The Pulse of Motion: Measuring Physical Frame Rate from Visual Dynamics

Xiangbo Gao[1] Mingyang Wu[1] Siyuan Yang[1] Jiongze Yu[1] Pardis Taghavi[1] Fangzhou Lin[1]
Zhengzhong Tu[1,2]
[1]Texas A&M University [2]Visko Platform

Abstract

Recent video generators produce visually smooth motion, yet their outputs are rarely grounded in a stable physical time scale. A clip saved at 24 FPS may depict motion that is perceptually closer to slow motion, acceleration, or a mixture of both across different segments. We call this failure mode chronometric hallucination*: the visual dynamics of a generated video imply an ambiguous and uncontrollable physical frame rate. We introduce Visual Chronometer, a predictor that estimates Physical Frames Per Second (PhyFPS) directly from visual dynamics, rather than trusting unreliable container metadata. The model is trained through controlled temporal resampling, including sharp capture, exposure blur, and rolling-shutter simulation, so that it learns motion-grounded time cues. We further build* `PhyFPS-Bench-Real` *for validating PhyFPS prediction and* `PhyFPS-Bench-Gen` *for auditing modern video generators. Across open- and closed-source systems, our measurements reveal large meta FPS–PhyFPS gaps and substantial intra- and inter-video temporal instability. Finally, PhyFPS-guided retiming significantly improves human-perceived temporal naturalness, suggesting that explicit time-scale measurement is a necessary step toward physically grounded video generation.*

1. Introduction

Modern video generators have made rapid progress in spatial realism and are increasingly described as physical world models [6, 9, 10, 16, 20]. Yet physical simulation requires more than smooth frame-to-frame interpolation: it requires a stable relationship between displacement and elapsed time. In practice, today's systems often generate motion whose visual speed is disconnected from the nominal frame rate of the output container. A video saved at 24 FPS may look as if it should be played at 35, 45, or even 60 FPS to match normal real-world motion.

We attribute this ambiguity to a common training practice: internet-scale video corpora mix normal footage, slow motion, time-lapse, edited clips, and metadata-corrupted files, while training pipelines often normalize them into a small set of saved frame rates. The model therefore observes many visually different physical speeds under the same metadata label. It can learn plausible kinematics, but not a reliable pulse of motion. We call the resulting failure **chronometric hallucination**: generated motion has an ambiguous, unstable, and weakly controllable physical time scale (Fig. 1).

This paper studies how to measure that pulse directly from visual dynamics. We define **Physical Frames Per Second** (PhyFPS) as the frame rate implied by the depicted physical motion, distinct from the file's nominal meta FPS. We then introduce Visual Chronometer, a regression model that estimates PhyFPS from raw frames. Unlike metadata-based approaches, Visual Chronometer asks what playback rate would make the visible motion correspond to real-world time. This lets us audit video generators, diagnose whether a generated clip is globally mis-timed or locally unstable, and retime outputs to improve perceptual naturalness.

Our contributions are threefold. First, we formalize chronometric hallucination and PhyFPS as a measurable time-scale property of video generation. Second, we train Visual Chronometer through controlled temporal resampling and camera-inspired augmentations, yielding a robust predictor of physical frame rate. Third, we introduce `PhyFPS-Bench-Real` and `PhyFPS-Bench-Gen` to validate prediction accuracy and to audit state-of-the-art generators. Our experiments show severe meta FPS–PhyFPS mismatch across current systems, while PhyFPS-guided post-processing is strongly preferred by human viewers.

2. Related Work

Video generation and world models. Large video generators now combine diffusion or autoregressive generation with temporal attention, 3D operators, and latent-space video representations [5, 9, 10, 16, 20]. These systems are increasingly evaluated not only as media synthesis tools, but also as candidates for physical world modeling. However, most architectures and training pipelines emphasize semantic alignment, spatial fidelity, and frame-to-frame smoothness. The physical duration represented by a frame transition is usually inherited from metadata or a fixed decoding convention,

