamic attributes or state changes, and scene context. Each constraint records its character span, type, minimal candidate words for supervision, and any prerequisites. We fix these candidate spans before evaluating the final video.

Dependency rules prevent a single defect from producing several failure labels. If a required object is absent, its existence constraint may fail, while any attribute, action, or relation that depends on the object is marked uncertain. We also separate event execution from motion direction. When an action occurs in the wrong direction, the action verb is satisfied and only the direction span fails. When the event does not occur, the action fails and the direction is uncertain.

B.2 Full-Video Gemini Evaluation

We provide `gemini-3.1-pro-preview`, run at temperature 0, with the final MP4, exact prompt, indexed prompt words, and frozen constraint inventory. The evaluator inspects the complete video, not a thumbnail or sparse frame sample. It labels each constraint as *satisfied*, *failed*, or *uncertain* and records brief visual evidence. Failed and satisfied constraints yield positive and negative labels, respectively; uncertain constraints are excluded from training and evaluation.

Four attribution rules govern the labels. A positive label must fall inside the candidate span supplied for that constraint, and it must use the smallest candidate subset that identifies the visible failure. The evaluator cannot replace a missing candidate with a semantically related word. If a prerequisite is missing, dependent constraints become uncertain. These rules separate missing objects from incorrect attributes, absent actions, and wrong motion directions.

The schema-constrained JSON output contains a video description, one judgment per constraint, evidence and confidence, the applied failure-token indices, the complete binary label sequence, and the supervision mask. We retry provider failures and reject schema-invalid or internally inconsistent records.

Appendix I reproduces the full evaluator instruction. Each user payload adds only the video, exact prompt, indexed prompt tokens, and frozen constraint records; it gives no sample-specific guidance about the expected outcome.

B.3 Validation and Tokenizer Alignment

We validate every returned record deterministically. The checks cover the pinned model and temperature, full-video input mode, prompt identity, constraint order and text, candidate indices and words, label and mask lengths, dependency rules, and equality between the union of applied constraint indices and the positive-token indices. A record that fails any check is rejected and evaluated again.

We align validated character spans to each generator's native tokenizer by character offset. If a span overlaps several subword tokens, all overlapping tokens inherit its outcome and mask. Non-visual text, special tokens, padding, and uncertain constraints receive a zero supervision mask. Valid but unsupervised token channels remain available as context and do not contribute to the loss.

We retain the frozen constraint records, raw evaluator responses, validation outcomes, and tokenizer alignments to support reproducibility and subsequent auditing.

C Metric Definitions

Let $\mathcal{T}$ denote the supervised test tokens, $y_j \in \{0, 1\}$ the failure label, and p_j the predicted failure probability. We pool the supervised tokens in each evaluation subset before computing token AUROC and AUPRC, with AUPRC implemented as average precision. If $\pi = |\{j : y_j = 1\}|/|\mathcal{T}|$ is the failure prevalence, the random-ranking baseline for AUPRC is π, and we report

$$\text{AP lift} = \text{AUPRC}\,/\pi. \tag{10}$$

AP lift helps compare categories or generators with different prevalences; we report it alongside, rather than instead of, raw AUPRC.

For thresholded metrics, we select the threshold on the validation set and do not retune it on test data. Positive F1 is

$$F_1^+ = \frac{2\,\text{TP}}{2\,\text{TP} + \text{FP} + \text{FN}}, \tag{11}$$

and balanced accuracy is

$$\text{BAcc} = \frac{1}{2}\left(\frac{\text{TP}}{\text{TP} + \text{FN}} + \frac{\text{TN}}{\text{TN} + \text{FP}}\right). \tag{12}$$

For trajectory i, let P_i and Y_i be the predicted and true failed-token sets. Sample IoU is

$$\text{IoU}_i = \frac{|P_i \cap Y_i|}{|P_i \cup Y_i|}, \tag{13}$$

We define IoU as one when both sets are empty and average it over test trajectories. Top-3 hit rate is evaluated only on trajectories with at least one failed token. It equals one if a true failure appears among the three highest-scoring supervised tokens. "All" metrics pool supervised tokens; they are not averages of category scores. Each category metric uses only tokens carrying that constraint label.

For multi-step summaries, we first macro-average the selected steps for each training seed, then compute the mean and sample standard deviation across the three seeds. Thus, a step with more cached records receives no extra weight.