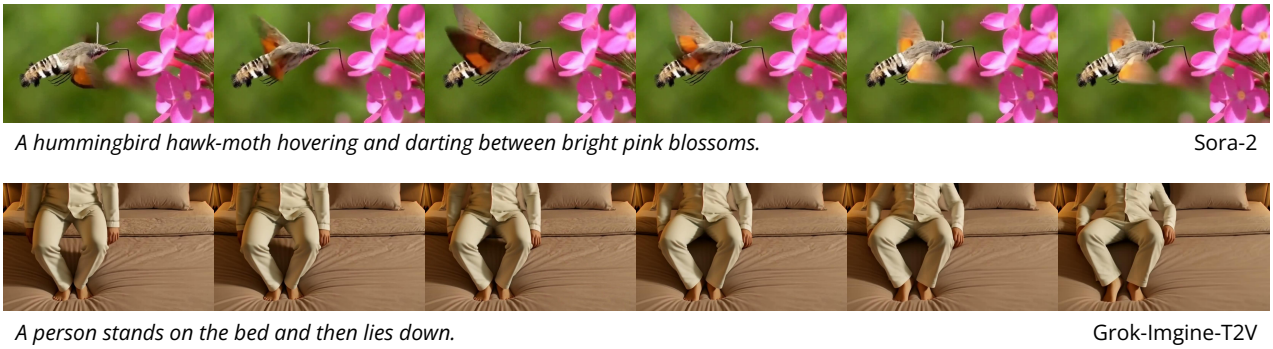

Figure 1. **Chronometric hallucination.** Generated videos can depict physically implausible time scales even without prompts such as "slow motion." Examples include a hummingbird hawk-moth rendered with slow wing beats and a falling person moving slower than gravity would suggest.

rather than supervised as a first-class condition. This leaves a gap between visually plausible motion and physically calibrated motion.

Visual time and speed perception. Prior work has shown that temporal structure is visually observable. SpeedNet-style self-supervision learns to distinguish normal-rate from sped-up videos, while arrow-of-time methods study whether a model can recover playback direction from dynamics. Hyperlapse and time-remapping methods further demonstrate that visual motion determines perceived temporal flow. These tasks, however, are usually categorical or relative: faster versus slower, forward versus backward, or which segments to sample. Visual Chronometer instead treats physical time scale as an absolute continuous variable and predicts a scalar PhyFPS.

Temporal evaluation for video generation. Standard video metrics such as FVD, LPIPS, and temporal consistency scores measure distributional realism or frame coherence, but they do not ask whether a generated action unfolds at the correct physical speed. Recent benchmark suites add dimensions for dynamics and physics-adjacent reasoning, yet time-scale calibration remains largely unmeasured. Our benchmarks complement these efforts by asking a specific chronometric question: given the visible motion, what physical frame rate does the video imply, and how stable is that rate within and across generated samples?

3. Visual Chronometer

3.1. Training data with known physical time

Training a PhyFPS predictor requires videos whose physical time scale is known. We therefore collect source clips from high-frame-rate and sensor-grounded datasets where the nominal frame rate is trusted to match real time, including Adobe240 [15], BVI-VFI [3], UVG [12], autonomous driving videos with sensor synchronization [14], physics-grounded human motion [11], and controlled in-house captures.

The central design constraint is that the training signal must not already contain chronometric ambiguity. We therefore avoid clips whose apparent speed may have been changed by editing, slow-motion export, or platform-side frame-rate conversion. We use the verified source clips as anchors where meta FPS and PhyFPS coincide, and then deliberately create new physical rates under known transformations. This gives the model supervision over time scale without relying on unreliable web-video metadata.

From these sources, we synthesize labeled PhyFPS variants through controlled temporal resampling. Each source video is first interpolated to a high-frequency base rate $F_H = 240$ FPS using RIFE [7]. For a target rate F_L, with $N = F_H/F_L$, we generate low-rate frames I_k^L from the high-rate sequence I^H under three camera-inspired mechanisms:

$$I_k^L = I_{\lfloor kN \rfloor}^H \qquad \text{sharp capture,} \tag{1}$$

$$I_k^L = \frac{1}{M} \sum_{i=0}^{M-1} I_{\lfloor kN \rfloor + i}^H \qquad \text{motion blur.} \tag{2}$$

For rolling shutter, a pixel at column x is sampled from a progressively shifted time index $\lfloor kN \rfloor + \lfloor Mx/W \rfloor$, where W is image width. We vary $M \in \{N, N/2, N/4\}$ to cover long, medium, and short exposures. This augmentation makes the model rely on intrinsic dynamics instead of memorizing scene categories or idealized displacement patterns.

3.2. Architecture and objective

Given a video clip $\mathbf{V} = \{I_t\}_{t=1}^T$, Visual Chronometer predicts the scalar PhyFPS implied by the observed motion. We use VideoVAE+ [19] as a spatiotemporal encoder, producing latent tokens $\mathbf{Z} = \mathrm{Enc}(\mathbf{V})$. A lightweight attention head pools these tokens with a learnable query, followed by an MLP that outputs $\hat{s} = \log \hat{y}$, the predicted log-PhyFPS.

Figure 2. **Physics-grounded temporal augmentation.** We synthesize low-rate videos from 240 FPS sources using sharp capture, exposure integration, and rolling-shutter simulation.

Query pooling lets the predictor operate on arbitrary clip lengths, while log-space regression emphasizes proportional timing error:

$$\mathcal{L}_{\log} = \frac{1}{n}\sum_{i=1}^{n}\left(\log y_i - \hat{s}_i\right)^2. \tag{3}$$

We train two variants. **VC-Wide** covers 18 rates from 2 to 240 FPS, while **VC-Common** focuses on common consumer/video-generation rates from 12 to 60 FPS. Unless otherwise stated, generator audits use VC-Common because modern text-to-video systems usually operate in this range. Models are optimized with Adam at 10^{-5} for 125k iterations on four RTX A6000 GPUs.

The two variants serve different purposes. VC-Wide is useful for diagnosing extreme capture regimes, including high-speed footage and heavily retimed clips. VC-Common is more specialized: by restricting the target support to common web and model-output frame rates, it trades range for precision in the region most relevant to generator auditing. At inference time, we avoid a single global prediction when possible. Instead, overlapping windows provide multiple local estimates, which improves robustness and exposes temporal drift within a clip.

3.3. Benchmarks and metrics

`PhyFPS-Bench-Real` contains 4,000 verified real clips split by source video, ensuring that training and testing do not share scene instances. It evaluates predictor accuracy with mean absolute error (MAE) and mean absolute percentage error (MAPE). `PhyFPS-Bench-Gen` contains 100 text prompts spanning humans, animals, vehicles, nature, fluids, camera motion, and close-up object dynamics. We deliberately exclude explicit speed-control phrases such as "slow motion" or "time-lapse" so that the default physical time scale can be measured.

The `PhyFPS-Bench-Gen` prompts are designed to contain observable dynamics rather than static scenes: humans walking or striking objects, animals with characteristic motion frequencies, vehicles with rigid-body displacement, fluids and fire with stochastic motion, and camera motions that create global optical flow. This diversity is important because a PhyFPS predictor should not merely recognize a few canonical actions. It must infer time scale from the coupling between scene content, displacement, blur, and temporal continuity.

Table 1. **`PhyFPS-Bench-Gen` audit of generated time scale.** PhyFPS is the physical frame rate predicted from visual dynamics. Lower error and lower CV indicate better time-scale fidelity and stability.

Model	Meta FPS	PhyFPS	Avg. Error ↓	Pct. Error ↓	Intra CV ↓	Inter CV ↓
Open-source models						
CogVideoX-2B	24	33.64	12.46	52%	0.11	0.46
CogVideoX-5B	24	38.26	17.96	75%	0.12	0.52
HunyuanVideo	24	35.89	13.82	58%	0.12	0.36
Wan2.1-T2V-1.3B	24	26.28	7.54	31%	0.11	0.38
Wan2.1-T2V-14B	24	32.37	10.87	45%	0.14	0.36
Wan2.2-T2V-A14B	24	31.52	10.74	45%	0.12	0.38
Wan2.2-TI2V-5B	24	32.81	11.63	48%	0.15	0.38
InfinityStar (5s)	16	34.41	18.46	115%	0.11	0.38
InfinityStar (10s)	16	36.15	20.19	126%	0.16	0.36
LTX-Video	24	46.52	23.67	99%	0.10	0.33
LTX-2	25	39.77	15.70	63%	0.13	0.34
Closed-source models						
Seedance-1.0-Lite	24	28.60	8.31	35%	0.15	0.37
Seedance-1.5-Pro	24	33.69	10.67	44%	0.16	0.25
Sora-2	30	36.21	8.40	28%	0.13	0.29
Grok-Imagine-T2V	24	36.97	13.97	58%	0.16	0.28
Kling-o3	24	30.04	9.10	38%	0.15	0.34
Veo-3.1-Fast	24	35.83	13.62	57%	0.17	0.33

For each generated video v, we extract overlapping 32-frame clips with stride 4 and predict clip-level rates $\hat{f}_{v,c}$. The video-level and model-level physical rates are

$$\bar{f}_v = \frac{1}{C_v}\sum_{c=1}^{C_v}\hat{f}_{v,c}, \qquad \hat{F} = \frac{1}{V}\sum_{v=1}^{V}\bar{f}_v. \tag{4}$$

We report meta FPS alignment via average absolute error $|\bar{f}_v - F_{\text{meta}}|$ and percentage error. Temporal stability is measured with coefficient of variation (CV): **intra-video CV** measures fluctuation across clips within a generated video, while **inter-video CV** measures variation across prompts from the same generator.