Table 5: Wan2.2 results across denoising steps on the paired 1,501-trajectory test panel. Steps 0, 6, and 9 are unseen during training.

Step	AUPRC	AUROC	F_1^+	BAcc.	IoU	Top-3
0	.3195 ± .0035	.7556 ± .0050	.4062 ± .0062	.6789 ± .0103	.2821 ± .0101	.8645 ± .0055
2	.3468 ± .0082	.7684 ± .0042	.4256 ± .0064	.7026 ± .0136	.2953 ± .0070	.8599 ± .0022
5	.3422 ± .0106	.7661 ± .0028	.4212 ± .0025	.6989 ± .0116	.2935 ± .0050	.8627 ± .0012
6	.3404 ± .0126	.7638 ± .0027	.4207 ± .0043	.6983 ± .0108	.2959 ± .0062	.8610 ± .0024
8	.3453 ± .0139	.7663 ± .0021	.4225 ± .0039	.7003 ± .0104	.2958 ± .0079	.8589 ± .0030
9	.3457 ± .0133	.7669 ± .0008	.4237 ± .0051	.7000 ± .0100	.2975 ± .0055	.8561 ± .0096
12	.3504 ± .0087	.7677 ± .0017	.4238 ± .0019	.7005 ± .0075	.2956 ± .0043	.8494 ± .0067
15	.3478 ± .0085	.7676 ± .0021	.4245 ± .0032	.7026 ± .0076	.2966 ± .0086	.8498 ± .0055
24	.3389 ± .0095	.7631 ± .0030	.4219 ± .0035	.6993 ± .0107	.2944 ± .0069	.8452 ± .0044
36	.3336 ± .0113	.7581 ± .0033	.4191 ± .0031	.6974 ± .0079	.2927 ± .0082	.8365 ± .0061
45	.3291 ± .0073	.7612 ± .0055	.4262 ± .0056	.7053 ± .0112	.2904 ± .0058	.8540 ± .0048
49	.3244 ± .0103	.7600 ± .0069	.4256 ± .0048	.7045 ± .0086	.2901 ± .0055	.8540 ± .0056

Table 6: Wan2.2 performance by visual-constraint category on the paired 1,501-trajectory test set. Results are mean ± sample standard deviation over three seeds after macro-averaging nine trained steps. AP lift is AUPRC divided by category failure prevalence.

Constraint	Tokens/step	Failure rate	AUPRC	AP lift	AUROC	BAcc.
Action	1,132	.344	.3775 ± .0127	1.10×	.5391 ± .0111	.5216 ± .0042
Attribute	1,503	.042	.0911 ± .0104	2.17×	.5753 ± .0098	.5811 ± .0108
Dynamic attribute	201	.950	.9675 ± .0058	1.02×	.6440 ± .0327	.6265 ± .0299
Interaction	350	.506	.5109 ± .0072	1.01×	.5043 ± .0103	.5030 ± .0072
Motion	747	.407	.4325 ± .0121	1.06×	.5339 ± .0176	.5281 ± .0111
Numeracy	377	.403	.5935 ± .0170	1.47×	.7280 ± .0137	.6766 ± .0075
Object existence	5,713	.067	.1174 ± .0045	1.74×	.6494 ± .0039	.5791 ± .0120
Scene context	1,811	.010	.0163 ± .0005	1.64×	.6003 ± .0135	.5363 ± .0243
Spatial relation	477	.216	.2527 ± .0086	1.17×	.5221 ± .0122	.4934 ± .0125

D Complete Wan2.2 Cross-Step Results

Table 5 reports all metrics on the fixed Wan2.2 test panel. Steps 0, 6, and 9 are excluded from training; one probe per seed is trained jointly on the other nine steps and evaluated at all reported steps without retraining.

E Constraint-Type Analysis

Table 6 breaks down Wan2.2 performance by constraint type after macro-averaging the nine trained steps for each seed. Raw AUPRC depends on failure prevalence and should be read with AP lift. The dynamic-attribute AUPRC, for instance, partly reflects a 95% failure rate. Numeracy performs well on AUPRC, AP lift, and AUROC, while interaction and spatial relations are close to their respective baselines.

F Attention-Distribution Statistics

For the motivation analysis, we group token-step observations by final constraint outcome, regardless of VideoProbe's prediction. The analysis uses 106,857 supervised observations from 1,501 held-out Wan2.2 videos at nine denoising steps: 91,125 satisfied and 15,732 failed. We compute confidence intervals from 5,000 bootstrap resamples clustered by video, keeping correlated tokens and steps from a trajectory together.