4. Experiments

4.1. Auditing generative video models

Table 1 summarizes our `PhyFPS-Bench-Gen` audit. The evaluated generators include open-source systems from the Wan [16], LTX [5, 6], CogVideoX [20], HunyuanVideo [9], and InfinityStar [10] families, as well as closed-source systems including Veo, Sora, Grok Imagine, Kling, and Seedance [2, 4, 8, 13, 18].

The dominant pattern is clear: saved frame rate and physical frame rate are frequently misaligned. Most systems produce videos whose visual dynamics imply a higher PhyFPS than the container meta FPS, consistent with the common observation that generated videos often appear "slow but smooth" [17]. Wan2.1-1.3B has the lowest open-source average error, while LTX models have strong stability but large absolute mismatch, suggesting that their outputs may be

Table 2. **Accuracy on `PhyFPS-Bench-Real`.** The average ground-truth PhyFPS is 38.81.

Model	Avg. Pred.	MAE ↓	MAPE ↓
VC-Common	39.20	**3.46**	**9%**
VC-Wide	45.48	7.76	21%
Gemini-3.1-Pro (video)	31.00	21.67	43%
Seed-1.6 (video)	29.60	20.40	41%
Qwen3.5+ (video)	4.46	45.54	91%
Gemini-3.1-Pro (images)	5.15	44.85	90%
Seed-1.6-Flash (images)	30.00	20.00	40%
Qwen3.5-397B (images)	22.03	27.97	56%

internally coherent yet saved at an unsuitable nominal rate. Closed-source models are somewhat better aligned, but they still show sizeable errors and nontrivial intra/inter-video CV, indicating that commercial-scale training does not eliminate chronometric hallucination.

The distinction between alignment and stability is important. A model can be globally miscalibrated but stable, in which case a single metadata correction may substantially improve playback. Conversely, a model can have a reasonable average PhyFPS but large intra-video CV, implying that different segments of the same clip operate at different physical speeds. This second failure is more problematic for world modeling because it cannot be fixed by simply changing the file header or playback rate. `PhyFPS-Bench-Gen` therefore separates absolute error, intra-video stability, and inter-video stability rather than collapsing them into one score.

4.2. Predictor validation and VLM baselines

We validate Visual Chronometer on the real-video benchmark and compare against strong VLMs asked to estimate the physical frame rate from either video input or unrolled image sequences. Table 2 shows that VC-Common achieves 3.46 FPS MAE and 9% MAPE, substantially outperforming VC-Wide and all VLM baselines. The narrower model performs better because the target range matches common generated-video speeds. In contrast, VLMs often collapse to generic answers such as 24 or 30 FPS, or fail to preserve frame timing after their own video subsampling. Even image-sequence prompting does not resolve the issue, suggesting that general-purpose VLMs do not currently possess a calibrated internal motion clock.

4.3. Perceptual value of PhyFPS correction

To test whether measured PhyFPS corresponds to human temporal naturalness, we use Visual Chronometer as a post-processing tool. For each generated video, we compare the original output against two retimed variants: **Pred**, which uses one global predicted PhyFPS, and **Pred Dyn**, which applies local clip-level retiming. We collect 1,490 pairwise comparisons from more than 15 participants and fit a Bradley–Terry preference model [1].

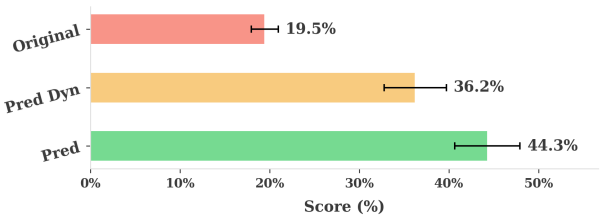

Figure 3. **Human preference.** Retimed videos are preferred.

As shown in Fig. 3, users strongly prefer both corrected variants over the original videos. The global correction obtains the highest Bradley–Terry score (44.2%), followed by dynamic local correction (36.9%) and the original output (19.0%). This supports two conclusions. First, Visual Chronometer is not only numerically accurate on real clips, but also captures a perceptually meaningful time-scale signal for generated videos. Second, locally varying playback speed can introduce its own perceptual discontinuities; for short generated clips, a single averaged physical frame rate is often the smoother correction.

4.4. Ablation

Physics-grounded augmentation is important for robust prediction. A model trained only with uniform temporal subsampling reaches 5.12 FPS MAE on `PhyFPS-Bench-Real`. Adding exposure blur improves MAE to 4.87, while the full VC-Common training recipe with rolling-shutter simulation reaches 3.46. Temporal context length also matters: very short clips lack enough displacement evidence, while extremely long clips reduce the benefit of sliding-window averaging. In practice, 32–64 frames provide the best balance between local sensitivity and stable estimation.

This gain is not simply due to more data: exposure blur and rolling shutter encode camera physics that are often the very cues revealing physical speed in real footage.

5. Discussion and Conclusion

Chronometric hallucination is not a claim that every generated video must run at real-time speed: slow motion and time-lapse are valid creative controls. The issue is controllability. A generator that cannot reliably produce a default $1\times$ physical time scale will also struggle to obey deliberate $0.5\times$ or $2\times$ speed controls.

We presented Visual Chronometer, a visual predictor of PhyFPS trained through physically motivated temporal resampling. Using `PhyFPS-Bench-Real` and `PhyFPS-Bench-Gen`, we showed that modern generators suffer from large meta FPS–PhyFPS gaps and that VLMs are unreliable temporal judges. Because PhyFPS-guided retiming improves human preference, explicit time-scale measurement can serve as both an immediate post-processing tool and a future supervision signal for temporally grounded video generation.

References

[1] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. 4

[2] ByteDance Seed Team. ByteDance Seed: Models and research. https://seed.bytedance.com/, 2025. Accessed: 2026-02-27. 3

[3] Duolikun Danier, Fan Zhang, and David R Bull. Bvi-vfi: A video quality database for video frame interpolation. *IEEE Transactions on Image Processing*, 32:6004–6019, 2023. 2

[4] DeepMind. Veo 3 technical report. Technical report, DeepMind, 2025. Accessed: 2026-02-18. 3

[5] Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, et al. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024. 1, 3

[6] Yoav HaCohen, Benny Brazowski, Nisan Chiprut, Yaki Bitterman, Andrew Kvochko, Avishai Berkowitz, Daniel Shalem, Daphna Lifschitz, Dudu Moshe, Eitan Porat, et al. Ltx-2: Efficient joint audio-visual foundation model. *arXiv preprint arXiv:2601.03233*, 2026. 1, 3

[7] Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. Real-time intermediate flow estimation for video frame interpolation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2

[8] Kling AI. Kling AI Omni / VIDEO O1 creative interface, 2025. 3

[9] 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. 1, 3

[10] Jinlai Liu, Jian Han, Bin Yan, Hui Wu, Fengda Zhu, Xing Wang, Yi Jiang, Bingyue Peng, and Zehuan Yuan. Infinitystar: Unified spacetime autoregressive modeling for visual generation. *arXiv preprint arXiv:2511.04675*, 2025. 1, 3

[11] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *2017 international conference on 3D vision (3DV)*, pages 506–516. IEEE, 2017. 2

[12] Alexandre Mercat, Marko Viitanen, and Jarno Vanne. Uvg dataset: 50/120fps 4k sequences for video codec analysis and development. In *Proceedings of the 11th ACM multimedia systems conference*, pages 297–302, 2020. 2

[13] OpenAI. Sora: Creating video from text, 2024. 3

[14] Vasili Ramanishka, Yi-Ting Chen, Teruhisa Misu, and Kate Saenko. Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7699–7707, 2018. 2

[15] Shuochen Su, Mauricio Delbracio, Jue Wang, Guillermo Sapiro, Wolfgang Heidrich, and Oliver Wang. Deep video deblurring for hand-held cameras. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1279–1288, 2017. 2

[16] 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. 1, 3

[17] Ziyi Wu, Anil Kag, Ivan Skorokhodov, Willi Menapace, Ashkan Mirzaei, Igor Gilitschenski, Sergey Tulyakov, and Aliaksandr Siarohin. Densedpo: Fine-grained temporal preference optimization for video diffusion models. *arXiv preprint arXiv:2506.03517*, 2025. 3

[18] xAI. Grok Imagine — ai image & video generation by xai, 2026. 3

[19] Yazhou Xing, Yang Fei, Yingqing He, Jingye Chen, Jiaxin Xie, Xiaowei Chi, and Qifeng Chen. Large motion video autoencoding with cross-modal video vae. *arXiv preprint arXiv:2412.17805*, 2024. 2

[20] 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. *arXiv preprint arXiv:2408.06072*, 2024. 1, 3