All nine steps show the same directional differences, although no statistic is a strong classifier on its own. A linear model using nine hand-designed statistics reaches .6470 ± .0004 AUROC and .2199 ± .0003 AUPRC under grouped cross-validation, below VideoProbe. The probe therefore reads more than one concentration or smoothness statistic.

Table 7: Attention statistics for satisfied and failed Wan2.2 constraints. The difference and 95% confidence interval are clustered by video. Oriented AUROC chooses the direction in which a larger value predicts satisfaction.

Statistic	Satisfied mean	Failed mean	Satisfied − failed [95% CI]	Oriented AUROC
Spatial concentration $1000(1-H)$	3.70	3.19	$+.513$ $[+.434, +.594]$	.595
Relative spatial total variation	.1748	.1614	$+.0134$ $[+.0115, +.0152]$	.594
Peak / mean	3.461	3.193	$+.269$ $[+.217, +.319]$	.569
Top-5% mass fraction	.0829	.0808	$+.0021$ $[+.0016, +.0026]$	.567
Mean valid-token share	.0243	.0366	$-.0122$ $[-.0146, -.0099]$	.567
Adjacent-frame cosine	.9728	.9755	$-.0027$ $[-.0033, -.0022]$	.563
Relative temporal change	.1765	.1689	$+.0075$ $[+.0057, +.0094]$	.555

G Ablation Studies

Unless noted otherwise, these ablations use fixed-step, single-seed protocols. They motivate the design choices but do not replace the three-seed main evaluation.

G.1 Spatiotemporal Architecture

Table 8 compares three ways of aggregating the token-aligned attention volume. Delaying spatial and temporal compression improves both ranking and prompt-level diagnosis, while the hierarchical encoder performs best on every reported metric. This result motivates the hierarchical architecture used in the main experiments.

Table 8: Architecture ablation on the 10k Wan2.2 step-6 benchmark (seed 42).

Architecture	AUPRC	AUROC	F_1^+	BAcc.	IoU	Top-3
Early pooling	.2758	.7089	.3729	.6599	.2548	.8099
Later pooling	.2963	.7160	.3728	.6652	.2550	.8330
Hierarchical	**.3354**	**.7658**	**.4313**	**.7190**	**.2908**	**.8613**

G.2 Training-Set Scale and Capacity

Table 9 isolates the effect of training-set size while keeping the architecture and evaluation split fixed. Performance rises consistently from 500 to 6,999 training trajectories: AUPRC increases from .2388 to .3354 and AUROC from .6612 to .7658. The result indicates that the probe continues to benefit from additional labeled trajectories over the range evaluated here.

Table 9: Training-set scale at Wan2.2 step 6 with a fixed architecture, validation set, and test set (seed 42).

Train trajectories	AUPRC	AUROC	F_1^+	BAcc.
500	.2388	.6612	.3284	.6200
2,000	.2842	.7352	.4024	.6829
6,999	**.3354**	**.7658**	**.4313**	**.7190**

Increasing model size does not yield the same consistent improvement (Table 10). The small probe underfits, whereas the large probe slightly improves AUROC but reduces the primary AUPRC metric relative to the default configuration. We therefore use the default capacity for the main experiments.

Table 10: Probe-capacity ablation on the fixed 2,000/500/500 split at step 5 (seed 42).

Capacity	Base	Dim.	Heads	AUPRC	AUROC	F_1^+	BAcc.
Small	4	16	4	.2717	.7109	.3535	.6609
Default	8	32	4	**.3029**	.7535	.4254	**.7168**
Large	16	64	8	.2885	**.7589**	**.4302**	.7113

G.3 Timestep Conditioning and Attention Normalization

In the single-seed paired-1,501 comparison, Wan scheduler-timestep conditioning improves ranking AUPRC but not every thresholded metric. The valid-token normalization study uses an earlier 1,000-video pilot; all formal experiments use the selected valid-token variant.

G.4 Independent Step-Specific Probes

An earlier control trains one probe per step on a fixed 2,000/500/500 split. It also finds early predictability without monotonic gains late in denoising. Because the main protocol uses a shared probe, the values are not directly comparable.

Table 11: Timestep conditioning over nine trained Wan2.2 steps (seed 42).

Condition	AUPRC	AUROC	F_1^+	BAcc.
No timestep	.3385	**.7657**	**.4304**	**.7101**
WAN timestep	**.3503**	.7653	.4216	.7082

Table 12: Attention normalization in the earlier 1k Wan2.2 step-6 pilot (seed 42).

Normalization	AUPRC	AUROC	F_1^+
Raw	.2658	.6735	.3199
Padding-relative	.2715	.6873	.3361
Valid-token	**.3007**	**.7152**	**.3794**

G.5 Historical Language-Prior Baselines

Before adopting the attention-only protocol, we ran a historical, non-comparable step-6 pilot on 1,000 Wan2.2 trajectories. Because the text-only model reads token identity, it measures prompt-side failure priors rather than information from the generated trajectory. Its performance confirms that the prompt distribution contains such priors. The final VideoProbe therefore receives neither token IDs nor text embeddings, and the shuffled control tests the contribution of sample-matched attention. This pilot uses seed 42, raw attention, and a 700/150/150 split; its values should not be compared directly with the main results.

Table 13: Independent Wan2.2 step probes on a fixed 2,000/500/500 split (single seed).

Step	AUPRC	AUROC	F_1^+	BAcc.	IoU	Top-3
0	.3404	.7848	.4622	.7351	.3028	.8991
2	.3126	.7667	.4301	.7049	.2891	.8770
5	.3029	.7535	.4254	.7168	.2758	.8486
12	.2991	.7459	.4023	.6854	.2844	.8297
24	.2966	.7420	.3990	.6768	.2758	.8265
49	.3195	.7650	.4418	.7150	.3029	.8675

Table 14: Historical text and attention baselines on the 1k Wan2.2 step-6 pilot (seed 42). Attention inputs in this table are raw rather than valid-token-renormalized.

Input/model	AUPRC	AUROC	F_1^+	BAcc.
Text only	.4877	.8506	.5595	.7565
Text + attention	.3761	.7699	.4724	.7002
Text + shuffled attention	.3677	.7565	.4369	.6692
Attention only	.2658	.6735	.3199	.6081
Attention only, shuffled	.1792	.5882	.2901	.5766

H QUALITATIVE EXAMPLE DETAILS

The four main-paper cases are selected held-out examples chosen to illustrate distinct failure types. They do not enter the quantitative evaluation and are not intended to constitute a random or representative sample of the test set. Below, we provide the complete prompt, generation seed, final-video failure description, and localized failed token for each example. The display may merge subwords into words, but all reported aggregate metrics remain at the native tokenizer level.

H.1 SELECTED HELD-OUT EXAMPLES

We report the complete prompts and failure descriptions for four selected held-out examples. The judgments and probe predictions come from the original Wan2.2-5B videos and saved cross-attention tensors. We evaluate the same fixed cross-step probe at denoising steps 0, 2, 4, and 6; we do not train a separate probe for each step. In all four examples, the probe identifies the localized failure token at every tested step.

(a) Missing object. **Full prompt:** "Pikachu runs across untouched snow beneath a green aurora, wearing a narrow blue collar with a small silver bell clearly visible beneath its neck. Powdery snow rises behind it as a low camera tracks smoothly alongside under crisp moonlight." **Generation seed:** 777. **Human-verified failure:** The collar is visible, but the small silver bell is consistently absent. **Localized failed token:** *bell*.

(b) Incorrect action. **Full prompt:** "Pikachu sprints beneath windblown cherry blossoms and makes one clean leap across a narrow forest stream, landing on the opposite bank without stopping. Pink petals and water droplets sweep past as a low camera tracks beside it in warm sunrise light." **Generation seed:** 2026. **Human-verified failure:** Pikachu runs toward the camera through the stream; no opposite-bank landing occurs. **Localized failed token:** *across*.

(c) Incomplete structure. **Full prompt:** "A miniature jade dragon climbs out of a white porcelain teacup, curls clockwise around the rim, and exhales one perfect ring of steam. The camera slowly pushes in across an elegant wooden tea table, with warm morning light and delicate cinematic detail." **Generation seed:** 42. **Human-verified failure:** The steam remains diffuse and irregular rather than forming the requested perfect ring. **Localized failed token:** *ring*.

(d) Incorrect attribute. **Full prompt:** "A woman in a long ivory coat carries a bright red umbrella and walks from left to right through a rain-soaked Tokyo alley. Neon signs reflect in the puddles while the camera tracks smoothly beside her, cinematic night photography." **Generation seed:** 42. **Human-verified failure:** The umbrella is pale green/white rather than bright red. **Localized failed token:** *red*.

I COMPLETE EVALUATOR INSTRUCTION

Below is the system instruction used for full-video constraint evaluation. Line wrapping in the paper does not alter the evaluator input.

```
You are producing strict token-level prompt-following supervision for VideoProbe.

INPUT
You receive:
1. The complete generated video.
2. The exact text prompt.
3. prompt_tokens with fixed integer indices.
4. atomic_visual_constraints. Every constraint contains a constraint_id,
   constraint text, candidate_token_indices, and candidate_tokens.

GENERAL RULES
- Inspect the complete temporal sequence, not a thumbnail or sparse preview.
- Judge only the supplied constraints.
- A positive label may come only from that constraint's candidate_token_indices.
- Use status "satisfied", "violated", or "uncertain" for every constraint.
- Use the smallest candidate-token subset that precisely identifies a failure.
- Never replace a missing candidate with a semantically different token.
- Tokens outside supplied candidate spans have supervision_mask=0.

MOTION EXECUTION AND DIRECTION MUST BE JUDGED SEPARATELY
- An event-execution constraint contains the verb candidate, such as "drives".
- A motion-direction constraint contains the direction candidate, such as
  "rightward".
- If the requested motion occurs but in the wrong direction: mark the event
  constraint satisfied and mark only the direction constraint violated.
- If the requested motion does not occur at all: mark the event constraint
  violated and mark its dependent direction constraint uncertain.
- If the moving subject is missing or cannot be judged: mark the relevant
  object-existence constraint violated or uncertain, and mark both event and
  direction constraints uncertain as appropriate.
- Do not mark the motion verb merely because the direction is wrong.
- If a direction failure is visible but no direction candidate is supplied,
  do not fall back to the verb. Mark the affected constraint uncertain and
  explicitly report the candidate-contract problem in reasoning.
- Distinguish object motion from camera pan, camera motion, and zoom.

OTHER MINIMAL ATTRIBUTION RULES
- Wrong count with requested objects present: mark only the number token.
- Wrong visible attribute with the object present: mark only the attribute.
- Missing or replaced object: mark only the missing object noun through its
  object-existence constraint; dependent constraints become uncertain.
- Wrong spatial relation with all objects present: mark only the relation.
- Action absent with all participants present: mark the action verb.
- Missing or wrong state transition: mark the state-change verb or minimal
  state-change phrase.
- Uncertain is not a soft failure: label=0 and supervision_mask=0.

CONSISTENCY RULES
- For satisfied constraints, candidate tokens receive label=0 and mask=1.
- For violated constraints, applied problem tokens receive label=1 and mask=1.
- For uncertain constraints, candidate tokens receive label=0 and mask=0.
- problem_token_indices must exactly equal positions with label=1.
- applied_problem_token_indices must be a subset of the corresponding
  candidate_token_indices.
- The union of applied_problem_token_indices must equal problem_token_indices.
- token_label_sequence and token_supervision_mask must match prompt_tokens in
  length and order.
- Copy every supplied constraint_id, constraint, candidate_token_indices, and
  candidate_tokens exactly; do not rewrite the input contract.
```

```
Return exactly one JSON object with no markdown or extra text:
{
  "observed_video": "brief factual full-video description",
  "constraint_judgments": [
    {
      "constraint_id": "c1",
      "constraint": "exact supplied constraint",
      "status": "satisfied|violated|uncertain",
      "confidence": "high|medium|low",
      "evidence": "brief factual evidence",
      "candidate_token_indices": [0],
      "candidate_tokens": ["token"],
      "applied_problem_token_indices": [],
      "applied_problem_tokens": []
    }
  ],
  "token_label_sequence": [0],
  "token_supervision_mask": [0],
  "problem_token_indices": [],
  "problem_tokens": [],
  "uncertain_token_indices": [],
  "all_provided_constraints_satisfied": false,
  "reasoning": "concise explanation"
}
```

J Code and Data Availability

Upon publication, we will release the VideoProbe implementation, attention-capture hooks, frozen splits, constraint schemas, evaluator instruction, label validator, tokenizer-alignment code, evaluation scripts, and run configurations. AttnPulse-10K will include a dataset card covering prompt provenance, file schemas, generator revisions, and applicable licenses. For generated assets that cannot be redistributed, we will release stable identifiers and complete generation manifests.